SIIM-TR-A-**03-02**

SIIM Technical Report

Budget Balanced Mechanisms for the Multicast Pricing Problem with Rates

by

Markus Bläser

Schriftenreihe der Institute für

Informatik/Mathematik

Serie A

February 12, 2003, revised June 5, 2003

Universität zu Lübeck
Technisch-Naturwissenschaftliche Fakultät

Email: blaeser@informatik.mu-luebeck.de

Phone: +49-451-7030-417
Fax: +49-451-7030-438

Budget Balanced Mechanisms for the Multicast Pricing Problem with Rates

Markus Bläser

June 5, 2003

Abstract

We consider the design of mechanisms for pricing multicast transmissions. Most mechanism provided for the multicast pricing problem are binary, that is, either a user gets the full transmission or nothing at all. In a network with widely differing bandwidth connections—such as the internet—it is however essential to have transmission of data at different rates. Adler and Rubenstein [1] propose two approaches, a layered approach and a split session approach, and analyze marginal costs mechanisms on fixed multicast trees within both approaches.

We investigate budget balanced mechanisms for the multicast pricing problem with rates. It is natural to try to construct rated mechanism for the multicast pricing problem from binary ones. We show that for the layered paradigm, there is a corresponding rated mechanism that is strategyproof and budget balanced whenever the binary mechanisms are group strategyproof and budget balanced, whereas for the split session paradigm, such a mechanism does not exist.

Next we consider the case when there is no underlying fixed multicast tree. Thus, our problem is now twofold: First we have to solve an optimization problem, then we have to design a mechanism for the computed structure. We solve the combined optimization/mechanism problem for transmission at different rates by plugging a binary mechanism by Jain and Vazirani [6] into our mechanism for the problem with rates. We generalize the mechanism by Jain and Vazirani to a wider range of multicast problems by combining it with ideas by Goemans and Williamson [5].

1 Introduction

Multicast transmissions allow huge savings of network traffic compared to unicast transmission when the same data is sent to a lot of users. These savings are achieved by the fact that users may "share" links, since each node in a multicast network can send an incoming transmission to an arbitrary number of neighbours. If there are costs incurred when using an edge, then this sharing of edges is an obstacle for pricing. Such pricing issues and the design of multicast networks in general under socio-economic and game-theoretic aspects has received a lot of attention recently.

Formally, the (binary) multicast pricing problem is defined as follows: Let $G = (V, E)$ be an edge weighted undirected graph. The graph G models the underlying network, the edge weight c_e represents the costs for using the edge e. There is a distinguished set $N \subseteq V$, which represents the users to be served. We assume w.l.o.g. that $N = \{1, \dots, n\}$. (Note that V has more than n nodes.) Furthermore, there is a distinguished node $r \notin N$, the service provider. A cost-sharing mechanism determines which users receive the transmission and assigns a price to each of these users. Each of the users $i \in N$ has a (secret) utility u_i. He derives a benefit of u_i from getting the transmission. If i gets the transmission at price x_i, his individual welfare is $u_i - x_i$. If i does not get the transmission, his welfare is $-x_i$. However, the cost sharing mechanism does not a priori know the values u_i. Thus it has to rely on the users to report these values. The users are considered selfish and thus might not be willing to report their true utility. In a game-theoretic framework, their set of strategies is to report any value $b_i \geq 0$ as their utility. Given these bids b_i of the users, our task is to select a subset Q of the users, find a multicast tree F serving Q, and assign prices x_i to the users. The complexity of computing F heavily depends on the structure of the network. We will address this point later. The cost-sharing mechanism for the tree F should meet some of the following socio-economic and game-theoretic properties:

No Positive Transfer (NPT): For all i, $x_i \geq 0$.

Voluntary Participation (VP): If $i \in Q$, then $b_i - x_i \geq 0$, otherwise $x_i = 0$, that is, no user is charged more then he is willing to pay and only user getting the transmission are charged.

Consumer Sovereignty (CS): Every user is included as long as his bid b_i is high enough (independent of the bids of the other users).

Group Strategyproof (GSP): Even if a set of users C collude, their dominant strategy is to report their true utility u_i as b_i for all $i \in C$. If this property holds only for sets C of size one, then we speak of **Strategyproof (SP)**.

Budget Balance (BB): $\sum_{i \in Q} x_i = \sum_{e \in F} c_e$, i.e., neither a deficit nor a surplus is created. (Equality may look strange at a first glance but is necessary in a competetive environment.)

Efficiency (EFF): The overall (reported) welfare $\sum_{i \in Q} b_i - \sum_{e \in F} c_e$ is maximized.

It is a classical result in game theory that there is no strategyproof mechanism that meets both BB and EFF. From a computational point of view, we also want that the mechanism can be computed efficiently. In a distributed setting, it might also be desirable that the mechanism can be computed with low communication costs.

