CoDABot : An Agent-Based Continuous Double Auction System

[a] Yong-Gyu Jung, [b]Kwang-Su Kim, [b]In-Cheol Kim

Department of Computer Science, Kyonggi University
San 94-6, Yiui-Dong, Paldal-Gu, Suwon, 442-760, KOREA
Tel : +82-31-249-9669, Fax : +82-31-249-9673
Email : [a] ygjung@shjc.ac.kr, [b]{suddol, kic}@kuic.kyonggi.ac.kr

Abstract

Most of current Internet auction systems are simple server programs that partly automated the function of the conventional auction house. These systems do not provide sufficient independence, distribution and parallelism between the functions of the conventional auction house. In these auction systems, only part of auction management functions are automated by server program, but still users need to execute a repetitive task to monitor the dynamic progress of a certain auction, decide proper bid price, and submit the bid. Another problem is that they support only single auctions such as English auction and Dutch auction, but they do not support the double auctions that are superior to the single auction with respect to speed, efficiency, and the fair distribution of profit. In this paper, we present design and implement of an agent-based continuous double auction system, called CoDABot, in order to mitigate the limitations of current auction systems. CoDABot supports continuous double auction, provides various bidding agents for users to select, and has been implemented into a multi-agent system that realize more independent, distributed, and parallel subsystems.

Keywords:

CoDABot, multi-agents, continuous double auction

1. Introduction

The increasing omnipresence of computers and Internet infrastructures induced the conventional markets to use on-line electronic commerce systems. Through the Internet, we could overcome the time and space barriers and have a wide variety of goods available on-line. At any time, we can participate in the markets only by connecting to on-line systems instead of visiting real markets. While the on-line auction system is a type of application in the domain of electronic commerce, it could negotiate through the bidding the price of buyers and sellers [9] [13]. But current on-line auction systems have some limitations.

First, these systems do not provide sufficient independence, distribution and the parallelism between the functions of the conventional auction house. It is caused by the fact that system employs a single server program.

Secondly, we need to visit the on-line auction market repeatedly for bidding price even though the systems are operated automatically in management and processing. We need to be informed of the progress status of auction and check the current price and others. Therefore we should keep the line connected, which increase network traffics and system loads.

Finally, these systems support only single auctions such as English auction or Dutch auction, while double auction proved to be superior to single auction in terms of speed, efficiency, and fair profit [2]. Therefore the current systems are considered to be less efficient, and incur much difficulties in sharing profits among all participants appropriately.

We introduce an agent-based continuous double auction system called CoDABot [7]. In this paper, we try to address the problems of current on-line auction systems. Our system supports the CDA (Continuous Double Auction), which also consists of various auction types. Also the system could be operated automatically with multi-agents consisting of user interface agents, auctioneer agents, and bidder agents.

2. Related Works

2.1 Agent-based Auction Systems

There are many agent-based on-line auctions that are currently used. Some sites run by discounters selling multiple goods, others open to the public and any can sell (Person-to Person auctions). Auction is a type of market in which sellers and buyers propose their prices, and a price could be determined. The types of auction are said to be single and double auction. Figure 1 shows sealed bid and outcry auction which are characterized by auction methods, and ascending and descending auction which are differentiated the bid ordering [6] [9] [10] [12].

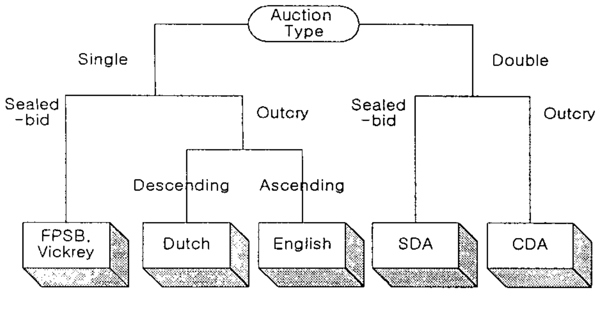

Figure 1 - Taxonomy of Auction Types

There are several agent-based virtual marketplaces that use auction protocols for negotiation. Of these, it is felt that the most similar systems to agent-based auction are AuctionBot, eMediator and Fishmarket.

