

# Combinatorial Auctions

Nathan Immerman  
College of Engineering, University of Michigan  
Ann Arbor, Michigan  
`immerman@umich.edu`

December 17, 2015

## 1 Abstract

Combinatorial auctions are used to sell multiple goods at the same time and are particularly relevant when selling complementary and substitutive goods. In many ways combinatorial auctions are very similar to the standard auctions and matching markets that were covered in EECS 498.3 but the interdependencies between goods add additional layers of complexity that make them very interesting to analyze. In the following paper, I will summarize and analyze part of Blumrosen and Nisan's survey on combinatorial auctions focusing on applications of the combinatorial auction, the single minded bidder, the walrasian equilibrium, and bidding languages [1].

## 2 Introduction

In a standard auction, there is a single good being sold to one or more players who have a demand for that good. The auctioneer determines which player is allocated the good based on which player displays the greatest demand. There are many ways for the auctioneer to coordinate the auction, such as the ascending and descending price auctions that we discussed in class. Combinatorial auctions generalize this concept of an auction by allowing multiple goods to be sold at once to players that have demands for bundles or subsets of goods. The authors, Liad Blumrosen and Noam Nisan, in their paper on Combinatorial Auctions, analyze the computational complexity of possible allocation algorithms, how to represent and communicate the value functions of bidders, and the strategic behavior of rational bidders in combinatorial auctions. In the remainder of this paper, I will be summarizing and analyzing the main arguments made by Blumrosen and Nisan [1].

Formally, combinatorial auctions are defined as an auction where each player  $i$  has a valuation function  $v_i$  which describes their value for all bundle of goods in the auction. For each bundle  $S$ , player  $i$  receives the value  $v_i(S)$  if they receive the bundle. A player's valuation must be monotone, namely if  $S \subseteq T$  then  $v_i(S) \leq v_i(T)$  and must be normalized to  $v_i(\emptyset) = 0$ . By defining a player's valuation function in this manner we allow players to express complementary and substitutive goods in their valuations. Complementary goods are valued more together than the sum of their values individually,  $v(S \cup T) > v(S) + v(T)$ , and substitute goods are valued more individually,  $v(S \cup T) < v(S) + v(T)$ . By allowing such expressions, players can fully express their demands for many sets of goods.

As in any auction, the auctioneer in a combinatorial auction has to determine an allocation of goods. In combinatorial auctions the auctioneer has the added constraint that any one good can only be allocated to one player. Due to this constraint every allocation in a combinatorial auction is in the form  $S_1, \dots, S_n$  where  $S_i \cap S_j = \emptyset$  for every  $i \neq j$ . And from this, the social welfare of the allocation is equal to  $\sum_i v_i(S_i)$ .

In the remaining sections of this paper, I will discuss some real world applications of combinatorial auctions, a simplified bidder (the single minded bidder) that the authors introduce to simplify the complexity and strategic analysis of the auction, what is a walrasian equilibrium, and finally, how bidders can communicate their valuations through bidding languages.

### 3 Applications of Combinatorial Auctions

The primary justification for the use of an auction is when the seller does not know the values that the players have for the good being sold. Without knowing the values that players have, how could a seller pick a reasonable price? By using an auction the players determine the selling price amongst themselves. Combinatorial auctions add the additional complexity that players have values for sets of goods rather than just on a single good. This additional layer of complexity makes combinatorial auctions well suited for many real world applications including: the London bus routes, airport runway slots, and the Federal Communications Commission (FCC) airwaves spectrum auctions [2].

When the London bus system was deregulated in 1984, the committee in charge needed to determine the best way to commission companies to operate different bus routes across London. Ultimately the committee in charge decided to use a form of a combinatorial first price auction to commission the bus routes. They made this decision so companies could take advantage of economies of scale by bidding on multiple bus routes at a time. Not only could companies bid on multiple routes at a time but by using a combinatorial auction the commission allowed companies to bid on bundles of routes without requiring the company to also bid on the individual routes separately. This allowed companies to take advantage of the benefits of owning similar bus routes, such as lower fixed costs, without the risk of being allocated only some of the desired similar bus routes. By using a combinatorial auction, the commission made it possible for companies to more accurately express their valuations than if they had auctioned off each route separately, this turn lead to a more socially optimal allocation of the bus routes.

