Approximation Algorithms for Budget-Constrained Auctions

Rahul Garg, Vijay Kumar*, and Vinayaka Pandit

IBM India Research Lab
IIT Delhi, New Delhi - 110016, India
email: {grahul, pvinayak}@in.ibm.com

Abstract. Recently there has been a surge of interest in auctions research triggered on the one hand by auctions of bandwidth and other public assets and on the other by the popularity of Internet auctions and the possibility of new auction formats enabled by e-commerce. Simultaneous auction of items is a popular auction format. We consider the problem of maximizing total revenue in the simultaneous auction of a set of items where the bidders have individual budget constraints. Each bidder is permitted to bid on all the items of his choice and specifies his budget constraint to the auctioneer, who must select bids to maximize the revenue while ensuring that no budget constraints are violated. We show that the problem of maximizing revenue is such a setting is NP-hard, and present a factor-1.62 approximation algorithm for it. We formulate the problem as an integer program and solve a linear relaxation to obtain a fractional optimal solution, which is then deterministically rounded to obtain an integer solution. We argue that the loss in revenue incurred by the rounding procedure is bounded by a factor of 1.62.

Keywords: auctions, winner determination, approximation algorithm, rounding.

1 Introduction

In recent years, there has been a great deal of interest in the research community in various formats and methodologies for auctions [2, 7, 8]. This has been driven in part by the widespread popularity of Internet auctions. Another major factor in the revival of interest in the theory of auctions has been the wave of privatization of public property in many parts of the world, which has involved the design and use of a variety of auction formats. Such auctions have ranged from the sale of spectrum rights in several countries [10] to the public assets in the former Soviet Union. Simultaneous auctions are also popular in electricity markets and equities trading [3, 4]. Issues of interest include both the design

* Vijay Kumar can be reached at amazon.com, 605 5th Avenue South, Seattle, WA 98104, USA. Email:vijayk@amazon.com

of suitable auction formats as well as optimization problems related to winner determination.

In many scenarios, such as that of the auction of public assets, the simultaneous auction of several assets is likely to impose financial or liquidity constraints on the bidders. For instance, in the simultaneous auction of the expensive spectrum rights of several territories, bidders may be constrained in matching rivals' bids by the limitations of total resources available [1]. For efficient price discovery, it would be helpful to insulate the bidders against the risk of their winning bids exceeding their financial capacity.

One format that answers these concerns is a *budget-constrained auction.* In a budget-constrained auction, several items are simultaneously auctioned in sealed-bid auctions. Each bidder submits a collection of bids and informs the auctioneer of his aggregate financial capacity or *budget.* The auctioneer then allocates the objects to the bidders, charging each winner a price no higher than his bid for the item won, and making sure that the total charge for each bidder is within his specified budget. The auctioneer may for some item choose to charge a price lower than the corresponding bid in view of the winning bidder's budget constraint.

The problem of auction when bidders have budget constraints has been studied in several different contexts. Rothkopf [6] discusses how the computation of best responses is affected by budget constraints. J-P Benoit and Vijay Krishna [1] discuss ordering bids in sequential auctions when bidders have budget constraints. Palfrey [5] has studied the effects of budget constraints in a multiple-object setting with complete information. Both [1] and [5] are game theoretic in treatment, and try to prevent bidders from bringing down valuations in the face of complete information.

In this paper we consider the problem of winner determination in a budget-constrained auction, which we term the *Budget Constrained Auction Problem (BCAP).* We show this problem to be NP-hard and present a polynomial-time 1.62-approximation for it. Our approach involves formulating the problem as an integer program, solving a linear relaxation, and deterministically rounding the resulting optimal fractional solution to derive a feasible integer solution. We argue that the feasible integer solution so obtained is within a factor of 1.62 of the optimal fractional solution.