The AuctionBot [10] implements a server-side auction whose participants can configure their strategies through an API that is accessible through a web interface. It supports the English auction, Vickrey, periodic double auction (PDA) [15]. The scheduler, daemon process constantly monitors the database. It introduces Mth price algorithm [6] [9]. 'M' denotes the total number of bidding counts of buyers and sellers. Among these bidding prices, the best price could be Mth bidding price. The system clones three agents; a competing agent, price modeling agent and bidder modeling agent.

The eMediator [12] system of Washington University supports combinatorial auction. It is possible one of various auction types with five bidder agents in eAuctionHouse. Contract optimizer determines clearing price with artificial intelligent technologies and game theories. An agent called eExchangeHouse assists a user in exchanging goods.

The Fishmarket [11] system also differentiates buyers from sellers, although the auction is not conducted through the web. A significant feature of Fishmarket is that its conventions are readily customized. This has been used to simulate a real world Fishmarket where the auctioneer chooses what to offer and invites bids from interested parties. It supports knowledge query and manipulation language (KQML). KQML is an agent communication protocol and parallel virtual machine (PVM) to reduce network congestion resulting from the increase in the number of agents [3].

2.2 Continuous Double Auction

The double auction denotes the system that buyers and sellers negotiate the prices of homogeneous goods. In synchronous double auction, the clearing price with collecting bids is decided at specified interval of time. In asynchronous double auction, sellers and buyers consecutively submit bids and then clearing prices are determined likewise.[1] [4] [10] [15]

The mechanism of clearing price in a continuous double auction can be explained with well-known supply and demand curves. In general, higher price induces less demand. At varying prices and quantities in any market, buyers and sellers decide to buy and sell any of these goods as any of each price.

We characterize price discovery mechanisms. Figure 2 shows the supply and demand curves. The intersection point in the above curves means the agreed price, which means that sellers and buyers want to supply and demand as same price and same quantities. Hence, they are willing to trade at this equilibrium price P_0 and an equilibrium quantity Q_0. In the continuous double auction, sellers' quote price is matched constantly against buyers' proposed prices, which corresponds to the shaded area in Figure 2 [1]. At the result, the continuous double auction is superior to the single auction in speed, efficiency and the benefits to users [2].

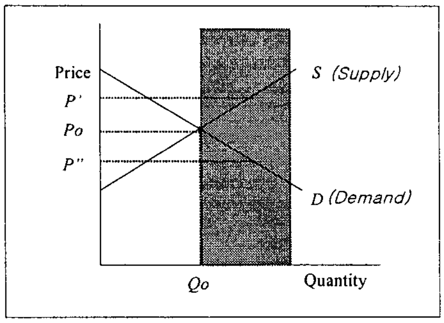

Figure 2 - Supply and Demand Curves

3. The CoDABot System Design

3.1 System Architecture

The fundamental elements of CoDABot system architecture is distributed set of agents that can support variety auction types. They are user interface agents and auctioneer agents. Other agents are specialized and instantiated by these agents. Through the user interface agents, all participants could initiate an auctioneer agent and acquire/check the bidding data in the database. The auctioneer agent monitors auction proceeds. The auctioneer agent creates bidder agents for the purpose of representing auction participants. The task of the auction agent is to determine the clearing price and the winner during the bidding process. Through the bidder agent, the bidding data is submitted to the auctioneer agent and the bidding information is notified to the auction participants by e-mail. Figure 3 shows the CoDABot architecture.

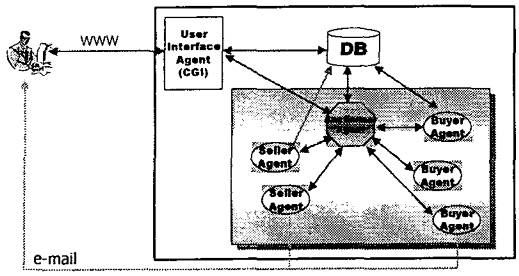

Figure 3 - The CoDABot System Architecture

3.2 Interaction with Agents

Auctions are simple and robust mechanisms for selling non standard and short supply items with uncertain market values. Auction mechanisms discover the optimal price for a good through the bidding action of self-interested agents [8]. These agents are continuously communicating with other agents for getting their goals. now, we introduce the message-based interactions for negotiation with other agents. It is said that CoDABot system is automatically operated by these agents' interactions. Furthermore, when we consider the trust properties of auctions, there are various steps in using auction system; registration, login and out, creation, participation and withdrawal from/to auction system and so on. It is through the user interface agent that the user could interact with other auctioneer agent and bidder agent utilizing various messages.

