An Iterative Auction Mechanism for Combinatorial Exchanges

Shantanu Biswas [1] **Y Narahari**
Electronic Enterprises Laboratory
Department of Computer Science and Automation
Indian Institute of Science
Bangalore - 560 012, India
{biswas, hari}@csa.iisc.ernet.in

November 7, 2003

[1]This research is supported in part by IBM Research Fellowship awarded to the first author by IBM India Research Lab, New Delhi

Abstract

Combinatorial exchanges are generalizations of combinatorial auctions. They allow multiple buyers and sellers to trade with each other and provide for supply and demand aggregation. The allocation problems in combinatorial exchanges are NP-Hard and therefore, in practice, we can only solve these problems approximately. In this paper we develop an iterative auction mechanism for combinatorial exchanges with periodic clearance to maximizes the allocative efficiency.

Keywords: Combinatorial exchanges, iterative auction, integer formulations, primal-dual formulation, complementary slackness conditions.

1 Introduction

Exchanges are double-sided marketplaces where buyers and sellers submit bids and asks for trading. Markets have three main functions [1]: (1) matching buyers to sellers; (2) facilitating exchange of information, goods, services, and payments associated with a market transaction; and (3) providing an institutional infrastructure. Combinatorial exchanges are generalizations of one-sided combinatorial auction mechanisms [10]. Combinatorial bids and asks allow the agents to express synergies and substitutabilities between items. Applications of combinatorial exchanges have been suggested to many areas such as bandwidth exchanges [4], excess steel inventory procurement [5], and supply chain coordination [12]. Private marketplaces (eg. procurement exchanges) run by companies are becoming quite popular [4].

The main issues in an exchange are

- *winner determination* i.e. determining what is traded and by which agents
- *pricing* i.e. determining the net payment from or to each agent when the exchange clears.
- *timing* i.e. Continuous or periodic clearance

The rules used to clear and price the trades impact the allocative efficiency of the exchange. Also there can be two kinds of bid submissions:

- Iterative exchanges iterate the bid submission, clearing, and information disclosure to the bidders.
- In one-shot exchanges, agents submit sealed bids in a single round.

Periodic clearance can be more efficient because it allows more opportunities for aggregations across multiple bids and asks [10].

Useful economic properties of an exchange include [10]:

- **Allocative-efficiency** (EFF): Trade should be executed to maximize the total increase in value over all agents.
- **Individual-rationality** (IR): No agent should pay more than its net increase in value for the items it trades.
- **Budget-balance** (BB): The total payments received by the exchange from agents should be at least the total payments made by the exchange to agents.
- **Incentive-compatibility** (IC): The equilibrium strategy for the agents is to reveal *truthful* information about their preferences in their bids.

The impossibility result of Myerson and Satterthwaite [8] demonstrates that no exchange can be simultaneously EFF, BB and IR. This result holds with or without IC and for both dominant strategy and Bayesian-Nash equilibrium [10]. Also to achieve allocative efficiency in a combinatorial exchange, we have to solve set packing or set covering problems which are known to be NP-Hard problems [2, 11]. So it is impossible to achieve all the desirable economic properties. We present a mechanism which is budget balanced and individually rational and achieves allocative efficiency approximately.

Bikhchandani et al [2] have shown that in a combinatorial exchange with multiple buyers and sellers, even with discriminatory pricing, competitive equilibrium may not exist. Therefore we can only achieve competitive equilibrium under special conditions. Our motivation in this paper is to formulate an iterative auction scheme which drives the exchange towards a market clearing price instead of exploring conditions under which competitive equilibrium exists.

1.1 Contributions and Paper Outline

We know the allocation problems in combinatorial exchanges are NP-Hard. Thus, in practice, we can only solve these problems approximately. In this paper our aim is to develop an iterative exchange with periodic clearance to maximize the allocative efficiency. We propose an iterative auction mechanism where the trade is settled at the final market clearing price. Since it is an iterative mechanism the agents bid only if it is individually rational for them. So we do not have to impose the individual rationality as a hard constraint.