More formally, *the budget constrained auction problem (BCAP)* with N bidders and M items can be formulated by the following integer program (IP1).

$$\max \sum_{i=1}^{N} \left(\sum_{j=1}^{M} b_{i,j} x_{i,j} - d_i \right) \tag{1}$$

subject to

$$\sum_{i=1}^{N} x_{i,j} \leq Q_j \;\; j \in [1, M] \tag{2}$$

$$\sum_{j=1}^{M} b_{i,j}x_{i,j} - d_i \leq D_i \; i \in [1, N] \tag{3}$$

$$x_{i,j} \in [0, q_{i,j}] \tag{4}$$

Q_j is an integer representing the available stock of item j, $b_{i,j}$ is the price per item bid by the bidder i on item j, $q_{i,j}$ is an integer representing the maximum quantity of item j that bidder i wants to buy, and D_i is the budget of bidder i. In any solution, the integer variables $x_{i,j}$ represent the number of items of type j assigned to bidder i. The variable d_i represents the *discount* given to bidder i – that is, the difference between the sum of the winning bids of bidder i and his budget (recall that the auctioneer may choose to charge a bidder less than the sum of his winning bids). The objective function represents the total revenue which is to be maximized. (2) represents the *stock constraint* of the items, (3) represents the *budget constraint* of the bidders, and (4) represents the *bid constraint.*

The structure of the integer program under consideration is similar to that of the *generalized assignment problem* [9]. Consider the variation of BCAP in which an item can be assigned to a bidder only at his bid value. In the special case where all the Q_j values are unity, this problem reduces directly to the generalized assignment problem, for which a factor-2 approximation is presented in [9]. It is possible to obtain such a reduction (and approximation) in the general case as well. Ostensibly, the flexibility provided by allowing the auctioneer to accept certain bids at below the quoted value permits an improved approximation factor of 1.62 (this may be an interesting fact from the point of view of its implications for the many scheduling problems [9] that have the structure of the generalized assignment problem).

1.1 Paper Outline

In Section 2 we present the basic framework of our approach. Section 3 contains the details of various procedures used to round a fractional solution into a integral one. The algorithm is analysed in Section 4. A proof of the NP-hardness of the problem under consideration is contained in Appendix A.

2 Our Algorithm

Our algorithm is based upon deterministic rounding of an optimal fractional solution. We begin by solving a linear program to obtain an optimal fractional solution. In this solution we make certain transformations or modificatons which, without reducing the value of the objective function, give us what we call a *simple* solution, which is a fractional solution with certain properties. Next we make a series of modifications to this simple solution, to obtain a succession of simple solutions each of which contains fewer fractional values than the previous. At the end of this process we are left with an integral solution. We show that the value of this integral solution is within a certain constant factor of the value of the initial simple solution.

Our linear program $LP1$ is obtained from $IP1$ by substituting the bid constraint (4) by the following *relaxed bid constraint*:

$$0 \leq x_{i,j} \leq q_{i,j} \tag{5}$$

In order to describe our algorithm, we need the following definitions. In the context of a solution $A = x^A_{i,j}$ of LP1, we define the *unsold stock* of item j as $Q_j - \sum_{i=1}^{N} x^A_{i,j}$, the *sold quantity* of item j as $\sum_{i=1}^{N} x^A_{i,j}$, and the *amount spent* by bidder i as $\sum_{j=1}^{M} b_{i,j} x^A_{i,j}$. Item j is said to be *sold out* if its unsold stock is zero, *partially sold* if its unsold stock and sold quantity are both positive, and *unsold* if its sold quantity is zero. We say that bidder i is *unsatisfied* if the amount spent by him is less than the maximum permitted by his budget constraint and discount (that is, $\sum_{j=1}^{M} b_{i,j} x^A_{i,j} < D_i + d_i$ — incidentally, in our algorithm either this inequality is tight or $d_i = 0$). Define the *residual graph* $R(A)$ corresponding to a solution A as a bipartite graph with N vertices $(u_i, i \in [1, N])$ on the bidder side and M vertices $(v_j, i \in [1, M])$ on the item side. The edge (u_i, v_j) is present in $R(A)$ iff the value of $x^A_{i,j}$ is not an integer.