1.1 Related Work

Most of the current pricing mechanisms for multicast transmission assume that the underlying multicast tree is fixed, see for instance [1, 2, 3, 4]. Thus for any subset of the user to be served, the used tree is a subtree of the underlying fixed tree. From the viewpoint of combinatorial optimization, this problem is not very interesting. (However computing a tree that maximizes efficiency is NP-hard and even hard to approximate [4].) For fixed trees, mechanisms are designed and analyzed that meet—beside NPT, VP, and CS—either GSP and BB or SP and EFF. The work of Jain and Vazirani [6] is a notable exception, as they do not assume that there is a fixed multicast tree. They design a class of mechanisms that meet NPT, VP, CS, GSP, and BB and that moreover compute for a chosen set of users a multicast tree whose costs are twice within the costs of an optimum Steiner tree, provided that the edge costs fulfill the triangle inequality.

All of the pricing mechanism mentioned above are binary, that means, either a user gets the full transmission or no transmission at all. In a network with widely differing bandwidth connections, it is however unavoidable to have transmission of data at different qualities or *rates*. Adler and Rubenstein [1] propose two approaches to handle different rates, a layered approach and a split session approach (see the next section for details). For both approaches, Adler and Rubinstein give mechanisms (based on marginal costs) that meet NPT, VP, CS, SP, and EFF. They assume that a fixed multicast tree is given. They do not treat budget balanced mechanisms or general graphs and pose those extensions as an open problem.

1.2 Our Results

Since a binary mechanism can be interpreted as a rated mechanism with only one possible rate, it is a natural to construct rated mechanisms from binary ones. We design a meta mechanism under the layered paradigm which uses a binary mechanism for each layer as a blackbox (Theorem 1). This meta mechanism is interesting on its own and can be applied to other pricing problems with rates. The constructed meta mechanism is NPT, VP, CS, SP, and BB, if the binary mechanisms meet NPT, VP, CS, GSP, and BB. It remains an open question whether one can also achieve GSP for such a meta mechanism. Once we have this meta mechanism, we can plug various binary mechanisms into it. If the underlying multicast tree is fixed, we can for instance use the Shapley value, see e.g. [4]. Alternatively, we can use approximations of it to reduce the communication complexity, see e.g. [2]. If there is no underlying fixed multicast tree, then we can exploit the binary mechanism by Jain and Vazirani [6] to get a mechanism for the multicast problem with rates under the layered paradigm that meets NPT, VP, CS, SP, and BB (Theorem 3). This mechanism works for general graphs and computes for each layer a multicast tree whose weight is at most twice the weight of an optimum Steiner tree, provided that the edge weights fulfill the triangle inequality.

Then we show that for the split session paradigm, such a meta mechanism does not exists (Theorem 2) in a sense to be made precise in the next section, even if we have only two users and two possible rates. This insight complements nicely the results by Adler and Rubenstein that the split session paradigm is also harder than the layered paradigm in their setting.

Finally, we extend the techniques of Jain and Vazirani to a larger class of constrained forest problems by incorporating ideas of Goemans and Williamson [5]. This allows us to model extended multicast scenarios like having simultaneous (parallel) transmissions or several (mirrored) servers. Like in the work of Jain and Vazirani, we assume that there is a centralized authority (for instance the root, i.e., the service provider) who executes all necessary computations. This is also allowed in the model of Nisan and Ronen [9, 10] but might be questionable in a distributed setting.

2 Mechanisms for Problems with Rates

Most of the proposed mechanisms provided for the multicast pricing problem are *binary*, that is, every customer either receives the full transmission or nothing at all. In a network with widely differing bandwidth connections—such as the internet—we have to allow transmission of data at different rates $\rho_1 \le \rho_2 \le \cdots \le \rho_\ell$, where the number of rates ℓ is determined in advance. In the context of mechanism design, Adler and Rubenstein [1] propose two approaches, which both reflect practice: Under the *layered* paradigm, the transmission is sent in layers. Layer 1 has rate ρ_1 and every other layer $i > 1$ has rate $\rho_i - \rho_{i-1}$. To receive rate ρ_j, a user is sent layers $1, \dots, j$. Under the *split session* paradigm, there is a separate multicast transmission for each rate. Each user receives at most one of those multicast transmissions. Adler and Rubenstein study *marginal costs* mechanism under those paradigms.

We here study another class of mechanisms for those problems, namely mechanisms that meet NPT, VP, CS, SP and BB, as defined below. Each user i now has a utility vector $u_i = (u_{i,1}, \dots, u_{i,\ell})$ and $u_{i,\lambda}$ is the benefit of user i when receiving the transmission at rate ρ_λ. The possible strategies of each user i is to bid a vector $b_i = (b_{i,1}, \dots, b_{i,\ell})$, where $b_{i,\lambda} \ge 0$ indicates the price that i is willing to pay for rate ρ_λ. We are studying mechanisms that, given those n bids $b = (b_1, \dots, b_n)$, compute a function $q : N \to \{0, \dots, \ell\}$. (In the case of binary mechanisms, q simply is a characteristic function.) For each user i, $q_i := q(i)$ is the rate of the transmission received by i. $q_i = 0$ means that the user does not receive the transmission at all. For convenience, instead of saying that a user gets the transmission at rate ρ_i, we will often simply say that the user gets the ith rate or even shorter (and sloppier) rate i. Such a function q will be called a *rate function.* The mechanism also provides a function $x : N \to \mathbb{R}$. $x_i := x(i)$ denotes the price that user i has to pay to receive the transmission at rate ρ_{q_i}. The individual welfare of user i is $u_{i,q_i} - x_i$ provided that $q_i > 0$, since he gets the transmission at rate ρ_{q_i} for the price x_i. Otherwise, his welfare is $-x_i$. Finally, $\mathrm{Cost}(q)$ denotes the true costs incurred by the service provider when serving the users at rates according to the function q.