Figure 4 shows the interactions between auctioneer agent and bidder agent.

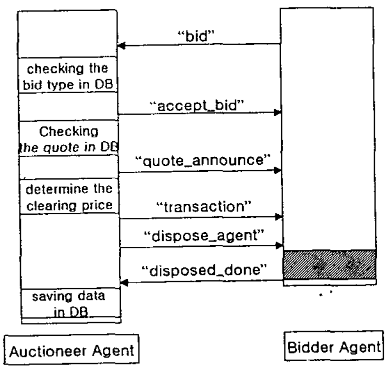

Figure 4 - The interaction between agents

The bidder agent submits bidding data to the auctioneer agent. The auctioneer agent checks the bid type and then notifies an "accept_bid" message to the bidder agent as return. If the bid is a quote, the auctioneer agent multicasts a "quote_announce" message to all of the bidder agents. The auctioneer agent determines the clearing price and the winner and then multicasts "transaction" message announcing the clearing price, the winner agent and the type of goods sold. And the auctioneer agent notifies the winner agent to a "dispose_agent" message. When the winner agent receiving a "dispose_agent" message is disposed, sends a "disposed_done" message to the auctioneer agent.

3.3 Bidder Agents

The design of bidder agents should be appropriate to the auction mechanism, and sensitive to the cost characteristics of users. An autonomous agent that places bids up to the value of fixed reservation price. The CoDABot supports five bidder agents for a auction participant to choose from. They include the simple agent, the incremental and decremental bidder agent, the target-oriented bidder agent, the clearing-price modeling agent and the time-bound bidder agent. Each agent could be created in the auction server bids on behalf of participants.

First, an auction participant takes bidding directly with the simple bidder agent, which has no information to help its client. The auction participant submits bid price to the simple bidder agent and then the agent sends the price to the auctioneer agent. Figure 5 shows the simple bidder agent and the user interface agent.

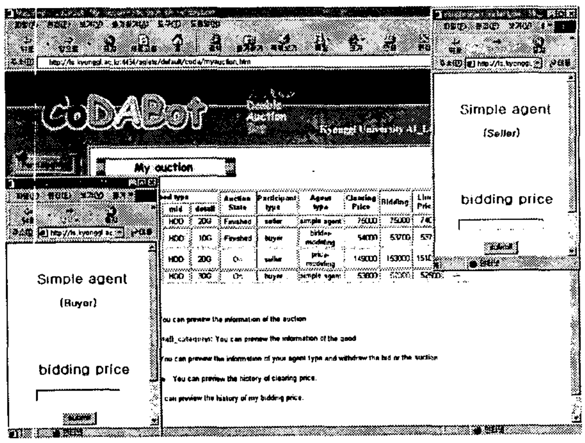

Figure 5 - Simple Agents with the User Interface Agent

Second, the incremental or decremental bidder agent bids the price based on quote price until it reaches the limit price.

- P_b : the expected bid price
- P_{limit} : the limit price
- Q^b : the quote price of the buyer
- Q^s: the quote price of the seller
- α : the percentage that user defined

$$P_b = \min\left(Q^b + \frac{|P_{limit} - Q^b|\alpha}{100},\ P_{limit}\right)(buyer)$$
$$P_b = \max\left(Q^s - \frac{|P_{limit} - Q^s|\alpha}{100},\ P_{limit}\right)(seller) \qquad (1)$$

Third, the target-oriented bidder agent determines the bid price based on the specific items or buyer.

- P_b' : the latest bid price
- P_{target} : the bid price of the targeted agent

$$P_b = \min\left(Q^b + \frac{|P_{limit} - Q^b|\alpha}{100},\ P_{limit}\right)(buyer) \qquad (2)$$

$$P_b = \max\left(Q^s - \frac{|P_{limit} - Q^s|\alpha}{100},\ P_{limit}\right)(seller)$$

Fourth, the clearing-price modeling agent determines bid price. In other words, it bids the price based on the average of the past clearing price.

