

Fair Mechanisms for Recurrent Multi Unit Combinatorial Auctions

Javier Murillo and Beatriz López¹

Abstract. Auctions have been used to deal with resource allocation in multi-agent systems. In some environments like service-oriented electronic markets, it is advisable to use recurrent auctions since resources are perishable and auctions are repeated over time with the same or a very similar set of agents. Recurrent auctions are a sequence of one-shot auctions of any kind. As a drawback some problems do appear that could cause the market to collapse at mid-long term. Previous works have dealt with these problems by adding fairness to the auction outcomes but they dealt with multi-unit auctions, in which several units of an item are sold. In this paper, we present a new fair mechanism for multi-unit combinatorial auctions, in which different items, and several units per item are sold in each auction.

1 INTRODUCTION

In auctions, traditionally, the auctioneer exhibits an utilitarian behavior in which he tries to maximize his own benefits by giving the resources to the agents that offer the highest bids. Recent works, as [3, 5] have shown that this utilitarian behavior can cause the collapse of the market at the mid-long run when dealing with recurrent markets. Thus, this rational behavior exhibited in a single shot game, becomes unsuccessful when repeating the game or auction several times. The reason for that is that this behavior produces the inevitable starvation of certain buyers that could leave the market [4]. So, when poor agents leave, richer bidders have the power to set the prices of the resources, provoking a fall in prices that can collapse the market. This problems worsen when dealing with perishable resources [5], as in our case (bandwidth communication allocation).

Some authors have equipped the traditional mechanisms with reservation prices [6]. But with reservation prices, many resources are not sold, and if the resources cannot be stored, they are lost. Thus, if the auctioneer wishes to obtain the maximum revenue from the resources along all the auctions, he needs to look for alternative mechanisms to solve the auction. Particularly, [3, 5] show that if each auction is solved with a fair mechanism, at the end of the recurrent auction higher benefits are obtained. A fair solution means that at long term, all of the participants accomplish their goals in the same degree, independently of their wealth. The inclusion of fairness can be somewhat acting against short-term optimality, since the result of an auction may differ from the optimal solution if a suboptimal solution is fairer. However, its mid or long-term effect produces an increase of auctioneer benefits, since it maintains the interest of bidders in continuing in the auction process.

However, such previous works consider that the auctioneer sells a constant quantity of units of a unique resource and bidders can only send a unique bid to obtain one unit of the resource. Real

world, however, is different, and agents can compete for different resources. Thus, we are dealing with a multi-unit combinatorial auction (MUCA), in which different kind of resources, and several units of each resource are auctioned.

In this paper we present a new auction mechanism that we have called dynamic reservation price MUCA (DRPMUCA). The extension from a unique kind of resources to multiple kind of resources is not straightforward, and this is the main contribution of this paper. Fairness and reservation prices are combined in order to obtain the largest benefits at mid-long run.

2 Dynamic Reservation Price MUCA (DRPMUCA)

Our starting point are recurrent multi-unit, multi-item, sealed-bid combinatorial auctions. *Recurrent* auctions are a sequence of auctions (of any kind) that are solved individually, but the outcomes of the auctions could be influenced by previous auctions. A *combinatorial* auction is an auction in which bidders can place bids on combination of items, or packages, rather than just individual items [1, 2]. In a *multi-unit* auction the auctioneer offers several (identical) units of each auctioned item. In a *multi-item* auction the auctioneer offers several items or resources. Finally a *sealed-bid* auction is one where the auctioneer is the only agent that knows the bids sent by the bidders. Regarding the payments, in our case each bidder pays the price he bids (*first-price* auction).

Then, we propose a mechanism for clearing the auction based on reservation prices and fairness, DRPMUCA. In DRPMUCA the auctioneer has a reservation price for each pair bidder/package sent by the bidder agent. The reservation price $rp_{i,m}$ is defined as the minimum price at which the auctioneer is willing to sell the package $S_{i,m}$ to the agent g_i . A package $S_{i,m}$ contains the items required by the agent and the number of units of each item (e.g. $\langle bandwidth, 2 \rangle, \langle memory, 5 \rangle$). DRPMUCA is composed by three phases: Reservation-price clearing, priority-base clearing, and updating.

Reservation-price clearing. In the first phase the auction is solved in the traditional way but only taking into account the bids that are higher than the corresponding reservation price.

Priority-base clearing. Once the winners of the first phase are determined, the idea of the second phase is to sell the resources that are not being distributed in the first phase between the loser bidders with higher priority without taking into account the bid price. But in order to an agent become a winner in the second phase, the priority of the bid has to be greater than a parameter ξ . This minimum priority value avoids the rich agents to obtain the resources at low price due to the fair distribution. The value of each priority depends on the number of lost auctions since the bidder won a package for the last time. The

¹ University of Girona, Spain, email: {jmurillo, blopez}@eia.udg.edu

more auctions the bidder has lost the greater the priority. The priority $p_{i,m}$ of an agent g_i and a package $S_{i,m}$ is calculated using Equation 1 where λ is the initial priority and γ indicates the number of lost auctions necessary so that the priority arrives to its maximum value (1.0). $l_{i,m}$ is the number of lost auctions in the current cycle for package $S_{i,m}$.

$$p_{i,m} = \begin{cases} l_{i,m} < \gamma & p_{i,m} = \lambda + \frac{1-\lambda}{\gamma} \cdot l_{i,m} \\ l_{i,m} \geq \gamma & 1 \end{cases} \quad (1)$$

To reduce the risk of cheating, we define a *cycle* as a set of consecutive auctions where the priority of the agent g_i for the package $S_{i,m}$ is not decreasing. Each agent has a different cycle for each package. When a cycle ends the priority is initialized with an initial value. A cycle ends and another starts when a bidder agent wins the package or when the agent sends a bid for the package with a price higher than the bid price that the agent sent at the first auction of the cycle. If a bidder agent sends more than one bid, the agent could have a different priority for each bid since each bid requests a different package.

