CMSC 858F: Algorithmic Game Theory
Fall 2010
Frugality & Profit Maximization in Mechanism Design

Instructor: Mohammad T. Hajiaghayi
Scribe: Barna Saha

September 29th, 2010

1 Overview

In this lecture, we start with some of the issues of combinatorial auction and then delve into the notion of *frugality* in auctions and mechanism design. Finally we conclude with a discussion on profit-maximization in mechanism design.

2 Issues in Combinatorial Auctions

Recall from the previous lecture that in combinatorial auction each bidder has an associated real-valued valuation function V defined for each subset of items S. An allocation of items $S_1, S_2, \ldots, S_n$ among the bidders with valuation function $V_1, V_2, \ldots, V_n$ respectively is socially efficient if the allocation maximizes the social welfare $\sum_i V_i(S_i)$.

Combinatorial auction is a very general auction setting and it is well-known that if we use VCG payments, then this is incentive-compatible. However there are some issues.

The two major issues in combinatorial auctions are,

- **Computational Complexity.** As it turns out the allocation problem is NP-hard even for some pretty simple cases. We will show for a very simple kind of bidders known as *single-minded* bidders the allocation problem s not only NP-hard but it is even NP-hard to approximate it within an approximation factor of $n^{1-\epsilon}$ for any constant $\epsilon > 0$.

- **Representation & Communication.** The valuation functions have an exponential sized domain. So even how these functions can be represented is not clear. This issue forces us to look for languages that allow succinct representation of valuations to be used in practice. We call them bidding languages. Here we face the issue of expressiveness vs. simplicity. We want the bidding language to be simple enough for human to express and for programs to work. On the otherhand we want it to be capable of expressing any naturally occurring valuation succinctly.

We now give some examples of bidding languages and then dwell into the hardness proof of allocations for single-minded bidders.

Definition 1 (Atomic Bid) *An* atomic bid *is an offer of* P *for a set of items* S *or any* $T \supseteq S$ *and zero otherwise.*

We can use combination of atomic bids with XOR and OR.

Example 1

1. *If we have* $(\{a,b\},3)\,\mathsf{XOR}\,(\{c,d\},5)$*, then* $V(\{a,c\}) = 0, V(\{a,b,c,d\}) = 5$
2. *If we have* $(\{a,b\},3)\,\mathsf{OR}\,(\{c,d\},5)$*, then* $V(\{a,c\}) = 0, V(\{a,b,c,d\}) = 8$
3. $(\{a,b\},3)\,\mathsf{OR}\,(\{c,d\},5)$*, then* $V(\{a,b,c\}) = 3$ *(since we can satisfy only the first one)*

Combining atomic bids with OR, XOR to obtain more expressive language is a very natural approach. Constructing bidding language is an active area of research (e.g., for Google). Refer to the Section 11.4 of [2] for more details.

Definition 2 (Single-minded Bidder) *A single-minded bidder is a bidder with an atomic bid, i.e., a bidder for which there is a set* $A \subseteq S$ *of goods and a value* $\alpha \geq 0$*, such that*

- $V(T) = \alpha$*, whenever* $T \supseteq A$
- $V(T) = 0$*, otherwise.*

Therefore a single-minded bidder can be represented by a tuple (A, α)*.*

Even for this special case of single-minded bidders, it is not possible to implement the VCG mechanism in polynomial time. As we mentioned earlier the allocation problem is NP-hard even to obtain a reasonable approximation.

Theorem 1 *Given single-minded bidders* $(A_1, \alpha_1), (A_2, \alpha_2), \ldots, (A_n, \alpha_n)$*, granting a set of* disjoint *bids (i.e., a subset of players such that the corresponding* A_i*'s are pairwise disjoint) to maximize the sum* $\sum_i \alpha_i$ *of the values of the granted bids is NP-Hard (indeed* $\Omega(n^{1-\epsilon})$*-hard to approximate for any positive* $\epsilon > 0$*)*

