Virtual Worlds: Fast and Strategyproof Auctions for Dynamic Resource Allocation

Chaki Ng* David C. Parkes* Margo Seltzer*

ABSTRACT

We consider the problem of designing fast and strategyproof exchanges for dynamic resource allocation problems in distributed systems. The exchange is implemented as a sequence of auctions, with dynamically arriving requests from agents matched with each auction. Each auction is associated with some consignment of the resources from a single seller. We provide a simple Virtual Worlds (VW) construction, that extends a fast and strategyproof mechanism for a single auction to apply to this sequence-of-auctions setting. Rather than match each buyer with a single auction, the VW mechanism allows buyers to be considered for multiple auctions while retaining strategyproofness.

Categories and Subject Descriptors

F.2 [**Theory of Computation**]: Analysis of Algorithms and Problem Complexity; J.4 [**Computer Applications**]: Social and Behavioral Sciences—*Economics.*

General Terms

Algorithms, Economics.

1. INTRODUCTION

Computation is increasing distributed and performed by devices representing multiple individuals and businesses. Consider a data staging scenario, in which multiple users with PDAs are in Times Square and trying to read and access corporate databases. Each user would like to stage (encrypted) data within nearby physical environment to reduce latencies. A socially-efficient allocation of data staging capabilities in Times Square would allocate capacity to maximize the total value across all users.

With cooperative users, and other computational considerations aside, one could simply ask devices to state their utilities for various outcomes and then implement the socially-efficient allocation. However, rational and self-interested users would overstate their utility for the ability to stage their own data, and cause the system performance to quickly unravel.

In this paper we propose a fast and scalable mechanism for dynamic resource allocation. The mechanism is novel, in that it allows agents to be matched against a sequence of auctions while retaining strategyproofness. We view this as a step towards the development of general infrastructure-level support for *strategyproof computing* in distributed systems [3].

2. THE VIRTUALWORLDS MECHANISM

To start, we model a simplified problem with one seller, with C units of a homogeneous resource available in each period of time t. Request agent i arrives at time t_i, with private information about her value v_i for s_i units of resource for duration l_i. All request agents (RA's) are assumed to have *patience* Δ, which is the amount of time that any RA will wait to receive the resource. We assume that the arrival time, and patience, is known to the system.[1]

The VIRTUAL WORLDS (VW) mechanism partitions the available resources into $k = 1, 2, \ldots$ auctions, where auction k provides C_k units of resource, of duration l_k, starting at time t_k. Let A_k denote the set of RA's that arrive into the system between auction $k-1$ and auction k. We refer to each new auction as a new *period*. The mechanism allows an RA to be matched against a sequence of auctions. The RA receives resources from the first auction in which it is a winner, and immediately upon a successful match. However, the RA's final payment is not determined until the first period k, with $t_k \geq t_i + \Delta$ (i.e. after its patience duration). Let $p_{\mathrm{virt}}(i)$ denote the current payment calculated for agent i. This is defined to be zero for an agent that has not yet received a match, and otherwise will *monotonically decrease* while the agent remains in the system.

The VW mechanism maintains a set of *active bidders*, N_k, at the start of each period. These are the RA's with $t_k - \Delta \leq t_i \leq t_k$, that are either winners in some previous period with $p_{\mathrm{virt}}(i) > 0$, or have not yet received a successful match. The set N_k includes the agents A_k that have arrived into the system between period $k-1$ and period k. Let $W_k \subseteq N_k$ denote the set of active bidders that are already winners, at the start of period k. This captures the essential information about the state of the system. In VW, each winner has an associated *virtual world* while it is active.

Definition. The Virtual World for active bidder i in period k defines the state of the system that would exist if bidder i had not arrived until period k but everything else was unchanged.

Let VW_k denote the set of active bidders that are already winners at the start of period k, and also the bidders that win for the first time in period k. The set of winners, $W_k(i) \subseteq N_k$, that would exist if bidder i had just arrived in period k, captures the essential state information about the virtual world.

In describing the VW mechanism, we are careful to distinguish

*Division of Engineering and Applied Sciences, 33 Oxford Street, Harvard University, Cambridge, MA 02138. Email: {chaki,parkes,margo}@eecs.harvard.edu. Supported in part by National Science Foundation grant IIS-0238147.

EC'03, June 9–12, 2003, San Diego, California, USA.
ACM 1-58113-679-X/03/0006.

[1] See Friedman & Parkes [1] for a recent discussion of a VCG-based approach to *online mechanism design*, where agents can misstate arrival times.

between the “real world”, in which resource allocation decisions are made and initial payments determined, and the virtual worlds in which the agent’s payments are refined. In each period, the VW mechanism performs the following two phases:

Phase I. Run the greedy auction scheme of Lehmann et al. [2] (hereafter LOS) to clear the auction in the real world. In particular, we take bids from the subset of agents not yet matched, $N_k \setminus W_k$, that request resources that fit within the capacity C_k and duration l_k offered in period k.[2] These are bids from all agents $B_k = \{i : i \in N_k \setminus W_k, s_i \leq C_k, l_i \leq l_k\}$. The LOS scheme clears auction k as follows:

[1] Sort the bids in B_k in order of decreasing v_i/s_i, with $v_1/s_1 \geq v_2/s_2 \geq \ldots \geq v_N/s_N$.

[2] Let $x_i \in \{0, 1\}$ denote whether bid i is accepted. Initialize $x_i = 0$ for all i. Walk down the bids from 1 to N. Greedily accept bid j while $C_k \geq \sum_i x_i s_i \ + s_j$, and set $x_j = 1$. Let x^* denote this solution.