The properties NPT, VP, CS, (G)SP, and BB are refined as follows to handle multiple rates.

No Positive Transfer (NPT): For all users i, $x_i \geq 0$.

Voluntary Participation (VP): If $q_i > 0$, then $b_{i,q_i} - x_i \geq 0$, otherwise $x_i = 0$.

Consumer Sovereignty (CS): For every user i and for every rate r_λ, there is an ℓ-vector $\hat{b}_i^\lambda$ such that if i bids $\hat{b}_i^\lambda$ then i will get the service at rate ρ_λ (independent of the other bids).

Group Strategyproof (GSP): Even if a set of users C collude, their dominant strategy is to report their true utility vector u_i as b_i for all $i \in C$. If this property holds only for sets C of size one, then we speak of **Strategyproof (SP)**.

Budget Balance (BB): $\sum_{i\in N} x_i = \mathrm{Cost}(q)$, i.e., neither a deficit nor a surplus is created.

Since budget balance is hard to achieve in general, Jain and Vazirani [6] proposed the following relaxation:

α-approximate Budget Balance (α-BB): $\mathrm{Cost}(q) \leq \sum_{i\in N} x_i \leq \alpha \cdot \mathrm{Cost}(q)$.

A binary mechanism for the multicast pricing problem can be interpreted as a rated mechanism with only one possible rate. Since such mechanisms are well studied, it is a natural design paradigm to construct rated mechanism for the multicast pricing problem from binary ones.

Let $M_1, M_2, \ldots, M_\ell$ be mechanisms for rates $\rho_1, \rho_2 - \rho_1, \ldots, \rho_\ell - \rho_{\ell-1}$ under the layer paradigm or for rates $\rho_1, \rho_2, \ldots, \rho_\ell$ under the split session paradigm, respectively, such that all of $M_1, M_2, \ldots, M_\ell$ meet NPT, VP, CS, GSP, and BB. Moulin [7] (see also Moulin and Shenker [8]) has shown that for each such mechanism M_λ, there is a cross-monotonic cost sharing function ξ_λ such that $\xi_\lambda(q,i)$ is exactly the costs user i has to pay if the mechanisms selects users according to the characteristic function q. This function ξ_λ is budget balanced, that is, $\sum_{i=1}^n \xi_\lambda(q,i) = \mathrm{Cost}(q)$. Furthermore, there is a generic mechanism M such that $M(\xi_\lambda)$ is welfare equivalent to M_λ. Thus we only work with the cost sharing functions $\xi_1, \ldots, \xi_\ell$ in the following.

For a function $q : N \to \{0, 1, \ldots, \ell\}$ and $1 \leq k \leq \ell$, let $q_{=k}$ and $q_{\leq k}$ denote the characteristic functions defined by

$$[q_{=k}]_i = \begin{cases} 1 & \text{if } q_i = k, \\ 0 & \text{otherwise,} \end{cases} \qquad \text{and} \qquad [q_{\leq k}]_i = \begin{cases} 1 & \text{if } 1 \leq q_i \leq k, \\ 0 & \text{otherwise,} \end{cases}$$

respectively. $q_{=k}$ is the characteristic function of all users to whom rate ρ_k is assigned. $q_{\leq k}$ is the characteristic function of all users to whom one of the rates $\rho_1, \dots, \rho_k$ is assigned. Now the fact that the rated mechanism should be composed of binary mechanisms manifests in one of the following two properties.

Layered Costs (LC): For all users i, $x_i = \sum_{\lambda=1}^{q_i} \xi(q_{\leq\lambda}, i)$

Split Session Costs (SSC): For all users i, $x_i = \xi(q_{=q_i}, i)$.

Under the layered paradigm, the price x_i user i has to pay is exactly the sum of the first q_i cost shares of i with respect to $\xi_1, \dots, \xi_\ell$, since to get rate ρ_{q_i} user i has to receive the first q_i layers. Under the split session paradigm, the price is simply $\xi(q_{=q_i}, i)$, the share of i in the q_ith group.

2.1 Layered Paradigm

Figure 1 shows a mechanism, called L, which assigns rates q to the users and computes the prices x they have to pay. The inputs of L are the cost sharing functions as well as the bids of the users. In the description of L, $q :\equiv c$ means that the rate function $q : N \to \{1, \dots, \ell\}$ is set to the constant function that assigns to all users the value c. For a rate function r, let $r|_{i\mapsto k}$ denote the function that is defined by

$$[r|_{i\mapsto k}]_j := \begin{cases} k & \text{if } j = i, \\ r_j & \text{otherwise.} \end{cases}$$

The price x_i user i has to pay is $\sum_{k=1}^{q_i} \xi_k(q_{\leq k}, i)$. Thus, we can associate the cost sharing function $\xi(q, i) := \sum_{k=1}^{q_i} \xi_k(q_{\leq k}, i)$ with L. In particular, L meets LC. Since each ξ_k is cross-monotonic, ξ fulfills the following property, which we call *rate cross-monotonicity*:

$$q \leq q' \text{ and } q_i = q'_i \quad \Longrightarrow \quad \xi(q, i) \geq \xi(q', i) \tag{1}$$

where $q, q' : N \to \{0, 1, \dots, \ell\}$ are rate functions and $q \leq q'$ is meant componentwisely.