Proof: We give a simple reduction from NP-hard weighted independent set problem. Here input is a graph $G = (V, E)$ and a weight function $w : V \to \mathbb{R}_{\geq 0}$ and the goal is to obtain an independent set of maximum weight. We create an item for each edge of G and a player for each vertex of G. For a vertex/player $v \in V(G)$, set $\alpha_v = w(v)$ and A_v equal to the set of edges of G incident to v. A set of vertices forms an independent set iff the corresponding players can be granted simultaneously and the weight of the independent set is exactly equal to the social welfare of the allocation. Since the weighted independent set is $\Omega(n^{1-\epsilon})$-hard to approximate for any positive $\epsilon > 0$, the claim of hardness follows. ■

3 Frugality in Auctions & Mechanism Design

n this setting the auctioneer is a buyer who wants to purchase goods or services. Agents are sellers who have costs for providing the goods or service. The auctioneer's goal is to maximize the social welfare; but the question is how much he should overpay. *Frugality of mechanism is the amount by which he overpays.*

When the auctioneer was a seller, the goal was to design mechanism to maximize his profit. Here our goal is to design a mechanism to minimize the payments auctioneer makes. Hence, analyzing the frugality of mechanism becomes an important aspect.

Note that for a single good in Vickrey's auction, payment equals the second cheapest price. In general, we might have more complicated system.

Example 2 (Path Auction) *Given a graph (network), the auctioneer wants to buy an s-t path. Each edge is owned by a different agent and has an internal cost to do the transfer (say). The auctioneer will try to buy the shortest path.*

Example 3 (Spanning Tree Auction) *The same setting but the auctioneer wants to buy a spanning tree instead. The auctioneer will try to buy the minimum weight spanning tree.*

Remark. For single-item auctions, in the absence of any prior information about agent's valuations; we can show Vickrey auction is optimal and of course achieves a profit equal to the value of the second highest bidder. Thus, a natural first mechanism to consider in this setting is the VCG auction.

We can use VCG, but the main question is *when VCG or any other incentive compatible mechanism achieves a total payment of at most the second cheapest solution ?* Let us see a few examples.

In Figure 1 the weight of the shortest s-t path is 4. VCG pays for each weighted 1 edge a payment of $11 - (4 - 1) = 8$ and pays $4 - 4 = 0$ for 11-weight edge. Thus in total VCG pays 32, whereas the second cheapest path has a weight of 11. Hence the frugality ratio is $32/11$. Thus comparing to the second option, the payment is high. This ratio can indeed be made arbitrarily bad ($\Omega(n)$ where n is the length of the path). This is a simple generalization of the same example.

Figure 1: An Example of a Path Auction

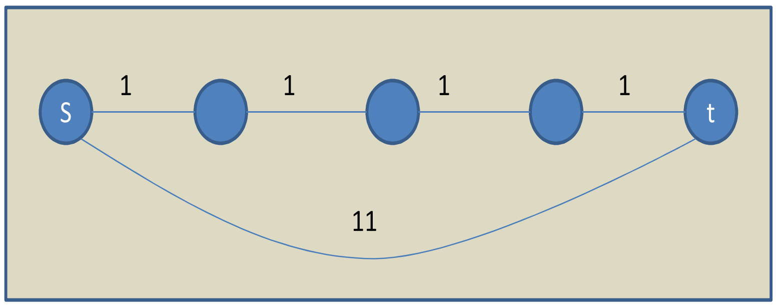
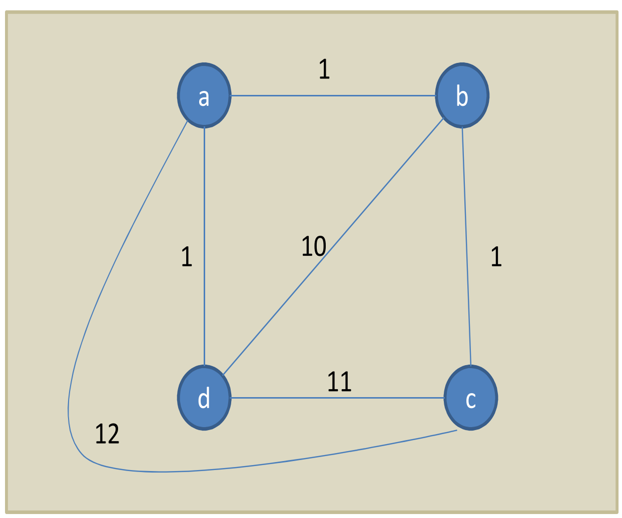