Given a path $P = (u_1, v_1, \ldots, u_k, v_k)$ in $R(A)$, let

$$\begin{aligned} e_1 &= 1 \\ e_{l+1} &= e_l \tfrac{b_{l+1,l}}{b_{l,l}} \text{ for } l \in [1, k-1] \\ b^P_{1,k} &= \tfrac{b_{k,k}}{e_k} \\ &= b_{k,k} \prod_{l=2}^{k} \left(\tfrac{b_{l-1,l-1}}{b_{l,l-1}}\right) \end{aligned}$$

We call $b^P_{1,k}$ the *effective bid* for item k by bidder 1 along the path P.

Let $P = (u_j, \ldots, v_k, u_l)$ be a path joining bidder vertices u_j and u_l. Define the *exchange ratio* between bidder j and l along the path P as $e(P) = \frac{b_{l,k}}{b^P_{j,k}}$.

We define a solution B of LP1 to be a *simple* solution if $R(B)$ contains

- no cycle,
- no path from a partially sold item to another partially sold item, and
- no path P from an unsatisfied bidder to another bidder such that $e(P) \leq 1$.

The last property implies that $R(B)$ can not contain a path P from an unsatisfied bidder to another unsatisfied bidder.

In the case of a simple solution A, the residual graph $R(A)$ has no cycles. Therefore the path joining any two bidder vertices is unique. Let $P = (u_j, \ldots, v_k, u_l)$ be the unique path joining bidder vertices u_j and u_l. Denote the exchange ratio between bidder j and l as $e(j, l) = e(P) = \frac{b_{l,k}}{b^Q_{j,k}}$, where $Q = (u_j, \ldots, v_k)$. Note that if u_i, u_j and u_k are connected bidder vertices, then $e(i, k) = e(i, j)e(j, k)$.

Theorem 1. *From a feasible solution A of LP1, a simple solution B of same or higher value can be constructed in polynomial time.*

In Section 3, we prove Theorem 1 by presenting a polynomial-time algorithm *simplify(A)* that, given a feasible solution A, constructs and returns a simple solution of equal or greater value.

Our algorithm can now be presented. It is described in Figure 1. The algorithm first obtains an optimal fractional solution of LP1, which is then converted into a simple solution. The algorithm then proceeds by selecting the bidder vertices by examining maximal paths in the residual graph and modifying x_{ij}'s corresponding to selected vertices. Each selected vertex is marked. The algorithm terminates when there are no edges left in the residual graph. The detail algorithm is presented in Figure 2, and its correctness and the performance are analysed is Section 4.

algorithm budget_constrained_auction
begin
    Let A be an optimal simple solution
s1: *while* there are edges in $R(A)$
        Let P be a maximal path in $R(A)$
        if (P starts with an item vertex)
            $P = reverse(P)$
        else if (both ends of P are bidder vertices) AND ($e(P) > 1$)
            $P = reverse(P)$
        endif
        Let u_{i_1} and v_{j_1} be the first two vertices in P
        if ($b_{i_1 j_1} x_{i_1 j_1} \leq pD_{i_1}$), where p is a chosen constant
s2.1   mark (u_{i_1})
s2.2   Obtain solution A' from A by setting $x_{i_1 j_1}$ to 0
s2.3 $A = simplify(A')$;
        else
s3.1   Modify A by setting d_{i_1} to $(1-p)D_{i_1}$
s3.2 $A = simplify(A)$;
s3.3   Modify A by setting d_{i_1} to $\sum_{j=1}^{M} b_{i_1,j} x_{i_1,j}^{A} - D_{i_1}$
        end if
    end while
end

Fig. 1. Our Algorithm

3 Obtaining Simple Solutions

Given a feasible solution, the procedure *simplify* converts it into a simple solution by repeated application of four simple transformations, which involve limited local redestribution of money and items among a set of bidders. These four transformations are described below.