Another application is the auction of airport runway slots. Throughout the course of a day, a single airport has a limited number of runway time slots, a time when an airplane can either takeoff or land. The number of slots an airport has is based on the size of the airport, current weather conditions, the sizes of airplanes using the slots, and many other factors. These slots are well suited for a combinatorial auction because airlines have to posses many other goods to be able to utilize the runway slot. For an airline to use the runway slot they also have to have purchased an airplane gate at a terminal, baggage services, a landing runway slot at the destination airport and many other items. These interdependencies make runway slots ideally suited to be auctioned in a combinatorial auction setting, so situations where an airplane is allocated a slot to take off but not a slot to land do not happen.

Lastly, the FCC is moving towards using a combinatorial auction to sell the rights to use different spectrum bands in different geographical areas. Spectrum auctions are well suited for combinatorial auctions because for most companies having similar bands in different areas is beneficial. By using the same band in many locations, companies are able to create hardware components that are more specialized for that band. Ideally, by moving towards a combinatorial auction, the FCC can create more optimal allocations that aids in the development of the use of wireless communication.

### 4 Single Minded Bidders

There are three characteristics of combinatorial auctions that the authors primarily analyze: computational complexity, strategic behavior, and representing the valuation functions. To simplify the analysis of computational complexity and strategic behavior, the authors remove the issue of representing the valuation functions by introducing the single minded bidder who has a very simple valuation function. The single minded bidder has a valuation function  $v$  such that for a set  $S'$  and value  $v'$ ,  $v(S) = v'$  for all  $S \supseteq S'$  and  $v(S) = 0$  for all other  $S$ . This simplification allows for a valuation function that is very easy to represent, namely the pair  $(S', v')$ , and therefore simplifies the complexity and strategy analysis.

#### 4.1 Complexity Analysis

An allocation for the combinatorial auction, as with all auctions, strives to maximize social welfare, which is defined as  $\sum_i v_i(S_i)$ . Since an allocation must allocate disjoint sets of goods to each winning bidder, to maximize social welfare, we should either allocate a bundle equal to a bidders demand bundle  $S'$  or allocate the  $\emptyset$ . By doing so, there are no goods that are being wasted by being allocated to a bidder that doesn't gain value for them. Therefore the authors define an allocation to single minded bidders as follows:

**Definition 4.1.** Allocation for Single Minded Bidders

**Input:**  $(S'_i, v'_i)$  for each bidder  $i = 1, \dots, n$ .

**Output:** The winning bids  $W$  is a subset of the bids from all of the players,  $W \subseteq \{1, \dots, n\}$ . Each winning bidder in  $W$  must possess a disjoint demand bundle from all other bundles of bidders in  $W$ : for every  $i \neq j \in W$ ,  $S'_i \cap S'_j = \emptyset$ .

Definition from Blumrosen and Nisan's Combinatorial Auctions (a survey) pg. 271 [1]

This algorithm is similar to a “weighted packing” problem and is NP-complete. This can be proven by comparing the algorithm to the independent set problem which is known to be NP-complete. So even when the valuation function of bidders is simplified as much as possible without trivializing the nature of a combinatorial auction finding an optimal allocation is NP-complete. Consequently, when valuation functions are not restricted to single minded bids, an algorithm for that case must also be NP-complete because it is inherently more difficult.

The authors note that when a problem is NP-complete there are three ways to approach solving the problem: approximation, special cases, and heuristics.

**Approximation:** Not only is it known that the independent set problem is NP-complete but it is also known that approximating the independent set problem is NP-Hard. Therefore, approximating the optimal allocation of a combinatorial auction is NP-Hard.

**Special Cases:** There are some special cases of combinatorial auctions that can be solved efficiently because of simplifications that they make. The case where each bidder only demands at most two goods and the case where if all of the goods are given a linear order bidders demands are made up of continuous sets of the goods, both of which can be solved in polynomial time.

**Heuristics:** Generally, there are many heuristics that you can apply to NP-complete problems to approximate their solutions rather efficiently. By using heuristics on very large input sizes and assuming that it is possible to solve the problem optimally on smaller input sets then most real world applications can be solved efficiently.

While the authors comprehensively analyzed the complexity of the single minded bidders I wondered about more special cases and heuristics that could improve the complexity of the algorithm. For instance, what if there are specific goods that are only demanded by one bidder? Then surely to maximize social welfare it only makes sense to allocate those to the one bidder that demands it. Stemming from this, there might be other ways to simplify the algorithm given certain conditions about the contents of the demand sets of the bidders.