In Section 2, we discuss the mathematical formulations for the combinatorial exchanges. In Section 3, we present our iterative auction mechanism for combinatorial exchanges. We conclude and discuss the need for further work in Section 4.

2 Mathematical Formulation of Combinatorial Exchanges

2.1 Description of Combinatorial Exchanges

2.1.1 Objective of the Exchange

We can formulate a combinatorial exchange in two ways [5, 6].

- **Surplus Maximization**: Maximize the difference between the revenue collected from the buyers and the amount paid to the sellers, i.e. maximize the revenue of the exchange.
- **Maximize Trade**: Maximize the total amount traded over all goods with budget balance constraint.

If we consider revenue maximization then we do not have to consider budget balance as a separate constraint, because the trade is budget balanced whenever the maximum surplus is positive. In this paper, we shall consider the model of an exchange with surplus maximization as its goal.

2.1.2 Bid Structure

We shall assume that the agents use XOR bidding language. *XOR Bids (All-Or-Nothing Bids):* Each bidder can submit multiple bids that are XOR'ed together. A bid has the form

$$(S_1, p_1) \oplus (S_2, p_2) \oplus ... \oplus (S_j, p_j)$$

where:

- S_i: Subset in which the agent is interested
- p_i: Price at which the agent is willing to buy or sell the bundle S_i.
- at most one (S_i, p_i) is to be accepted

Such bids can encode complementarity and substitutability. They can also be used for expressing volume discounts.

2.1.3 Free Disposal

We assume that the extra goods can be freely disposed, i.e. the availability can be greater than or equal to the demand.

2.2 Integer Programming Formulation

The exchange problem can now be formulated as:

$$\begin{aligned}
V \quad &= \max \left\{ \sum_{j \in N} \sum_{S \subseteq G} v(S,j) y(S,j) - \sum_{i \in M} \sum_{S \subseteq G} v(S,i) y(S,i) \right\} \\
s.t. \quad & \\
&\sum_{S \subseteq G} y(S,i) \leq 1 \;\; \forall i \in M \\
&\sum_{S \subseteq G} y(S,j) \leq 1 \;\; \forall j \in N \\
&\sum_{S \ni k} \sum_{i \in M} a(S,k) y(S,i) \geq \sum_{S \ni k} \sum_{j \in N} a(S,k) y(S,j) \;\; \forall k \;\; \forall S \subseteq G \\
&y(S,i) = 0, 1 \;\; \forall S \subseteq G, \forall i \in M \\
&y(S,j) = 0, 1 \;\; \forall S \subseteq G, \forall j \in N
\end{aligned}$$

where

- $i = 1, ..., M$: Seller agents

- $j = 1, ..., N$: Buyer agents
- $k = 1, ..., K$: is any item
- $G =$ The set of all items
- $S \subseteq G$ is any subset of G
- $a(S, k) =$ quantity of item k in the subset S
- $v(S, i) =$ Value (or Reservation Price) of the subset S to seller i
- $v(S, j) =$ Value (or Valuation) of the subset S to buyer i
- $y(S, i),\ y(S, j) =$ Boolean variables indicating whether an agent gets the set S

The values $v(S, i), v(S, j)$ are private values and are known only to the respective agents.

2.3 Linear Programming Relaxation

The linear programming relaxation (LP-1) of the integer formulation of the exchange is:

$$\begin{aligned}
V &= \max\left\{\sum_{j\in N}\sum_{S\subseteq G} v(S,j)y(S,j) - \sum_{i\in M}\sum_{S\subseteq G} v(S,i)y(S,i)\right\} \qquad (1)\\
s.t.&\\
&\sum_{S\subseteq G} y(S,i) \leq 1 \ \ \forall i \in M\\
&\sum_{S\subseteq G} y(S,j) \leq 1 \ \ \forall j \in N\\
&\sum_{S\ni k}\sum_{i\in M} a(S,k)y(S,i) \geq \sum_{S\ni k}\sum_{j\in N} a(S,k)y(S,j) \ \ \forall k \ \ \forall S \subseteq G\\
&y(S,i) \geq 0 \ \ \forall S \subseteq G, \ \ \forall i \in M\\
&y(S,j) \geq 0 \ \ \forall S \subseteq G, \ \ \forall j \in N
\end{aligned}$$