3.1 Indirect Purchase

Consider a solution A that has a path P connecting an unsatisfied bidder to an unsold or partially sold item. WLOG, let the vertices of $R(A)$ be numbered so that $P = (u_1, v_1, \ldots, u_k, v_k)$.

Given such a path, an *indirect purchase* is the following transaction. Bidder 1, who has not yet spent all his budget, spends an additional amount δ to buy another small quantity of item 1. Item 1 may be sold out, in which case the required amount of item 1 is bought back from bidder 2. This leaves some spare money with bidder 2, which he spends to buy some more quantity of item 2, and so forth. This process terminates when we reach item k – an item which is not sold out. If δ is kept small enough, all the budget constraints, stock constraints and relaxed bid constraints are still satisfied and the solution so obtained is of higher value.

More formally, define

$$\begin{aligned} e_1 &= 1 \\ e_{l+1} &= e_l \tfrac{b_{l+1,l}}{b_{l,l}} \text{ for } l \in [1, k-1] \end{aligned} \tag{6}$$

From A, we obtain another solution B as follows.

$$x^B_{l,l} = x^A_{l,l} + \delta \frac{e_l}{b_{l,l}} \qquad 1 \le l \le k \tag{7}$$

$$x^B_{l+1,l} = x^A_{l+1,l} - \delta \frac{e_l}{b_{l,l}} \qquad 1 \le l \le k-1 \tag{8}$$

where

$$\delta = \min \begin{cases} \min_{l=1}^{k} (\lceil x^A_{l,l} \rceil - x^A_{l,l}) \frac{b_{l,l}}{e_l} \\ \min_{j=1}^{k-1} (x^A_{l+1,l} - \lfloor x^A_{l+1,l} \rfloor) \frac{b_{l,l}}{e_l} \\ D_1 - \sum_{j=1}^{M} b_{1,j} x^A_{1,j} \\ (Q_k - \sum_{i=1}^{N} x^A_{i,k}) \frac{b_{k,k}}{e_k} \end{cases} \tag{9}$$

For all other i, j,

$$x^B_{i,j} = x^A_{i,j}. \tag{10}$$

We refer to the perturbation of the solution A along a path P as given by (7 - 10) as an *indirect purchase* of item k by bidder 1 along the path P. Note that the value of δ is positive and non-zero, since the bidder 1 is unsatisfied, item k is not sold out and all the edges of the form $(u_l, v_l), l \in [1, k]$ and $(u_{l+1}, v_l), l \in [1, k-1]$ are present in $R(A)$.

It is easy to verify that

Lemma 1. *As a result of an indirect purchase along a path $P = (u_1, v_1, \ldots, u_k, v_k)$, bidder 1 spends more money and the sold quantity of item k increases, while the expenses of all other bidders and the sold quantities of all other items remain unchanged; and the resulting solution is a feasible solution.* ∎

In particular, bidder 1 spends δ more money and the sold quantity of item k increases by $\frac{\delta}{b^P_{1,k}}$.

3.2 Cycle Elimination

Consider a situation where the residual graph $R(A)$ corresponding to the current solution A has a cycle C. Let v_k, u_i and v_j be three consecutive vertices in C. Let P be the longer path from u_i to v_k and Q the longer path from u_i to v_j along the cycle C. We observe that $b^P_{i,k} = \frac{b_{i,j} b_{i,k}}{b^Q i,j}$, which implies that either $b^P_{i,k} \geq b_{i,k}$ or $b^Q i, j > b_{i,j}$.

We call the following procedure a *cycle elimination*: if $b^Q_{i,j} \geq b_{i,j}$ then bidder i reduces by quantity ϵ his purchase of item j using his direct bid $b_{i,j}$ and uses all the money saved to indirectly purchase the same item along the path Q. Otherwise, bidder i reduces his direct purchase of item k by quantity ϵ and makes up for this reduction by an indirect purchase of item k along path P.

In either case, the expenses of all the bidders remain the same as in A and therefore the objective function value remains unchanged. The sold quantities of items j and k either remain the same or decrease. The sold quantity of all other items remain the same. By making ϵ large enough, we ensure that the transaction removes at least one edge from the residual graph.