Figure 2: An Example of a Spanning Tree Auction



In Figure 2 minimum spanning tree has a weight of 3. VCG pays $(12-(3-1))$ for the edges (a,d) and (a,b) and pays $(13-(3-1)) = 11$ for (b,c). Thus the total payment is 31. The second minimum spanning tree that is disjoint from the first has a weight of $10+11+12 = 33$. Hence the frugality ratio is $31/33 < 1$.

A frugal mechanism should minimize the total payment. For example, the path auction as we saw was bad, whereas the spanning tree auction was good. But we would like to derive a more general statement.

Before deriving a more general statement regarding the path and the spanning tree auctions and the frugality ratio, let us recall the direct characterization of incentive compatible mechanism:

Direct Characterization of Incentive Compatible Mechanisms:

A mechanism is incentive compatible if and only if,

1. The payment P_i does not depend on V_i, but only on alternative (outcome) chosen $f(V_i, V_{-i})$.
2. The mechanism optimizes for each player. That is for every V_i, we have that $f(V_i, V_{-i}) \in \text{argmax}_a (V_i(a) - P_i)$, where the quantification is over all the alternatives in the range of $f(\cdot, V_{-i})$.

We are now ready to prove the following theorem,

Theorem 2 *For any incentive compatible mechanism M and any graph G with two vertex disjoint s-t paths P and P', there is a valuation profile V such that M pays an $\Omega(\sqrt{|P||P'|})$ factor more than the cost of the second cheapest path.*

Proof: Let $k = |P|$ and $k' = |P'|$. Ignore all the edges not in P or P' by setting their cost to infinity. We define $V_{i,j}$ as follows: The cost of the ith edge of P is $V_i = \frac{1}{\sqrt{k}}$. The cost of the jth edge of P' is $V_j = \frac{1}{\sqrt{k'}}$ and all other edges have cost zero.

Note that M on $V_{i,j}$ must select either all the edges in path P or all the edges in path P' as winner (since we have only two edge disjoint paths P and P' as options). Define directed bipartite graph $G' = (P, P', E')$ on edges in path P and P' as follows. For any pair of vertices (i,j) in the bipartite graph, there is either a directed edge (i,j) in E' saying M on $V_{i,j}$ selecting path P' (called forward edges) or a directed edge (j,i) denoting M on $V_{i,j}$ selecting path P (called backward edges). Note that $|E'| = kk'$. Without loss of generality assume E' has more forward edges and thus at least $\frac{kk'}{2}$ forward edges. Since there are k edges in path P, there must be one vertex i with at least $\frac{k'}{2}$ forward edges. Let $F(i)$ represent neighbors of i in the bipartite graph with $|F(i)| \geq \frac{k'}{2}$. Now consider the valuation profile V_{i0} in which the cost of the ith edge of P is $V_i = \frac{1}{\sqrt{k}}$ and all other edges have cost zero.

By the definition of $F(i)$, for any $j \in F(i)$, M on instance $V_{i,j}$ selects path P'. Since M is incentive compatible, its allocation rule must be monotone, i.e., if agent j is selected when bidding V_j, it must be selected when bidding 0 (it is called weak-monotonicity). Therefore M selects P' on $V_{i,0}$.

In addition, for $j \in F(i)$, the payment should be at least $\frac{1}{\sqrt{k}}$, since when the valuation profile is $V_{i,j}$, the payment should be at least $\frac{1}{\sqrt{k'}}$ (otherwise j will receive negative utility). By the direct characterization (mentioned just before Theorem 2) of incentive compatible mechanisms, we know when other bidders valuations and the outcome are the same, the payment should also be the same. So the payment for j is at least $\frac{1}{\sqrt{k'}}$, when the valuation profile is $V_{i,0}$. Therefore on $V_{i,0}$, the total payment of M is at least $F(i)\frac{1}{\sqrt{k'}} \geq \frac{\sqrt{k'}}{2}$.