Dual (DLP-1) of the relaxed exchange formulation is given by:

$$\begin{aligned}
DLP &= \min \sum_i p(i) + \sum_j p(j) \qquad (2)\\
s.t.&\\
&p(j) + \sum_{k\in S} a(S,k)p(k) \geq v(S,j)\\
&p(i) - \sum_{k\in S} a(S,k)p(k) \geq -v(S,i)\\
&p(i), p(j), p(k) \geq 0 \ \ \forall i, j, k
\end{aligned}$$

The feasible dual solution can be interpreted as:
Seller's Problem:

$$p(j) = \max_S \{v(S,j) - \sum_{k \in S} a(S,k)p(k)\} \;\; \forall j$$

Buyer's Problem:

$$p(i) = \max_S \{\sum_{k \in S} a(S,k)p(k) - v(S,i)\} \;\; \forall i$$

$$\Rightarrow \quad p(i) = \min_S \{v(S,i) - \sum_{k \in S} a(S,k)p(k)\} \;\; \forall i$$

Therefore we have the following interpretations:

- $p(k)$: Price of the item k
- $p(i)$: The maximum utility of seller agent i at prices $p(k)$
- $p(j)$: The maximum utility of buyer agent j at prices $p(k)$

2.3.1 Complementary Slackness Conditions

1. The first primal complementary slackness (CS-1) is:

$$y(S,i) > 0 \Rightarrow p(i) = \sum_{k \in S} a(S,k)p(k) - v(S,i)$$

2. The second primal complementary slackness (CS-2) is:

$$y(S,j) > 0 \Rightarrow p(j) = v(S,j) - \sum_{k \in S} a(S,k)p(k)$$

3. The first dual complementary slackness (CS-3) is:

$$p(i) > 0 \Rightarrow \sum_S y(S,i) = 1 \;\; \forall i, S$$

4. The second dual complementary slackness (CS-4) is:

$$p(j) > 0 \Rightarrow \sum_S y(S,j) = 1 \;\; \forall j, S$$

5. The third dual complementary slackness (CS-5) is:

$$p(k) > 0 \Rightarrow \sum_{S \ni k} \sum_{i \in M} a(S,k)y(S,i) = \sum_{S \ni k} \sum_{j \in N} a(S,k)y(S,j) \;\; \forall k \;\; \forall S \subseteq G$$

Since we have assumed free disposal, therefore this condition is equivalent to

$$\sum_{S \ni k} \sum_{i \in M} a(S,k)y(S,i) \geq \sum_{S \ni k} \sum_{j \in N} a(S,k)y(S,j) \;\; \forall k \;\; \forall S \subseteq G$$

2.4 Bounds

2.4.1 ϵ Complementary Conditions

For any bundle S

1.
$$p_{ask}(S) - \epsilon \leq p_{bid,j}(S) \leq p_{ask}(S) \tag{3}$$

2.
$$v(S,j) - p_{bid,j}(S) + \epsilon \geq \max_{S'}\{v(S',j) - p_{bid,j}(S')\} \tag{4}$$

3.
$$v(S,j) - p_{bid,j}(S) \geq 0 \tag{5}$$

Substitute

$$\sum_{k \in S} a(S,k)p(k) = p_{ask}(S)$$

$$\max_{S'}\{v(S',j) - p_{bid,j}(S')\} = p(j)$$

We get

$$v(s,j) - \sum_{k \in S} a(S,k)p(k) + 2\epsilon \geq p(j) \tag{6}$$

For any bundle S

1.
$$p_{bid}(S) \leq p_{ask,i}(S) \leq p_{bid}(S) + \epsilon \tag{7}$$

2.
$$p_{ask,i}(S) - v(S,i) + \epsilon \geq \max_{S'}\{p_{ask,i}(S') - v(S',i)\} \tag{8}$$