3.3 Item Exchanges

Let $P = (v_i, u_j, \ldots, , u_k, v_l)$ be a path from a partially sold item i to another partially sold item l. Let $Q = (u_j, \ldots, u_k, v_l)$ and $T = (u_k, \ldots, u_j, v_i)$. It is easily shown that $b^Q_{j,l} b^T_{k,i} = b_{j,i} b_{k,l}$, and it follows that either $b^Q_{j,l} \geq b_{j,i}$ or $b^T_{k,i} > b_{k,l}$. An *item exchange along the path* P is the following transaction: if $b^Q_{j,l} \geq b_{j,i}$ then bidder j reduces his purchase (using bid $b_{j,i}$) of item i and uses the saved money in an indirect purchase of item l along the path Q. Otherwise bidder k reduces his purchase of item l and instead indirectly purchases item i along the path T. The quantities involved can be chosen in such a way that either one edge in P is eliminated from the residual graph or one out of items i and l gets sold out. The amount of money spent by any bidder remains unchanged.

3.4 Bidder Exchanges

Let $P = (u_i, v_j, \ldots, v_k, u_l)$ be a path from unsatisfied bidder i to some bidder l such that $e(P) \leq 1$. Let $Q = (u_i, v_j, \ldots, v_k)$. We define a *bidder exchange along the path* P to be the following transaction: bidder l reduces by a small quantity ϵ his purchase of item k, and this quantity is bought by bidder i in an indirect purchase along the path Q. The quantities involved can be chosen in such a way that either one edge in P is eliminated from the residual graph or bidder i becomes satisfied. The sold quantity of every item remains unchanged. The total amount spent by all the bidders does not decrease, since $e(P) < 1 \Rightarrow b^Q_{i,k} \geq b_{l,k}$.

A proof of Theorem 1 is now straightforward: repeated application of cycle eliminations, item exchanges and bidder exchanges converts A into a simple solution B. As noted above, these procedures do not decrease the value of the objective function. ■

Having thus obtained a simple solution, the procedure *simplify* proceeds to carry out as many indirect purchases as may be possible, thus possibly increasing the value of the objective function. Note that *simplify* terminates in polynomial time since each cycle elimination, item exchange, bidder exchange or indirect purchase reduces the number of edges in the residual graph.

4 Correctness and Performance

In the following, we compare the value of the integral solution obtained by our algorithm to that of the optimal solution of $LP1$. Consider the situation at any particular time during the execution of the algorithm. Let A be the current solution. We define the potential function $f(A)$ corresponding to solution A as:

$$f(A) = \frac{1}{1-p} \sum_{i \in S} \sum_{j=1}^{M} b_{ij} x_{ij} + \sum_{i \notin S} \left(\sum_{j=1}^{M} b_{ij} x_{ij} - d_i \right) \tag{11}$$

where $S = \{i \mid u_i \text{ is marked}\}$. The constant p is chosen such that $\frac{1}{1-p} = 2 - p$, and thus the selected value is $\frac{3-\sqrt{5}}{2}$.

Let OPT denote the value of the optimal solution of $LP1$. We will show that the algorithm maintains the following invariants at the entry of the outer loop.

Invariant 1 *At least one end point of every maximal path in $R(A)$ is a satisfied bidder vertex.*

Invariant 2 $f(A) \geq OPT$.

We begin by establishing some properties of the solutions we work with.