Remember the second cheapest path is P with cost $\frac{1}{\sqrt{k}}$. Therefore the overpayment ratio is at least $\sqrt{kk'}/2$, establishing the result. ■

Thus no incentive compatible mechanism is more frugal than VCG in the worst case. As a direct corollary of Theorem 2 we get,

Corollary 1 *There exists a graph for which any incentive compatible mechanism has a worst-case $\Omega(n)$ factor overpayment.*

Proof: Just consider two disjoint paths each of length $n/2$. ■

However for spanning tree auction or in general when the set system are the bases of a matroid, VCG indeed has a frugality ratio at most one.

Theorem 3 *The total VCG cost for spanning tree auction is at most the cost of the second cheapest disjoint spanning tree.*

and more generally,

Theorem 4 *VCG has frugality ratio $\frac{\textit{payment}}{\textit{cost of second disjoint solution}}$ at most one if and only if the feasible sets of the set system are the bases of a matroid.*

The proof is involved and omitted (see References in Section 13.5 of [2]).

4 Profit Maximization in Mechanism Design

Now we concentrate on profit maximization in mechanism design; called *optmal mechanism design* in economics. We even get rid off the truthfulness. The topic was first studied by Guruswami et al. in [3]. We consider a natural case study here.

Theorem 5 *There is a simple $\log n + \log m$ approximation for pricing when the bidders are single-minded and the items are available in unlimited supply (where n is the number of bidders and m is the number of items and we price the items individually)*

Proof: We consider pricing in which all items are priced the same. Thus the candidate prices are $q_i = \frac{V_i}{|S_i|}$. (agent i wants set S_i with value V_i and buys it only if the price is less than q_i.)

Assume $q_1 \geq q_2 \geq \ldots \geq q_n$. If all items are priced at q_i, then the seller profit is $R_i = \sum_{1 \leq j \leq i} |S_j| \frac{V_i}{|S_i|}$. By rearranging we have, $V_i = \frac{|S_i| R_i}{\sum_{j=1}^{i} |S_j|}$. Note that the price is too much for $j > i$. Because the algorithm selects the price R maximizing the profit, we have that $R_i \leq R$ for all i and thus,

$$\begin{aligned}
\sum_{i=1}^{n} V_i &= \sum_{i=1}^{n} \frac{|S_i| R_i}{\sum_{j=1}^{i} |S_j|} \\
&= \sum_{i=1}^{n} \frac{|S_i| R_i}{|S_i| + \sum_{j=1}^{i-1} |S_j|} \\
&= \sum_{i=1}^{n} R_i \sum_{k=1}^{|S_i|} \frac{1}{|S_i| + \sum_{j=1}^{i-1} |S_j|} \\
&\leq R \sum_{i=1}^{n} \sum_{k=1}^{|S_i|} \frac{1}{k + \sum_{j=1}^{i-1} |S_j|} \leq R \ln \sum_{i=1}^{n} |S_i|
\end{aligned}$$

Now $\sum_{i=1}^{n} V_i$ is a trivial upper bound on the optimum, so the theorem follows because $\sum_{i=1}^{n} |S_i| \leq nm$.

The analysis of this algorithm is tight. However the algorithm is very simple and yet it follows from the hardness of a certain coverage problem called *unique coverage problem* (under reasonable hardness assumption), that we cannot design any better algorithm for it [4]. ■

References

[1] Hajiaghayi, Mohammad T. *Algorithmic Game Theory: Lecture 1– Handwritten notes.* 1 September 2010. University of Maryland.

[2] Nisan, Noam et al. *Algorithmic Game Theory.* New York: Cambridge University Press, 2007.

[3] V. Guruswami, J. D. Hartline, A. Karlin, D. Kempe, C. Kenyon, and F. McSherry. *On Profit-Maximizing Envy-free Pricing.* Proceedings of SODA 2005.

[4] E. D. Demaine, U. Feige, M. T. Hajiaghayi, and M. R. Salavatipour. *Combination can be hard: approximability of the unique coverage problem.* Proceedings of SODA 2006.