- $CP_{average}$: the average of the past clearing price

$$P_b = \min\left(Q^b + \frac{|P_{limit} - Q^b|\alpha}{100},\ P_{limit}\right)(buyer) \qquad (3)$$

$$P_b = \max\left(Q^s - \frac{|P_{limit} - Q^s|\alpha}{100},\ P_{limit}\right)(seller)$$

Finally, the time-bound bidder agent is similar to the clearing-price modeling agent. But the difference is that it determines the bid price based on the valid time of the agent. If the agent is not determined on the winner agent in the valid time, it will bid highly based on the valid time.

- V_t : the valid time
- P_t : the present time

$$P_b = \min\left(Q^b + \frac{|CP_{average} - Q^b|\alpha}{100(V_t - P_t)},\ P_{limit}\right)(buyer) \qquad (4)$$

$$P_b = \max\left(Q^s - \frac{|CP_{average} - Q^s|\alpha}{100(V_t - P_t)},\ P_{limit}\right)(seller)$$

3.4 Database Design

CoDABot system contains a real-time transaction-centric object model. Also the agents can interact with others by viewing centric database. The object model in the database was designed as Figure 6. Each semantic object is saved as a relation (table) in the relational database management system. The Database has 10 tables as follows.

- User table
- Participation table
- Auction table
- Agent table
- Good type table
- Goods table
- Requirement table
- Bid table
- Offer table
- Transaction table

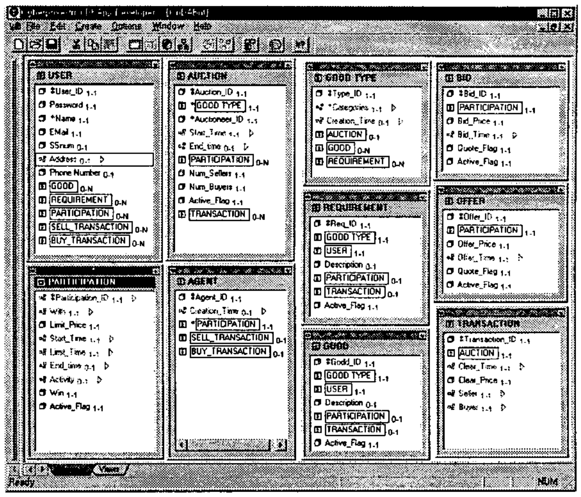

Figure 6 - Semantic Object Model

4. System Implementation

The CoDABot system is implemented on Unix of HP_UX 10.20 with Java 1.1.7 , Oracle server 8.0 and IBM aglet workbench (AWB) 1.0.2. For monitoring the status information functions, it gives screen in GUI. The user interface agent in the CoDABot displays the auction list. It has the information of auction ID, the type of goods, the quote price, etc. We need to create a new auction or to participate in a progressive auction in the auction list. To create a new auction, user has to submit a new type of goods. To participate in a progressive auction, the user needs various information including user ID, auction ID, the role of agent (seller or buyer), the agent type (of five bidder agents), the limit price and the valid time of participant. Messages for the auctioneer agent are sent through the user interface agent. Then each bidder agent checks the bids of price in the auctions server. Figure 7 shows the private auction information submitted by participants through the user interface agent.

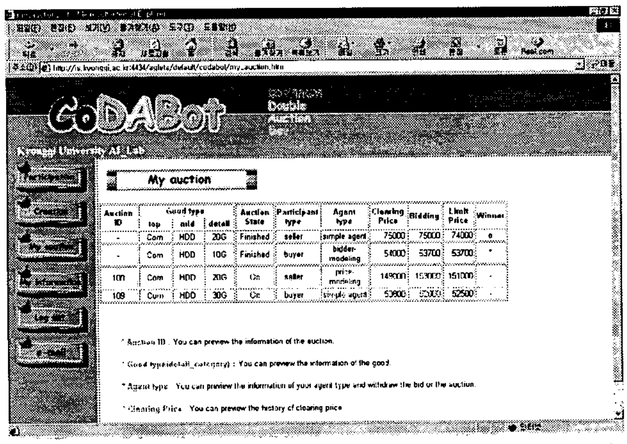

Figure 7 - Auction Status Information

5. Conclusion

On-line electronic commerce provides a cheap and readily available communication infrastructure. But largest overhead to user participation in an on-line auction is deciding what to buy, and how much it is worth. It is a much greater problem on-line than off-line because there are many different goods available across many different auctions.