Mechanism L first assigns to each user rate 0. With increasing λ it raises the rate of each user. In step 5 it then drops each user which has a greater individual welfare by retaining its old rate $q_i^{t,\lambda-1}$ instead of λ. By the following lemma, this process is independent of the particular order in which the users are considered.

Lemma 1 *Let* $b_{i,\lambda} - \sum_{k=1}^{\lambda} \xi_k(q_{\leq k}, i) < b_{i,\lambda'} - \sum_{k=1}^{\lambda'} \xi_k(q_{\leq k}, i)$ *with* $\lambda' < \lambda$ *and let* $q' \leq q$. *Then* $b_{i,\lambda} - \sum_{k=1}^{\lambda} \xi_k(q'_{\leq k}, i) < b_{i,\lambda'} - \sum_{k=1}^{\lambda'} \xi_k(q'_{\leq k}, i)$ *holds, too.*

Proof. Note that $\sum_{k=\lambda'+1}^{\lambda} \xi_k(q'_{\leq k}, i) \geq \sum_{k=\lambda'+1}^{\lambda} \xi_k(q_{\leq k}, i)$, since $q_{\leq k} \geq q'_{\leq k}$ and each ξ_k is cross-monotonic. ■

Input: cost sharing functions $\xi_1, \ldots, \xi_\ell$,
bids $b = (b_1, \ldots, b_n)$.
Output: rate function q,
prices x.

1. Set $q^{0,0} :\equiv 0$.
2. Set $t := 0$.
3. Set $\lambda := 1$.
4. Define a rate function p by
$$p_i := \begin{cases} q_i^{t,\lambda-1} & \text{if } q^{t,\lambda-1} \geq \lambda, \\ \lambda & \text{otherwise.} \end{cases}$$
5. Choose a user i with $p_i = \lambda$ such that
$$b_{i,\lambda} - \sum_{k=1}^{\lambda} \xi_k(p_{\leq k}, i) < b_{i,q_i^{t,\lambda-1}} - \sum_{k=1}^{q_i^{t,\lambda-1}} \xi_k([p|_{i \mapsto q_i^{t,\lambda-1}}]_{\leq k}, i).$$
If such a user is found, set $p_i := q_i^{t,\lambda-1}$ and repeat this step.
6. Set $q^{t,\lambda} := p$.
7. Set $\lambda := \lambda + 1$. If $\lambda \leq \ell$ then goto step 4.
8. If $q^{t-1,\ell} \neq q^{t,\ell}$ then set $q^{t+1,0} := q^{t,\ell}$, set $t := t + 1$, and goto step 3.
9. Set $q := q^{t,\ell}$.
10. Define x via $x_i := \sum_{k=1}^{q_i} \xi_k(q_{\leq k}, i)$.

Figure 1: Mechanism L (for the layered paradigm)

Step 5 is basically an extension of the mechanism by Moulin [7]. After λ has reached ℓ, we iterate this process until no changes occur. This iteration is essential: After some user were raised to some rate k, other user might benefit from being raised to some rate $< k$, since in the layered approach a user has to pay for rates $1, \ldots, k$ in order to get rate k.

We first show that L is strategyproof provided that $\xi_1, \ldots, \xi_\ell$ are cross-monotonic.

Lemma 2 *Assume that $\xi_1, \ldots, \xi_\ell$ are cross-monotonic. Let $u = (u_1, \ldots, u_n)$ be the true utilities of the users. Assume that a user i_0 manipulates at u by $\hat{u}$ (where $u_j = \hat{u}_j$ for all $j \neq i_0$). Let q and $\hat{q}$ be the output of L on input u and $\hat{u}$, respectively. If $u_{\hat{q}_{i_0}} - \xi(\hat{q}, i_0) \geq u_{q_{i_0}} - \xi(q, i_0)$, then this holds with equality.*

Proof. Set $C = \{i_0\}$. (We formulate the first part of the proof for an arbitrary coalition, since it sheds some light on the question whether L can be made GSP.) First, observe that for all $i \in N$

$$u_{i,q_i} \geq u_{i,j} \quad \text{for all } j \leq q_i. \tag{2}$$

This is due to the fact, that to receive rate q_i, the user has to pay for all layers $1, \dots, q_i$. Thus a user can only stay in layer q_i in step 5, if (2) holds.