## 4.2 Strategy Analysis

In the course, we found that the dominant strategy for bidders in a second-price auction is to bid their true value. This notion of bidding your true value is called incentive compatibility. Incentive compatibility is beneficial so that the auctioneer (and consequently the allocation algorithm) knows that it is working with true information. The other option is for bidders to shade their bid, which adds another layer of the complexity of determining how bidders are expected to shade their bid to the allocation algorithm. Since there are many benefits of having an incentive compatible auction, the author's asked will the single minded bidder bid truthfully in a combinatorial auction?

The first option that the authors explore is can we use VCG prices to obtain incentive compatibility? While an optimal allocation at VCG prices would produce incentive compatibility, it has already been shown that we cannot easily achieve an optimal allocation. In addition, VCG prices do not produce incentive compatibility for an approximate allocation but only for an exactly optimal allocation. Therefore, we can't simply use VCG prices to get incentive compatibility. To circumvent VCG prices, the authors outline a greedy algorithm that can be used to obtain incentive compatibility, which is described below.

**Definition 4.2.** A Greedy Algorithm for Single Minded Bidders

**Initialization:** Reorder the bids s.t.  $\frac{v'_1}{\sqrt{|S'_1|}} \geq \frac{v'_2}{\sqrt{|S'_2|}} \geq \dots \geq \frac{v'_n}{\sqrt{|S'_n|}}$

**For**  $i = 1 \dots n$  **do:** if  $S'_i \cap (\bigcup_{j \in W} S'_j) = \emptyset$  then  $W = W \cup i$

**Output:** The allocation is the set of winners,  $W$ . Each winning bidder is charged a price  $p_i = \frac{v'_j}{\sqrt{|S'_j|/|S'_i|}}$ , where  $j$  is the smallest index s.t.  $S'_i \cap S'_j \neq \emptyset$ , and for all  $k < j, k \neq i, S'_k \cap S'_j = \emptyset$  (if no such  $j$  exists the  $p_i = 0$ ).

Definition from Blumrosen and Nisan's Combinatorial Auctions (a survey) pg. 274 [1]

The authors then prove that this greedy algorithm produces honest bidders. Firstly, a bidder will never receive a negative payoff if they bid truthfully. Either they lose and receive a payoff of 0 or they win. If they win their value has to be greater than or equal to the next highest bidder, which is also the price they pay, so they can not have a negative payoff. Next, we must show that a bidder cannot improve their payoff by bidding  $(\tilde{S}, \tilde{v})$ , an alternate bid, instead of their honest bid  $(S, v)$ . If  $(\tilde{S}, \tilde{v})$  produces a losing bid then surely the bidder could only improve by being honest and if  $S \not\subseteq \tilde{S}$  then the bidder cannot ever gain any payoff so for the following analysis we can assume that  $(\tilde{S}, \tilde{v})$  is a winning bid and  $S \subseteq \tilde{S}$ .

- A bidder is not worse off bidding  $(S, \tilde{v})$  with price  $p$  than  $(\tilde{S}, \tilde{v})$  with price  $\tilde{p}$ .  $(S, x)$  will be a losing bid for all  $x < p$  because  $p$  is the bid that the next bidder placed so if  $x < p$  then the bidder that bid  $p$  will win and not the current bidder. Due to the structure of the ordering of the bids in the greedy algorithm, this also means that  $(\tilde{S}, x)$  will be a losing bid for all  $x < p$  therefore  $\tilde{p}$  must be  $\geq p$  so by bidding  $(S, \tilde{v})$  the bidder still wins and the price will not increase.
- A bidder is not worse off bidding  $(S, v)$  at price  $p$  than bidding  $(S, \tilde{v})$ . If  $\tilde{v} > p$  then the bidder will still win and pay the same price thus not gaining any value and if  $\tilde{v} < p$  then the bidder will lose and have zero payoff. So then it is in the bidders best interest to bid  $(S, v)$  and not  $(S, \tilde{v})$ .
- If bidding honestly causes a bidder to lose then they have nothing to gain by raising their bid because that could never cause them to gain any value. They will either continue to lose or they will win but receive a negative payoff.