3.
$$p_{ask,i}(S) - v(S,i) \geq 0 \tag{9}$$

Substitute

$$\sum_{k \in S} a(S,k)p(k) = p_{bid}(S)$$

$$\max_{S'}\{p_{ask,i}(S') - v(S',i)\} = p(i)$$

We get

$$\sum_{k \in S} a(S,k)p(k) - v(S,i) + 2\epsilon \geq p(i) \tag{10}$$

Summing Equation 6 over all agents in the final allocation and with $p(j) = 0$ for agents not in the allocation, we get

$$\sum_{j \in N} p(j) \leq \sum_{j \in N} v(S,j) - \sum_{j \in N} \sum_{k \in S_j} a(S_j,k)p(k) + 2\epsilon \min\{|G|, \max\{|M|,|N|\}\} \tag{11}$$

Summing Equation 10 over all agents in the final allocation and with $p(i) = 0$ for agents not in the allocation, we get

$$\sum_{i \in I} p(i) \leq \sum_{i \in M} \sum_{k \in S_i} a(S_i,k)p(k) - \sum_{i \in M} v(S,i) + 2\epsilon \min\{|G|, \max\{|M|,|N|\}\} \tag{12}$$

Adding Equations 11 and 12, we get

$$\sum_{i \in M} p(i) + \sum_{j \in N} p(j) \leq \sum_{j \in N} v(S,j) - \sum_{i \in M} v(S,i) + 4\epsilon \min\{|G|, \max\{|M|,|N|\}\} \tag{13}$$

The LHS is the value of the final dual solution V_{DLP} and the first two terms on the RHS is the value of the final primal solution V_{LP}. Therefore

$$V_{DLP} \leq V_{LP} + 4\epsilon \min\{|G|, \max\{|M|,|N|\}\} \tag{14}$$

We know

$$V_{LP}^* \leq V_{DLP} \tag{15}$$

Therefore, we have

$$V_{LP}^* \leq V_{LP} + 4\epsilon \min\{|G|, \max\{|M|,|N|\}\} \tag{16}$$

$$\Rightarrow V_{LP} \geq V_{LP}^* - 4\epsilon \min\{|G|, \max\{|M|,|N|\}\} \tag{17}$$

Because the primal solution we get using our algorithm is integral, we have the following bound

$$V_{LP}^* \geq V_{IP}^* \geq V_{IP} = V_{LP} \geq V_{LP}^* - 4\epsilon \min\{|G|, \max\{|M|,|N|\}\} \geq V_{IP}^* - 4\epsilon \min\{|G|, \max\{|M|,|N|\}\} \tag{18}$$

3 Iterative Auction Scheme for Combinatorial Exchanges

Iterative auction is a progressive or dynamic mechanism which involves a number of iterations before allocating objects to buyers and sellers. The exchange or auctioneer is the coordinating agent who iteratively updates the prices of objects starting from initial prices. During the progress of the auction, in each iteration, agents submit their revised bids and asks according to the declared prices. The market maker collects the new bids, computes and announces the updated prices, then proceeds with the next iteration. The goal of price updates is to drive the exchange towards market clearing prices.

In this mechanism a supply-demand conflict exists if the number of units of an item demanded is more than the number of units of that item supplied. In each iteration we have to solve NP-hard problems. The buyers' problem is a set covering problem, and the sellers' problem is a set packing problem. We can use greedy algorithms to solve the set packing and set covering problem in every iteration.