Define a rate function m by $m_i = \max\{q_i, \hat{q}_i\}$ for all $i \in N$. By construction, $m \geq q$ and $m \geq \hat{q}$. We first show that

$$u_{i,m_i} - \xi(m,i) \geq u_{i,q_i} - \xi(q,i) \quad \text{for all } i \in N \setminus C. \tag{3}$$

For users i with $m_i = q_i$, this follows from rate cross-monotonicity (1), as $m \geq q$ implies $\xi(m,i) \leq \xi(q,i)$. If now $m_i > q_i$ (and therefore $m_i = \hat{q}_i$), then

$$u_{i,m_i} - \xi(m,i) = u_{i,\hat{q}_i} - \xi(m,i). \tag{4}$$

Since $i \notin C$, $\hat{u}_{i,k} = u_{i,k}$ for all $1 \leq k \leq \ell$. In particular $u_{i,\hat{q}_i} = \hat{u}_{i,\hat{q}_i}$. Since $\hat{q}$ has been produced by a run of L on input $\hat{u}$, $\hat{u}_{i,\hat{q}_i} \geq \hat{u}_{i,j}$ holds for all $j \leq \hat{q}_i$, cf. (2). Altogether, $u_{i,\hat{q}_i} \geq u_{i,q_i}$. Thus (3) follows from (4), $m \geq q$, and rate cross-monotonicity of ξ.

Next, we show

$$u_{i,m_i} - \xi(m,i) \geq u_{i,\hat{q}_i} - \xi(\hat{q},i) \quad \text{for all } i \in C. \tag{5}$$

If $m_i = \hat{q}_i$, then (5) follows from rate cross-monotonicity, as $m \geq \hat{q}$. If $m_i > \hat{q}_i$ (and therefore $m_i = q_i$), then

$$u_{i,m_i} - \xi(m,i) \geq u_{i,q_i} - \xi(\hat{q},i)$$

follows from $m \geq \hat{q}$ and rate cross-monotonicity. Since $q_i > \hat{q}_i$, $u_{i,q_i} \geq u_{i,\hat{q}_i}$ follows from (2). This proves (5).

From (5),

$$u_{i,m_i} - \xi(m,i) \geq u_{i,q_i} - \xi(q,i) \quad \text{for all } i \in C. \tag{6}$$

follows from the assumption $u_{\hat{q}_i} - \xi(\hat{q},i) \geq u_{q_i} - \xi(q,i)$ for all $i \in C$. Now assume to the contrary, that coalition C successfully manipulates, that is, $u_{\hat{q}_{i_0}} - \xi(\hat{q},i_0) > u_{q_{i_0}} - \xi(q,i_0)$ holds for at least one $i_0 \in C$. Then (6) is strict for i_0.

There is a j such that $m_j > q_j$. Otherwise $m = q$ and (6) would be an equality for all $i \in C$. Consider the last iteration t_0 of the loop in steps 3 to 8 on input u. Since this is the last iteration t_0, $q^{t_0-1,\ell} = q^{t_0,0} = \cdots = q^{t_0,\ell} = q$ holds. In particular, in every iteration of the loop in step 5, $p \geq q$ holds. Consider the m_jth iteration of the loop in steps 4 to 7 (within the last iteration t_0 of the outer loop). Since $m_j > q_j$, user j is then dropped from rate m_j to $q_j^{t_0,m_j} = q_j$ in the loop in step 5. This can only happen if

$$u_{j,m_j} - \xi(p,j) < u_{j,q_j} - \xi(p|_{j\mapsto q_j}, j). \tag{7}$$

We have $j \in C$. Otherwise, $u_{j,m_j} = u_{j,\hat{q}_j} \geq u_{j,q_j}$, see the argumentation right after (4). This contradicts (7), since $p \geq p|_{j\mapsto q_j}$.

Since $p \geq q$, (7) and Lemma 1 implies

$$u_{j,m_j} - \xi(q|_{j\mapsto m_j}) < u_{j,q_j} - \xi(q,j).$$

As $\#C = 1$, there is only one $j \in C$, namely $j = i_0$. For all other $i \neq i_0$, $m_i = q_i$, since each j with $m_j > q_j$ fulfills $j \in C$. Thus $q|_{j\mapsto m_j} = m$. This contradicts (6). ∎

We only used $\#C = 1$ in the last step. It remains open, whether L or a variant of it is group strategyproof. We do not know any counter example.

Theorem 1 *If $\xi_1, \dots, \xi_\ell$ are cross-monotonic and budget balanced, then mechanism L meets NPT, VP, CS, SP, BB, and LC. If each ξ_λ is only α_λ-approximate budget balanced, then L is only α-approximate budget balanced, where $\alpha = \max\{\alpha_1, \dots, \alpha_\ell\}$.*

Proof. NPT: Obviously fulfilled by construction.

VP: Also fulfilled by construction, since ultimately $q_i = 0$ means that a user is not charged anything at all.

CS: Assume that user i wants to obtain the transmission at rate ρ_j. Since ξ_λ meets CS, there is a bid β_λ such that i gets the (binary) service from ξ_λ when biding β_λ. If i bids $(0, \dots, 0, \beta_1 + \dots + \beta_j, 0, \dots, 0)$, where $\beta_1 + \dots + \beta_j$ stands in the jth entry of the vector, then i will get the service at rate ρ_j.