In addition, the authors also show that the greedy algorithm approximates the optimal allocation to within a degree of  $\sqrt{m}$ .

$$\sum_{i \in OPT} v'_i \leq \sqrt{m} \sum_{i \in W} v'_i$$

The formal proof can be found in the original paper on page 275 [1]. It is interesting to note that the approximation depends on  $m$ , the number of goods, but not  $n$ , the number of bidders. This means that for a constant number of goods the algorithm will have the same degree of approximation regardless of how many bidders there are. This property will later be of great use in the discussion on bidding languages in section 6.

While the authors note that VCG and even “VCG like” prices cannot be used to produce an incentive compatibility auction because VCG only is appropriate for exactly optimal situations, I would argue that the construction of the prices for the greedy algorithm described by the authors is very much in the spirit of VCG prices. VCG prices are based on the “harm” that a bidder causes to all of the other bidders by participating in the auction. So in a standard second priced auction the top bidder pays a price equal to the bid of the second highest bidder because if the top bidder had not participate then the second highest bidder would have won the auction. Similarly in the greedy algorithm, a player is charged the price that is proportional to the bid of the player that was not allocated any good because of the original player. While it is not as obvious with sets of goods as compared to single good, the prices that are charged in the greedy algorithm can be thought of as the harm that the player is causing by participating in the auction.

## 5 Walrasian Equilibrium

In all of the auction types we discussed in class it was always important to ask the question, what happens in equilibrium? First, we need to define what the demand for each bidder is formally. The authors define the demand of a bidder as follows:

**Definition 5.1.** The Demand of a Bidder: For a bidder with valuation  $v_i$  and given item prices  $p_q, \dots, p_m$ , a bundle  $T$  is called a demand of bidder  $i$  if for every other bundle  $S$  we have that  $v_i(S) - \sum_{j \in S} p_j \leq v_i(T) - \sum_{j \in S} p_j$ . Assuming that the price of each bundle is the sum of the prices of the items in the bundle.

Using this definition of a demand of a bidder, the authors define a Walrasian equilibrium as a set of “market clearing” prices where every bidder receives a bundle in his demand set, and unallocated items have prices of zero. This result also shows that a Walrasian Equilibrium will be socially optimal, this is known as The First Welfare Theorem.

As an example, there are two players Jack and Jill and two items  $\{x, y\}$ . Jack has the following value-item pairs  $\{3, x\}$  and  $\{6, x \text{ and } y\}$  and Jill has the following valuations  $\{4, y\}$  and  $\{6, x \text{ and } y\}$ . A Walrasian equilibrium exists at prices 3 for  $x$  and 4 for  $y$ . At these prices, Jack will win good  $x$  and Jill will win good  $y$  which gives a total social welfare of  $3 + 4 = 7$  which is the maximum that is possible for this auction. To prove that these prices are in equilibrium, we can deviate the prices and see that the overall social welfare decreases. If the price of good  $x$  goes down to 2 then the demand of both Jack and Jill becomes  $\{x \text{ and } y\}$  and since both Jack and Jill cannot both get  $x$  and  $y$  then either Jack or Jill will not be allocated their demand. Either Jack or Jill will be allocated their demand making the social welfare equal to 6 which is less than the maximum of 7. If the price of  $x$  goes up to 4 then no one will demand good  $x$  and  $x$  will be unallocated but since it has a price that is non-zero this violates the definition of a Walrasian equilibrium. Similarly, the price of  $y$  can be fluctuated and shown to also decrease the social welfare of the allocation. Since changing both the prices of  $x$  and  $y$  are shown to decrease the total social welfare, it is shown that the prices are a Walrasian equilibrium.

## 6 Bidding Languages

Up until this point we have assumed that the auctioneer is just given the valuation function,  $v_i$  of every bidder. In reality, for  $m$  number of goods each bidder has to communicate their value for the  $2^m - 1$  number of non-empty bundles of goods. Since this process becomes cumbersome after only just a few dozen items, this communication becomes the limiting factor in our algorithm because we are able to compute allocations efficiently for thousands of goods. For this reason, it is important to discuss different mechanisms for communicating valuation functions to the auctioneer.