3.1 Forward Auction

The exchange would like to allocate or sell the goods to the bidders to maximize revenue. The corresponding integer programming formulation is given by:

$$\begin{aligned}
V^* &= \max \sum_{j \in N} \sum_{S \subseteq G} v(S,j) y(S,j) \qquad (19)\\
s.t. \quad & \sum_{S \ni k} \sum_{j \in N} a(S,k) y(S,i) \leq A_k \;\; \forall k, \; \forall S \subseteq G \\
& \sum_{S \subseteq G} y(S,j) \leq 1 \;\; \forall j \in N \\
& y(S,j) = 0,1 \;\; \forall S \subseteq G, \; \forall j \in N
\end{aligned}$$

where:
A_k = Total availability of item k

3.2 Reverse Auction

The exchange would like to minimize the procurement cost. The integer formulation for this is given by:

$$\begin{aligned}
V^* &= \min \sum_{i \in M} \sum_{S \subseteq G} v(S,i) y(S,i) \qquad (20)\\
s.t. \quad & \sum_{S \ni k} \sum_{i \in M} a(S,k) y(S,i) \geq D_k \;\; \forall k, \; \forall S \subseteq G \\
& \sum_{S \subseteq G} y(S,i) \leq 1 \;\; \forall i \in M \\
& y(S,i) = 0,1 \;\; \forall S \subseteq G, \; \forall i \in M
\end{aligned}$$

where:
D_k = Total demand for item k

3.3 Greedy Algorithms for Combinatorial Auctions

A Greedy Algorithm for Set Covering Problem:
The forward combinatorial auction is actually a set packing problem. The following is a greedy algorithm for the set packing problem given by Chvátal [3]:

1. Set $\mathcal{R} := \phi$ and $W := \phi$.
2. While $W \neq U$ do:
   Choose a set $R \in \mathcal{S} \backslash \mathcal{R}$ for which $\frac{c(R)}{|R \backslash W|}$ is minimum.
   Set $\mathcal{R} := \mathcal{R} \cup R$ and $W := W \cup R$.

A Greedy Algorithm for Set Packing Problem:

The reverse combinatorial auction is actually a set covering problem. The following is a greedy algorithm for this problem given by Lehmann et al [7]:

The algorithm is executed in two phases.

1. In the first phase, the bids are sorted by some criterion i.e. a norm is defined and the bids are sorted in decreasing order following this norm.

2. In the second phase, a greedy algorithm generates an allocation. Let L be the list of sorted bids obtained in the first phase. The first bid of L, say $< S, V(S) >$ is granted, i.e. the set S is allocated at price $V(S)$ and then the algorithm examines each bid of L, in order, and grants it if it does not conflict with any of the bids previously granted. If it does, it denies, i.e. does not grant, the bid.

The best known bound for the packing problem [7] is a factor of $\sqrt{|M|}$ which can be achieved by a greedy allocation scheme with norm $\frac{V(S)}{\sqrt{|S|}}$.

3.3.1 Calculation of A_k and D_k

In Equations 19 and 20, A_k and D_k are not known. Since we are using greedy algorithms, we can calculate A_k and D_k from the solution obtained by the greedy heuristics.

3.3.2 Myopic Bidders

We assume that the bidders, both buyers and sellers, follow a myopic best response strategy. This means that the agents are myopically rational and they play their utility maximizing response to the current prices in the auction.

3.3.3 Approximate Winner Determination

An approximate winner determination algorithm must satisfy bid monotonicity property to maintain the same incentives for myopic agents to follow the same bidding strategy as when an exact winner determination algorithm is used [9].

The greedy algorithms give feasible integer solutions to the set packing and covering problems. Also the greedy solutions satisfy the bid monotonicity property.

3.3.4 Price Update

We generalize the price update rule of an English auction for the price update rule for our iterative scheme for combinatorial exchanges. Suppose we start with an initial price $p(k) = 0$ for each item k. We update the price of an item k if:

$$A_k < D_k$$

The price of the item k is increased by ϵ from the current price.

3.4 Termination Conditions

Optimum solution to the exchange is obtained when all the complementary slackness conditions are satisfied.

3.4.1 Complementary Slackness Conditions

1. CS-1 means that all seller agents must only receive bundles which maximize their utility at current prices. Since bundles are allocated according to the best response bids of the seller agents, therefore CS-1 is satisfied throughout the auction.