SP: This follows from Lemma 2.

BB: If $\xi_1, \dots, \xi_\ell$ are all budget balanced, then this follows from construction. Summing the costs $\xi_\lambda(q_{\leq\lambda}, i)$ over all users i pays exactly the costs for the λth layer. In the case of approximate budget balanced cost sharing functions, the largest α_λ dominates.

LC: Fulfilled by construction. ∎

Remark 1 *If $\xi_1, \dots, \xi_\ell$ are polynomial time computable, so is L.*

2.2 Split Session Paradigm

For the split paradigm, there is no analogue of Theorem 1. This even holds for only two players and only two rates. The reason for this is that when a user switches from one rate to another under the split session paradigm, he does not contribute to the costs of the old rate anymore.

Define a cost sharing function ξ by $\xi(c_1, 1) = \xi(c_2, 2) = 1$, $\xi(c_1, 2) = \xi(c_2, 1) = 0$, and $\xi(c_{1,2}, 1) = \xi(c_{1,2}, 2) = 0.7$. (Here c_i denotes the characteristic function of i and $c_{i,j}$ of $\{i, j\}$.) In other words, when one user gets the service, he has to pay 1 and if both user get it, each one has to pay 0.7. ξ is obviously cross-monotonic. We set $\xi_1 := \xi$ and $\xi_2 := \xi$. Define the costs for serving one user to be 1 and for serving both user to be 1.4. In this way, ξ_1 and ξ_2 become budget balance cost sharing functions.

Assume that for this example, there is a mechanism S that meets NPT, VP, CS, SP, BB, and SSC.

Claim 1: If user i bids $(1 + \epsilon, 0)$ or $(0, 1 + \epsilon)$ for any constant $\epsilon > 0$ then he gets the service at rate 1 or 2, respectively.

We only show Claim 1 for rate 1, the other case is symmetric. By CS, user i has a bid (b_1, b_2) such that i will always get the service a rate 1. Assume that the other user always bids $(0, 0)$, then he always gets rate 0 by VP, BB, and SSC. We may assume that $b_2 = 0$. If not, then consider user i having true utilities $(b_1, 0)$. He would not get the service at rate 1 if bidding $(b_1, 0)$ but if bidding (b_1, b_2), contradicting SP. Next, assume to the contrary, that the smallest b_1 such that i gets the service when bidding $(b_1, 0)$ would be $1+\epsilon$ for some $\epsilon > 0$. Assume that the true utilities of i are $(1+\epsilon/2, 0)$. When bidding his true utilities, he would get the service at rate 0 (i.e., not get it at all) by the definition of ϵ. If he however bids $(1 + \epsilon, 0)$, he gets the service. By BB and SSC, his welfare is $1 + \epsilon/2 - 1 = \epsilon/2 > 0$. This violates SP. Thus the claim is proven.

Claim 2: If user 1 bids $(1.1, 1.2)$ and user 2 bids $(1.2, 0)$, then rate 1 is assigned to both users.

Assume that the above bids are the true utilities of both users. By Claim 1, user 2 gets rate 1. If user 1 would get rate 2, then his benefit would be $1.2 - 1 = 0.2$. If he bids $(1.1, 0)$ instead, then he also gets rate 1 by Claim 1. In this case, his welfare is $1.1 - 0.7 = 0.4$. This violates SP.

Claim 3: If user 1 bids $(1.1, 1.2)$ and user 2 bids $(1.2, 1.1)$ then both user get the same rate $r \in \{1, 2\}$.

Assume that this is not the case. Let the above bids be the true utilities of the users. Then the individual welfare of each user is at most $1.2 - 1 = 0.2$. If user 2 would bid $(1.2, 0)$ instead, then by Claim 2, his benefit is $1.2 - 0.7 = 0.5$. This violates SP.

Consider the situation in Claim 3. Assume that the bids are the true utilities of the users. W.l.o.g. assume that the assigned rate is 2. (The other case follows symmetrically, we only need a symmetric version of Claim 2.) Then the benefit for user 2 is $1.1 - 0.7 = 0.4$. If on the other hand, he just bids $(1.2, 0)$ instead, both users get rate 1 instead. Here the welfare of user 2 is $1.2 - 0.7 = 0.5$. This contradicts SP.

Altogether, this proves the following result.

Theorem 2 *There are cross-monotonic functions ξ_1, ξ_2 such that there is no mechanism for ξ_1, ξ_2 that meets NPT, VP, CS, SP, BB, and SSC.* ■

3 Applications to Multicast Pricing

If there is a fixed multicast tree and users selected are served along the minimum subtree including the selected users, then one can take for instance the Shapley value for each layer (see e.g. [4]) and plug it into our meta mechanism. If there is no predetermined multicast tree given in advance, the mechanism has to choose the multicast tree by itself. Let G be a complete edge weighted graph with node set V such that the weights fulfill the triangle inequality. Jain and Vazirani [6] design (a class of) cross-monotonic cost sharing functions ξ_{JV} which are budget balanced and computable in polynomial time. Exploiting Moulin's mechanism, [7], this also gives mechanisms meeting NPT, VP, CS, GSP, and BB. When computing $\xi_{JV}(q, i)$, one gets in addition a multicast tree for serving the users according to the characteristic function q. Furthermore, the weight of the computed tree is at most twice the weight of an optimum Steiner tree for the selected users. If G is not complete, but the distance between two nodes is given by the shortest path metric, then the function ξ_{JV} is still 2-approximate budget balanced.