When developing a bidding language we have two goals, namely simplicity and expressibility. First to keep things simple all valuation functions will be based on atomic bids. Where atomic bids are a pair  $(S, p)$  where the bidder is bidding price  $p$  for the bundle  $S$ , which is the same definition we had for single minded bidders. Then bidders can combine these atomic bids to create for complex valuation functions. They can be combined in two ways:

**OR:** A bidder can OR atomic bids together which indicates that any amount of the atomic bids can be valued at the same time. In essence the atomic bids are independent. For example, if a bid where  $(\{a, b\}, 3) \text{OR} (\{c, d\}, 5)$  and the bidder is allocated  $(\{a, b, c, d\})$  then they would get a total value of 8 because both atomic bids are being satisfied. As another example, given the bid  $(\{a, b\}, 3) \text{XOR} (\{a, c\}, 5)$ , if the bidder is allocated  $(\{a, b, c\})$  the bidder only receives a value of 5 because not both sets can use the good  $a$  at the same time.

**XOR:** When a bidder XORs atomic bids together they are indicating that only 1 of the atomic bids can be satisfied at the same time. For example, if a bid where  $(\{a, b\}, 3) \text{XOR} (\{c, d\}, 5)$  and the bidder is allocated  $(\{a, b, c, d\})$  then they would get a total value of 5, because only one of the atomic bids is permitted to be satisfied.

By using a combination of ORing and XORing atomic bids it is possible to create very complex valuation functions that accurately describe a players valuations.

The authors further show that it is possible to simplify this bidding language even further by representing XOR bids as OR bids. This is done by creating dummy items for each bidder which can be used to make OR bids act like XOR bids. For example give the dummy good  $d$ , the bid  $(\{a\}, 3)XOR(\{b\}, 5)$  is equivalent to  $(\{a, d\}, 3)OR(\{b, d\}, 5)$ . This works because the dummy good can only be used by the bidder to satisfy one set, which therefore makes each set mutually exclusive.

Being able to represent any valuation function solely as a ORing of atomic bids is very powerful for the following reason. A bidder that submits many ORed atomic bids looks the same as multiple bidders that each submitted a single atomic bid to an allocation algorithm. This is because the allocation algorithm is just trying to optimize the total social welfare so the algorithm doesn't care who is getting the value from the set that is being allocated but just cares about how that set is effecting the total social welfare. This means that all of the analysis and algorithms that were discussed for single minded bidders can be applied to valuation functions of ORs and XORs, which are much more expressive and accurate than the simplified single minded bids.

It is important to reiterate the importance that the greedy algorithms approximation  $(\sum_{i \in OPT} v'_i \leq \sqrt{m} \sum_{i \in W} v'_i)$  is not dependent on the number of bidders. Due to this lack of dependency, the generalization that a single bidder with multiple ORed atomic bids is equivalent to multiple bidders with single bids does not effect the quality of the greedy algorithm.

## 7 Conclusion and Suggestions for Future Work

Combinatorial auctions can be applied to many applications where a set of goods needs to be auctioned off and when players have valuations that depend on bundles of goods and not just single goods, such as the London bus routes, airport runway slots, and FCC spectrum auctions. Blumrosen and Nisan outlined a survey of many theoretical considerations when analyzing and designing a combinatorial auction and in this paper I summarized and analyzed a few of these topics [1]. We started with the single minded bidder, whose valuation pair  $(S', v')$  provided a simple way to begin analyzing the computational complexity and strategic behavior of the auction. We next discussed walrasian equilibria, which possess market clearing prices where every player is allocated a bundle in their demand. And finally, we discussed bidding languages and how bidders can effectively communicate their valuations to the auctioneer. While bidding languages turned out to be the computational bottle neck of the auction algorithm, we still determined a way to represent very complex valuations while still being able to apply all of the analysis and algorithms for single minded bidders.

One of the key skills that was taught in the course was the ability to simplify a general definition of a concept based on specific constraints for a particular problem. For example, finding the page rank of an undirected and connect graph. I think that it would be beneficial to develop a set of simpler, special cases of combinatorial auctions that can then be applied alone or in combination to different real world applications. By analyzing more special cases, it is then more likely that a new real world application can be applied to a special case instead of the general auction which would make running and analyzing the auction simpler.

## References

- [1] Liad Blumrosen and Noam Nisan. *Combinatorial Auctions (a survey)*. In Algorithmic Game Theory, 2007.
- [2] Peter Cramton, Yoav Shoham, and Richard Steinberg, editors. *Combinatorial Auctions*. MIT Press, 2006.