2. Similarly CS-2 is satisfied throughout the auction for buyer agents.

3. CS-3, CS-4, CS-5 are satisfied only at the end of the auction.

The complementary slackness conditions could also be interpreted as follows:

1. Maximum revenue for the sellers i.e. optimal solution to Eq. 19.

2. Minimum procurement cost for the buyers i.e. optimal solution to Eq. 20

3. The supply and demand constraint is satisfied :

$$\sum_{S \ni k} \sum_{i \in M} a(S,k)y(S,i) \geq \sum_{S \ni k} \sum_{j \in N} a(S,k)y(S,j) \ \ \forall k \ \ \forall S \subseteq G$$

$$\Rightarrow \ \ A_k \ \geq \ D_k \ \ \forall \ k$$

Since we solve for optimum (or approximately optimum) solution for both buyers' and sellers' problem in each iteration, therefore the iterative auction mechanism terminates when the supply and demand constraint is satisfied. The termination prices are therefore the market clearing prices.

3.5 Iterative Auction Algorithm

1. Start with initial prices $p_k = 0 \ \ \forall k$.

2. Collect the bids and asks.

3. Solve the sellers' and buyers' problems Eq. 19 and 20 using the greedy heuristics. Check for the termination conditions. End the auction if the termination conditions are satisfied. Else go to Step 4

4. Update the item prices.

5. Go to Step 2.

We can also have an iterative auction algorithm where the auction begins with very high initial prices and the prices are decreased in every iteration.

3.5.1 Convergence of the Algorithm

The price updates in the algorithm are monotonically nondecreasing. And since no bidders have infinite valuation for any subset, therefore we get a monotonic increasing sequence which is bounded above. Therefore, the algorithm converges in a finite number of steps.

4 Conclusion

We have introduced an iterative auction scheme for the combinatorial exchange problem. But we have not addressed the issue of incentive compatibility and Vickrey prices. We need to design incentives so that the agents' dominant strategy is to bid truthfully. We also need to study the effect of variable price updates. Also we need to generate test data for combinatorial exchanges to study the solutions of these iterative mechanisms and compare these solutions to the actual optimal solutions.

References

[1] M. Bichler. *The Future of E-Markets: Multi-Dimensional Market Mechanisms*. Cambridge University Press, 2001.

[2] S. Bikhchandani and J.M. Ostroy. The package assignment model. Technical report, Working Paper, University of California, Los Angeles, June 2001.

[3] V. Chvátal. A greedy heuristic for the set cover problem. *Mathematics of Operations Research*, 4:233–235, 1979.

[4] M. Eso, J. Kalagnanam, L Ladanyi, and Y.G Li. Winner determination in bandwidth exchanges. Technical report, RC22219 (W0110-087) IBM TJ Watson Research Center, 2001.

[5] J. Kalagnanam, A.J. Davenport, and H.S. Lee. Computational aspects of clearing continuous call double auctions with assignment constraints and indivisible demand. *Electronic Commerce Research*, 1:221–238, 2001.

[6] A. Kothari, T. Sandholm, and S. Suri. Solving combinatorial exchanges: Optimality vis a few partial bids. In *AAAI*, 2002.

[7] D. Lehmann, L.I. O'Callaghan, and Y. Shoham. Truth revelation in approximately efficient combinatorial auctions. Technical report, Working Paper, Hebrew University, Jerusalem, 2001.

[8] R. B. Myerson and M. A. Satterthwaite. Efficient mechanisms for bilateral trading. *Journal of Economic Theory*, 28:265–281, 1983.

[9] D. Parkes. *Iterative Combinatorial Auctions: Achieving Economic and Computational Efficiency.* PhD thesis, Department of Computer and Information Science, University of Pennsylvania, may 2001.

[10] D. C. Parkes, J. Kalagnanam, and M. Eso. Achieving budget-balance with Vickrey-based payment schemes in combinatorial exchanges. Technical report, IBM Research Report RC 22218 W0110-065, 2001.

[11] R. Vohra and S. de Vries. Combinatorial auctions, a survey. *INFORMS Journal of Computing*, 2003.

[12] W. E. Walsh, M. P. Wellman, and F. Ygge. Combinatorial auctions for supply chain formation. In *ACM Conference on Electronic Commerce*, pages 260–269, 2000.