Lemma 2. *Let A be an optimal solution of LP1. There is no path in $R(A)$ joining an unsatisfied bidder to an unsold or partially sold item.*

Proof. If there were such a path P, then it follows from Lemma 1 that it is possible to obtain another feasible solution which increases the total revenue. ■

Lemma 3. *Every sold-out item vertex in $R(A)$ either has no edge or at least two incident edges.*

Proof. Assume item j is sold-out, i.e. $\sum_{i=1}^{N} x_{ij} = Q_j$. Since Q_j is an integer, either none or at least two of the x_{ij} values must be fractional. Therefore either there is no incident edge, or there are two or more incident edges on v_j in $R(A)$. ■

Lemma 4. *One of the endpoints of every maximal path in the residual graph of an optimal simple solution is a satisfied bidder vertex.*

Proof. Let P be any maximal path in $R(A)$ for which this is not true. Lemma 3 implies that neither endpoint of P is a sold-out item vertex. Thus P must connect two partially sold items, two unsatisfied bidders or a partially sold item to an unsatisfied bidder. The first two possibilities are ruled out since A is a simple solution, while the last is ruled out by Lemma 2. ■

Theorem 2. *The algorithm maintains Invariants 1 and 2 at the time of every visit to step* **s1**.

Proof. The proof is by induction. We show that the invariants are true at the time time of the first visit to step **s1**, and we show that they are preserved from one visit to the next.

Lemma 4 implies that Invariant 1 is true when step **s1** is first visited. Since there are no marked vertices at this time, the Invariant 2 is also true.

Next, we show that the two invariants are preserved from one visit to step **s1** to the subsequent one. That is, the execution of the while loop beginning at step **s1** preserves the invariants.

The algorithm proceeds by selecting a maximal path P in the residual graph. From the first invariant, it follows that one end of P must be a satisfied bidder vertex (u_{i_1}). The algorithm orients P appropriately such that u_{i_1} is the first vertex of P. The rest of the body of the loop consists of a conditional statement in which either steps **2.1** − **2.3** or steps **3.1** − **3.3** are executed. We separately analyse the two possibilities resulting from the conditional statement.

Case 1: If the only edge from u_{i_1} contributes less than p times the amount spent by bidder i_1, then the edge is dropped, setting $x_{i_1 j_1}$ to zero, and u_{i_1} is marked. This could lead to a new maximal path ending at vertex v_{j_1}, from either a partially sold item vertex, or from a unsatisfied bidder vertex, either of which would violate Invariant 1. However, observe that paths of either kind are eliminated by the procedure *simplify*, which is executed at step **s2.3**. Invariant 1 is thus restored.

Since $b_{i_1 j_1} x_{i_1 j_1} \leq p \sum_{j=1}^{M} b_{i_1 j_1} x_{i_1 j_1}$, the execution of steps **s2.1** and **s2.2** does not decrease the value of the potential function. Step **s2.3** does not decrease the value of the potential function since it only affects unmarked vertices (note that marked vertices are not present in the residual graph) and causes a non-negative change in the contribution of these vertices to the objective function (this contribution equals the second term of the potential function (11)). Thus Invariant 2 is maintained through the execution of the loop.

Case 2: If the only edge in $R(A)$ incident on u_{i_1} corresponds to an expense of more than pD_{i_1}, then at step **3.1** the discount d_{i_1} available to bidder i_1 is changed to $(1-p)D_{i_1}$. This makes the bidder vertex u_{i_1} unsatisfied and it is possible that Invariant 1 may not hold any more. As before, the invocation of *simplify* (at step **3.2**) removes any paths that may violate Invariant 1. At step **3.3** the discount d_{i_1} is reduced to the smallest quantity required to facilitate the current purchases of bidder i_1. Steps **3.1** and **3.3** do not affect the potential function while step **3.2** does not reduce it. Thus, Invariant 2 is maintained through the execution of steps **3.1** − **3.3**. ■

The algorithm terminates in polynomial time because

Lemma 5. *Every iteration of the outer loop reduces at least one edge from the residual graph.*

Proof. In Case 1 above, the first edge of the path P is removed from $R(A)$. Consider Case 2. After step **3.1**, u_{i_1} is an unsatisfied vertex connected by path P to either a partially sold item or a satisfied bidder. Thus either an indirect purchase or a bidder exchange would be possible when the procedure *simplify* is invoked at step **3.2**, either of which would remove at least one edge from the residual graph. If that operation is not carried out, it must be that either vertex u_{i_1} has been converted into a satisfied vertex, or some edge of path P has been removed, during the execution of *simplify*. In either case, at least one edge has been removed from the residual graph. ■