Now we can plug ℓ copies of ξ_{JV} into our mechanism L. This yields the following result for the multicast pricing problem with rates under the layered paradigm.

Theorem 3 *Let G be a complete edge weighted graph such that the weights fulfill the triangle inequality. There is a mechanism for the multicast pricing problem with rates under the layered paradigm meeting NPT, VP, CS, SP, BB, and LC (with respect to ξ_{JV}). This mechanism is polynomial time computable. For each layer, we can compute a corresponding multicast tree in polynomial time whose weight is at most twice the weight of an optimum Steiner tree for the served users.* ■

Note that the multicast trees for each layer are computed independently of each other. In particular, trees of higher layers are not neccessarily a subtree of the trees of the lower layers. Furthermore, we can have different edge weights for each layer, since ξ_{JV} of course depends on the weights.

Remark 2 *If G is not complete, but we have a shortest path metric, then the above theorem holds for 2-BB instead of BB.*

3.1 Constrained Forest Problems

We extend the ideas of Jain and Vazirani by incorporating the approximation algorithm of Goemans and Williamson [5] for constrained forest problems. Again we are given an edge weighted graph G with node set V such that the edge weights fulfill the triangle inequality.

Given a function $f : 2^V \to \{0,1\}$, we consider the following linear program:

$$\begin{array}{lll} \text{Minimize} & \sum_{e\in E} c_e x_e & \\ \text{subject to} & x(\delta(S)) \geq f(S), & \emptyset \neq S \subsetneq V, \\ & x_e \in \{0,1\}, & e \in E, \end{array} \tag{8}$$

where $\delta(S)$ denotes the set of all edges that have one node in S and the other in $V \setminus S$ and for every $E' \subseteq E$, $x(E')$ shorthands $\sum_{e\in E'} x_e$. The minimal solutions of (8) are constrained forests. The function f is assumed to be *proper*, as defined in [5]: A function f is proper, if for all $S \subseteq V$, $f(S) = f(V \setminus S)$ and if $f(A) = f(B) = 0$ implies $f(A\cup B) = 0$ for all disjoint $A, B \subseteq V$. Proper functions can model interesting multicast issues:

Simultaneous Multicast: Here we have groups of users N_j, $1 \leq j \leq p$, and each user of group N_j shall be served by provider r_j. Here we simply set $f(S) = 1$ iff there is a j such that $\emptyset \neq S \cap (N_j \cup \{r_j\}) \neq N_j \cup \{r_j\}$. This is merely the generalized Steiner tree problem.

Several Servers: Here a group of users N is served by several servers $R = \{r_1, \dots, r_p\}$ and each user needs only be served by any one of them. Here we set $f(S) = 1$ iff $N \cap S = \emptyset$ and $R \subseteq S$ or $N \cap S \neq \emptyset$ and $R \cap S = \emptyset$.

To solve (8), we relax the integrality constraints on the x_e to $x_e \geq 0$. The corresponding dual is

$$\begin{array}{lll} \text{Maximize} & \sum_{\emptyset\neq S\subsetneq V} f(S)\cdot y_S & \\ \text{subject to} & \sum_{S:e\in\delta(S)} y_S \leq c_e, & e \in E, \\ & y_S \geq 0, & \emptyset \neq S \subsetneq V. \end{array} \tag{9}$$

The algorithm of Goemans and Williamson [5] starts with each node as a single component. For conceptual reasons, we associate a notion of time with the algorithm: In one unit of time, the duals are raised one unit amount. The algorithm uniformly raises the dual variables corresponding to the components until some edge goes tight and includes this edge into the forest and hence starts growing larger and larger components. (For algorithmic purposes, we may assume that the time steps are discrete, since we always can determine at which time the next edge will go tight.)

Let $Q \subseteq N$ be given and let q be the characteristic function of Q. Assume we want to serve the users from Q. We run the algorithm of Goemans and Williamson, where the set of users N is replaced by Q. (But we do not remove nodes from V.) The cost sharing function is parameterized with a set of functions $g_i : \mathbb{R}_+ \to \mathbb{R}_+$, one for each user i, which gives the provider a way to adapt the pricing mechanism. (The simplest choice is to choose each g_i as the identity.) For an user $i \in Q$, let $T(i)$ be the first time when i is included in a component C with $f(C) = 0$. For time $t \leq T(i)$, let $C_i(t)$ denote the component i belongs to. The cost share $\xi_{GW}(q, i)$ for user i when users in Q are served is defined as

$$\xi_{GW}(q,i) = 2 \cdot \int_0^{T(i)} \frac{g_i(t)}{G_i(t)} dt \tag{10}$$

where $G_i(t) = \sum_{j \in C_i(t) \cap Q} g_j(t)$. If the g_i are "simple" functions, then the above integral can be computed exactly. Otherwise, one has to work with approximations. Since the duals are raised proportional to time, we simply distribute the duals among the users according to the g_i. Similarly to [6], we can now show:

Theorem 4 *The cost sharing function ξ_{GW} is cross-monotonic.*

Proof. Consider a set of users $Q' \supsetneq Q$ and let q' be its characteristic function. Let $i \in Q$ be an arbitrary user. Define $C'_i(t)$, $G'_i(t)$, and $T'(i)$ with respect to Q' in the same way as $C_i(u)$, $G_i(t)$, and $T(i)$ is defined above for Q. Then

$$\xi_{GW}(q,i) = 2 \cdot \int_0^{T(i)} \frac{g_i(t)}{G_i(t)} dt \quad \text{and} \quad \xi_{GW}(q',i) = 2 \cdot \int_0^{T'(i)} \frac{g_i(t)}{G'_i(t)} dt.$$

If i can reach a node from $Q' \setminus Q$ at some time step, then $C'_i(t) \supsetneq C_i(t)$, otherwise we have $C'_i(t) = C_i(t)$. Thus $G'_i(t) \geq G_i(t)$ for all t. A close inspection of the algorithm by Goemans and Williamson shows that $T'(i) \leq T(i)$ holds, too. Therefore, $\xi_{GW}(q,i) \geq \xi_{GW}(q',i)$. ■

Now we can use the mechanism of Moulin [7] to obtain a mechanism for (binary) constrained forest problems which meets NPT, VP, CS, and GSP. We charge the users twice the values of the dual variables. Goemans and Williamson show that

$$\sum_{e \in F} c_e \leq (2 - 2/|N|) \cdot \sum_{S \subsetneq V} y_S,$$

where F denotes the computed forest. Thus we charge at most twice the optimum costs. Thus the mechanisms is also 2-BB. Moreover, the costs of the underlying multicast structure is within twice of the costs of an optimum constrained forest.

3.2 Mechanisms for Constrained Forest Problems with Rates

In the same way we used our mechanism L to extend the approach of Jain and Vazirani to problems with rates, we can extend the function ξ_{GW} to handle multiple rates. We simply plug ℓ copies of ξ_{GW} (which of course depends on the proper function f and the weights of the graph for each layer) into L.

Theorem 5 *Let G be a complete edge weighted graph such that the weights fulfill the triangle inequality and let f be a proper function. There is a mechanism for the constrained forest problem with rates under the layered paradigm meeting NPT, VP, CS, SP, 2-BB, and LS (with respect to ξ_{GW}). This mechanism is polynomial time computable. For each layer, we can compute a corresponding constrained forest in polynomial time whose weight is at most twice the weight of an optimum constrained forest with respect to the users served of this layer.* ■

In particular, we can solve the simultaneous multicast problem with rates as well as the several server problem with rates under the layered paradigm.

References

[1] Micah Adler and Dan Rubenstein. Pricing multicasting in more practical network models. In *Proc. 13th Ann. ACM–SIAM Symp. on Discrete Algorithms (SODA)*, pages 981–990. ACM–SIAM, 2002.

[2] Joan Feigenbaum, Arvind Krishnamurthy, Rahul Sami, and Scott Shenker. Approximation and collusion in multicast cost sharing. In *Proc. 3rd Ann. ACM Conf. on Electronic Commerce (EC)*. ACM, 2001.

[3] Joan Feigenbaum, Arvind Krishnamurthy, Rahul Sami, and Scott Shenker. Hardness results for multicast cost sharing. In *Proc. 22nd conf. on Found. of Software Technology and Theoret. Comput. Sci. (FSTTCS)*, volume 2556 of *Lecture Notes in Comput. Sci.*, pages 133–144. Springer, 2002.

[4] Joan Feigenbaum, Christos Papadimitriou, and Scott Shenker. Sharing the cost of multicast transmissions. In *Proc. 31st Ann. ACM Symp. on Theory of Comput. (STOC)*, pages 218–227. ACM, 2000.

[5] Michel X. Goemans and David P. Williamson. A general approximation technique for constrained forest problems. *Siam J. Comput*, 24(2):296–317, 1995.

[6] Kamal Jain and Vijay V. Vazirani. Applications of approximations algorithms to cooperative games. In *Proc. 33rd Ann. ACM Symp. on Theory of Comput. (STOC)*, pages 364–372. ACM, 2001.

[7] Hervé Moulin. Incremental cost sharing; characterization by strategyproofness. *Social Choice and Welfare*, 16:279–320, 1999.

[8] Hervé Moulin and Scott Shenker. Strategyproof sharing of submodular costs: budget balance versus efficency. *Economic Theory*, 18:511–533, 2001.

[9] Noam Nisan. Algorithms for selfish agents. In *Proc. 16th Int. Symp. on Theoret. Aspects of Comput. Sci. (STACS)*, volume 1563 of *Lecture Notes in Comput. Sci.*, pages 1–15. Springer, 1999.

[10] Noam Nisan and Amir Ronen. Algorithmic mechanism design. In *Proc. 31st Ann. ACM Symp. on Theory of Comput. (STOC)*, pages 129–140. ACM, 1999.