Updating. Finally, in the last phase of DRPMUCA, the information related to priorities and reservation prices are updated. For each agent, if the price of a bid that requires the package $S_{i,m}$ is greater than $rp_{i,m}$, $rp_{i,m}$ will be incremented for the next auction since the bidder is showing that he can pay more for the package. Conversely if the price is lower, $rp_{i,m}$ will be decremented. Note that changes in $rp_{i,m}$ are independent of whether the bids are winners or losers, or they participate or not (if they do not participate is equivalent to send a bid with price 0 for all of the packages, and the reservation price is decreased). We define $\delta \in [0, 1]$ as the minimum percentage of increment or decrement. When a $rp_{i,m}$ is incremented, the increment is the maximum between the half of the difference between the price and the $rp_{i,m}$ and $rp_{i,m} \cdot \delta$. When a $rp_{i,m}$ is decremented, the new $rp_{i,m}$ is $rp_{i,m} - (rp_{i,m} \cdot \delta)$. Reservation prices are updated at the end of each auction and are used to solve the next auction.

3 EXPERIMENTATION

We have created a bandwidth allocation scenario based on extending the one described in [3] to multi unit combinatorial auctions. There are 4 auctioneers following different strategies plus our mechanism: TMUCA (traditional auction), PMUCA (based on priorities), RP-MUCA (based on reservation prices as in [4]) and DRPMUCA. Each auctioneer is a bandwidth provider that owns 60 units of bandwidth of different quality (i.e., five different kind of resources). There are 100 bidder agents that are requesting services. Bidders can be classified in six groups. Each group requests different combinations of the five different resources offered by the auctioneer. We have chosen the XOR bidding language because it allows the bidders to express any preference over resources [1]. The bidder sends a set XOR of bids, consequently only one or none can be winner. In order to implement the bidding policy of bidders and to model the bidder drop problem we have adapted the strategy proposed in [3] to combinatorial auctions. The wealth of bidders has been distributed following a uniform distribution in [2, 10].

Table 1 shows the total benefits, the resource waste produced, the fairness level, and the number of bidders remaining with the auctioneer at the end are shown. The table values are the average of 20 simulations with the standard deviation below. Results show that the bidder drop problem affects in a major way TMUCA and RP-MUCA. In the case of TMUCA, the drop of bidders has provoked

the collapse of the auction. Regarding the resource waste, RP-MUCA has been the mechanism most affected (31.87%). Below these mechanisms, DRPMUCA has lost 15.32% of resources and finally the mechanism not based on reservation prices are the ones that have wasted less resources. Regarding the total benefits, DRPMUCA has been the mechanism that has obtained the first position followed by RP-MUCA, and PMUCA. Finally regarding fairness, DRPMUCA is the mechanism with a higher value followed by PMUCA. We can see that the methods with higher fairness obtains the highest benefits.

Table 1. Results for all mechanisms with a uniform distribution of wealth in scenario 1. Averaged results of 20 simulations

Name	◇	□	△	⋈
PMUCA	550119.74 60183.99	5729.30 1480.90	68.60 2.43	3.51 0.08
DRPMUCA	666314.48 41969.16	18384.25 2283.99	66.50 2.18	4.10 0.12
TMUCA	178297.63 29690.01	2439.00 1835.44	52.25 1.58	2.71 0.08
RP-MUCA	549565.30 53360.16	38255.10 6779.03	45.25 3.64	3.10 0.03

◇ Total benefits □ Resource Waste △ Number of bidders ⋈ Fairness

4 CONCLUSIONS

In environments where resources are perishable and the allocation of resources is repeated over time with the same set or a very similar set of agents, recurrent auctions come up. A recurrent auction is a series of auctions where the result of one auction can influence the following ones. However, these kinds of auctions have particular problems when the wealth of the agents is unevenly distributed and resources are perishable. Fair mechanisms have already been proposed in order to deal with this problem in non-combinatorial auctions. In this work we have presented one mechanism for recurrent combinatorial auctions based on priorities and reservation prices that minimizes these problems. The results show that fairness gives bidders incentives to stay in the auction and reservation prices help maintaining the equilibrium in negotiation power.

REFERENCES

- [1] *Combinatorial Auctions*, eds., Peter Cramton, Yoav Shoham, and Richard Steinberg, MIT Press, 2006.
- [2] Sven de Vries and Rakesh Vohra, 'Combinatorial auctions: A survey', Discussion Papers 1296, Northwestern University, Center for Mathematical Studies in Economics and Management Science, (May 2000).
- [3] J-S. Lee and B.K. Szymanski, 'A novel auction mechanism for selling time-sensitive e-services', in *IEEE International Conference on E-Commerce Technology*, pp. 75–82. IEEE Computer Society, (2005).
- [4] J-S. Lee and B.K. Szymanski, 'Stabilizing markets via a novel auction based pricing mechanism for short-term contracts for network services', in *Integrated Network Management*, pp. 367–380, (2005).
- [5] J. Murillo, V. Muñoz, B. López, and D. Busquets, 'A fair mechanism for recurrent multi-unit auctions', in *Lecture Notes in Computer Science*, volume 5244, pp. 147–158. Springer, (2008).
- [6] John G. Riley and William F. Samuelson, 'Optimal auctions', *The American Economic Review*, **71**(3), 381–392, (1981).