Theorem 3. *Our algorithm is a polynomial time factor-$\frac{1+\sqrt{5}}{2}$ approximation algorithm for the budget-constrained auction problem.*

Proof. Let A denote the integral solution X obtained by our algorithm and Obj the corresponding value of the objective function. Every bidder i contributes the amount $\sum_{j=1}^{M} b_{ij}x_{ij} - d_i$ to the objective function. Note that for $i \in S$, $d_i = 0$. Since $\frac{1}{1-p} = 2 - p$, we have

$$\frac{Obj}{1-p} = \frac{1}{1-p}\sum_{i\in S}\sum_{j=1}^{M} b_{ij}x_{ij} + (2-p)\sum_{i\notin S}\left(\sum_{j=1}^{M} b_{ij}x_{ij} - d_i\right)$$

$d_i > 0$ for some i only if $D_i < \sum_{j=1}^{M} b_{ij}x_{ij}$. Since $d_i \leq (1-p)D_i < \sum_{j=1}^{M} b_{ij}x_{ij}$,

$$\frac{Obj}{1-p} > \frac{1}{1-p}\sum_{i\in S}\sum_{j=1}^{M} b_{ij}x_{ij} + \sum_{i\notin S}\sum_{j=1}^{M} b_{ij}x_{ij} = f(A)$$

By Invariant 2, $f(A) \geq OPT$. Since $\frac{Obj}{1-p} > f(A)$, it follows that $Obj > (1-p)OPT$, and thus our algorithm is a $\frac{1}{1-p}$-approximation. Substituting $\frac{3-\sqrt{5}}{2}$ for p yields the desired approximation factor. ■

The authors would like to thank David Shmoys, R. Ravi and Naveen Garg for helpful comments and suggestions.

References

1. J-P Benoit and V. Krishna. Multiple-object auctions with budget constrained bidders. A working paper at http://econ.la.psu.edu/ vkrishna/multi10.ps, April 1998.
2. Y. Fujishima, K. Leyton-Brown, and Y. Shoham. Taming the computational complexity of combinatorial auctions: Optimal and approximate approaches. In *International Joint Conferences on Artificial Intelligence*, 1999.
3. R. P. McAfee and J. McMillan. Analyzing the airwaves auctions. *Journal of Economic Perspectives*, 10(1):159–175, 1996.

4. J. McMillan. Selling spectrum rights. *Journal of Economic Perspectives*, 8(3):145–162, 1994.
5. T. R. Palfrey. Multiple object, discriminatory auctions with bidding constraints: A game theoretic analysis. *Management Science*, 26:935–946, 1980.
6. M. Rothkopf. Bidding in simultaneous auctions with a constraint on exposure. *Operations Research*, 25:620–629, 1977.
7. M. H. Rothkopf, A. Pekec, and R. M. Harstad. Computationally manageable combinatorial auctions. *Management Science*, 44:1131–1147, August 1998.
8. T. Sandholm. An algorithm for optimal winner determination in combinatorial auction. Technical Report WUCS-99-01, Washington University, Department of Computer Science, January 1999.
9. D.B. Shmoys and E. Tardos. An approximation algorithm for the generalised assignment problem. *Mathematical Programming A*, 62:461–474, 1993.
10. Martin Spicer. International survey of spectrum assignment for cellular and pcs. Technical report, September 1996. Federal Telecommunications Commission, available at http://www.fcc.gov/wtb/auctions/papers/spicer.html.

A Hardness of BCAP

Using a reduction from the 3-SAT problem, we show that BCAP is NP-hard.