[3] For each winner i, with $x^*_i = 1$, repeat step [2] with all bids B_k except this bid. Call this solution x^{-i}. Let p_i denote the per-unit bid price, v_j/s_j of the first bid j that is not a winner in x^* but is a winner in x^{-i}, or zero if there is no such bid.

[4] Implement allocation x^*, with each winner i with $x^*_i = 1$ making payment $s_i p_i$.

The payment $p_{\text{virt}}(i)$ to a winning agent is initialized to the payment computed in this LOS scheme. In addition, the new winners are added to the set of agents with virtual worlds, VW_k. Then, in the virtual world for agent i, the set of winners $W_k(i)$ is initialized to W_k.

Phase II. For each virtual world, $i \in VW_k$, run the LOS scheme twice. *First*, run the LOS scheme with $N_k \setminus W_k(i)$, where this includes agent i. If bidder i is a winner, then update the price $p_{\text{virt}}(i)$, to the smallest of the current $p_{\text{virt}}(i)$ and the price in this virtual world. *Second*, run the LOS scheme with $N_k \setminus (W_k(i) \cup i)$, i.e. without bidder i. Propagate this as the state of the virtual world for the next period. The first stage of Phase II is not necessary in the first period that the VW is created.

The effect of Phase II is to compute the payment in period k for an RA that has been matched in the previous period, as though it just arrived into the system for the first time in period k. If the RA wins for a lower payment in period k, then this payment becomes the new VW payment.

Interestingly, we can establish the strategyproofness of the VW mechanism as a simple corollary to the incentive properties of LOS. The VW mechanism maintains properties:

Exactness. Bids are either accepted in full or denied.

Monotonicity. If a bid (s_i, v_i) loses, then a bid (s'_i, v'_i) with $s'_i \geq s_i$ and $v'_i \leq v_i$ also loses.

Participation. Only winners make payments.

Critical. The payment is exactly equal to the minimal price at which a bid, for the true size, would have still been accepted.

Taken together, these properties are sufficient for strategyproofness, and imply our main result:

THEOREM 1. *Truth-revelation of (s_i, l_i, v_i) is a dominant strategy for RA’s in the* VIRTUALWORLDS *mechanism.*

PROOF. (sketch) Exactness and participation trivially hold, while monotonicity is inherited from the greedy LOS scheme. The VW construction provides the *critical* property by explicit construction across all periods in which the agent is equally happy to receive a match. □

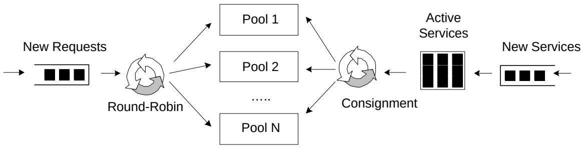

Figure 1: Exchange architecture

3. EMBEDDING WITHIN AN EXCHANGE

In a practical resource allocation mechanism there are multiple sellers, in addition to multiple buyers. In this section, we briefly explain how the VW mechanism is embedded within a two-sided exchange via a *consignment and pooling* mechanism. The reader is referred to the full paper for complete details [4].

Figure 1 depicts the overall architecture of the exchange. First, to provide scalability, incoming RA’s are divided into multiple pools, with each pool associated with a single VW mechanism. New RAs are queued up in the system. Before starting a new round of auctions, each queued RA is assigned to join one of the auction pools, in round-robin fashion. Second, arriving SA’s are also queued, before the resources are chunked into a *consignment* of resources and scheduled to a sequence of auctions (possibly within multiple pools). Each auction is associated with a consignment of resources from a single SA. The auction pools operate independently of each other and are monitored by the exchange. The basic information a pool needs to start a new auction is the current RAs assigned to the pool, the SA assigned to the next auction, and the consignment of capacity that it will make available.

Arriving SA’s report a capacity C_j and total duration l_j. The consignments are constructed to provide each SA with access to the same total number of auctions, irrespective of its reported (C_j, l_j). Instead, the consignment within a single auction is scaled. This is useful because it removes any incentive for an SA to *overstate* its available capacity. In the longer version of this paper [4], we analyze the remaining, but limited, opportunity that can exist for an SA to *understate* its capacity and increase its payments.

We have implemented VIRTUALWORLDS on Sun’s JXTA platform and Berkeley DB and obtained initial positive results. We are deploying pools on multiple servers for scalability and are currently conducting further validation of the performance across a number of different resource allocation scenarios. For the future, we will extend and deploy the mechanism for computational grids.

4. REFERENCES

[1] E. Friedman and D. C. Parkes. Pricing WiFi at Starbucks– Issues in Online Mechanism Design. In *Proc. Fourth ACM Conf. on Elec. Commerce (EC’03)*, 2003. Extended version at http://www.eecs.harvard.edu/econcs/pubs/online.pdf.

[2] D. Lehmann, L. I. O’Callaghan, and Y. Shoham. Truth Revelation in Approximately Efficient Combinatorial Auctions. *Journal of the ACM*, 49(5):577–602, September 2002.

[3] C. Ng, D. C. Parkes, and M. Seltzer. Strategyproof Computing: Systems Infrastructures for Self-Interested Parties. Technical report, Harvard University, 2003.

[4] C. Ng, D. C. Parkes, and M. Seltzer. Virtual Worlds: Fast and Strategyproof Auctions for Dynamic Resource Allocation. In *Fourth ACM Conf. on Elec. Commerce (EC’03)*, 2003. Extended version at http://www.eecs.harvard.edu/econcs/pubs/virtual.pdf.

[2] We are careful to allocate an RA with reported length l_i to *any* auction with $l_k \geq l_i$, rather than some subset of auctions with l_k “close” to l_i. Otherwise, there could be an incentive for an RA to overstate its required duration l_i in order to change the pool of auctions with which it is matched and change its payment.