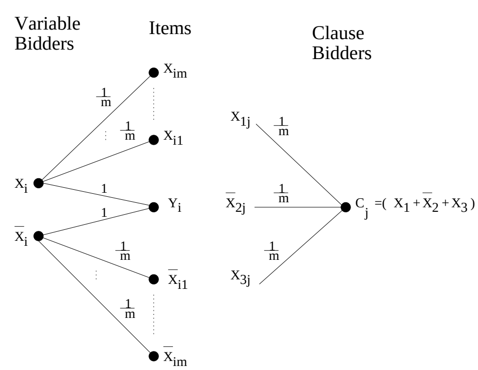

Fig. 2. The reduction

Given an instance I of a 3-SAT problem with n variables and m clauses, obtain an instance I' of BCAP as follows. Corresponding to each variable x_i in I, I' contains

- an item Y_i,
- two bidders, X_i and $\overline{X_i}$, both of which bid for Y_i with bid value 1.
- m items $\{X_{i1}, X_{i2}, \ldots, X_{im}\}$ called the *t-items* of X_i. X_i makes bids of value $\frac{1}{m}$ on each of the t-items of X_i.
- m items $\{\overline{X_{i1}}, \overline{X_{i2}}, \ldots, \overline{X_{im}}\}$ called the *f-items* of X_i. $\overline{X_i}$ makes bids of value $\frac{1}{m}$ on each of the f-items of X_i.

For each clause c_k, I' contains a bidder C_k which makes a bid of value $\frac{1}{m}$ for the object X_{ik} if variable x_i is present in clause c_k, and a bid of value $\frac{1}{m}$ for the object $\overline{X_{ik}}$ if $\overline{x_i}$ is present in clause c_k. Figure 2 shows the basic structure of the reduction for a variable x_i and a clause c_j. This structure is repeated for all variables and clauses of I, and appropriate bids are added. The budget constraint for each bidder of the type X_i or $\overline{X_i}$ is 1, and the budget constraint for each C_k is $\frac{1}{m}$. The maximum revenue possible in I' is $2n+1$ where n is number of variables in I. Clearly I' can be obtained from I in polynomial time.

Lemma 6. *If I is satisfiable then there exists an assignment of items to bidders in I' such that the total revenue is $2n+1$.*

Proof. Let A be asatifying assignment for I. If x_i is true in A then assign Y_i to X_i at price 1, and assign all f-items of X_i to $\overline{X_i}$ at price $\frac{1}{m}$ each. Otherwise assign Y_i to $\overline{X_i}$ at price 1, and assign all t-items of X_i to X_i at price $\frac{1}{m}$ each. For each clause c_j, there exists at least one true literal. Choose one such true literal x_i (or $\overline{x_i}$), and assign the j t-item (f-item in case of a negated literal) of X_i to C_j. It can be easily verified that we can assign exactly one item at price $\frac{1}{m}$ to each clause. ■

Lemma 7. *If there exists an assignment of items to bidders in I' which results in revenue more than $(2n+1-\frac{1}{m}+\epsilon)$ for some $\epsilon > 0$, then I is satisfiable.*

Proof. Obtain a solution to I as follows: for all i, let x_i be true if Y_i has been assigned to X_i, and false otherwise.

As the revenue from all bidders of the type X_i or $\overline{X_i}$ can be no more than $2n$, the revenue from bidders of the type C_j is at least $(1-\frac{1}{m}+\epsilon)$. Similarly, the revenue from each bidder of the type X_i or $\overline{X_i}$ must be at least $(1-\frac{1}{m}+\epsilon)$. Clearly each C_j has been assigned an item at a price greater than zero. Consider any C_j and an item assigned to it. Let us suppose that the clause has been assigned a t-item (respectively f-item) of a variable X_i. Since the income from X_i (respectively $\overline{X_i}$) is at least $(1-\frac{1}{m}+\epsilon)$, Y_i has been assigned to X_i (respectively $\overline{X_i}$). Thus x_i (respectively $\overline{x_i}$), one of the literals in clause c_j is true, and c_j is satisfied.

It follows from Lemmas 6 and 7 and the NP-completeness of 3-SAT that

Theorem 4. *The Budget Constrained Auction Problem is NP-hard and cannot be approximated to within a factor of $1+\frac{1}{2n}$ of the optimal, where n is the number of bidders.*