Trying to overcome these problems, our system gives an agent-based continuous double auction mechanism. It provides various bidder agents on behalf of participants. It is implemented with multiple of user interface agent, auctioneer agents, and bidder agents for providing sufficient independence, distribution and parallelism. It supports the continuous double auction of various auction types. The continuous double auction (CDA) is superior to the single auction in terms of speed, efficiency, and the profit of users. Users find it easy to participate in the auction with various user interfaces.

People have different goals, Knowledge, preferences, constraints, influences, and attitudes during any given shopping experience.[5]. We tried to include these factors in bidding multi-agents by interacting with other agents. But our proposed agent-based auction system is to approach to negotiate in limited participants needs. As to our future work, we will head towards the construction of actual agents capable of negotiation in actual auction markets under the rules of any various participants' factors. And then, our system will be evaluated in various viewpoints.

Reference

[1] Cason, Timothy N. and Daniel Friedman, "Price formation in double auction markets", Journal of Economics Dynamics and Control vol. 20. 1307-1337, 1996.

[2] Cliff, Dave, "Evolving parameter sets for adaptive trading agents in continuous double-auction markets", In Agent-98 workshop on Artificial Societies and Computational Markets, 38-47, 1998.

[3] Finin, T. and Y. Labrou. "KQML as an Agent Communication Language", MIT Press, Cambridge, Mass., p291-316, 1997.

[4] Friedman, Daniel and John Rust, "The Double Auction Market: Institution, Theories and Evidence", Addison-Wesley Publishing, Reading, MA, 1993.

[5] Guttman, Robert H. and pattie Maes, " Agent-mediated Integrative Negotiation for Retail Electronic Commerce", In Lecture Notes in Artificial Intelligence, Springer, Minneapolis, MN, USA, May 10th, 1998.

[6] Hu, Junling, Daniel Reeves and Hock-Shan Wong, "Agent Service for On-line Auctions", Proceedings of the AAAI-99 workshop on AI for Electronic Commerce, AAAI Press, 1999.

[7] Kim, Kwang-Su and In-Cheol Kim, "Agent-based Continuous Double Auction System", Proceeding of The 27th Korea Information Science Society Fall Conference, p113 ~ 115, 2000.

[8] Parkers, David C., Lyle H. Ungar, and Dean P. Foster, "Accounting for Cognitive Costs in On-line Auction Design", In Lecture Notes in Artificial Intelligence, Springer, Minneapolis, MN, USA, May 10th,1998.

[9] Preist, Chris, "Commodity Trading Using An Agent-Based Iterated Double Auction", Proceeding of the Third International Conference on Autonomous Agents Seattle, WA USA, Ma 1-5, 1999.

[10]Razan, Vuay, James R. Slaglet, John Dickhaut and Aruit Mukherjt, "Decentralized Problem Solving Using the Double Auction Market Institution", Expert Systems with Applications, vol 12. p1-10, 1997.

[11]Rodriguez, J. A., P. Noriega, C. Sierra and J. Padget. "A Java-based Electronic Auction House.", Proceedings of PAAM '97, p207-224, April 1997.

[12]Sandholm, Tuomas W., "eMediator: A Next Generation Electronic Commerce Server", AAAI Workshop on AI in Electronic Commerce, Orlando, FL, July. AAAI Workshop Technical Report, p46-55, 1999.

[13]Wellman, M. P. and Wurman P. R. "Market-aware agents for a multi-agent world", Robotics and Autonomous Systems, 1998.

[14]Wurman, P. R., M. P. Wellman and William E. W., "The Michigan Internet AuctionBot: A Configurable Auction Server for Human and Software Agents", In Second International Conference on Autonomous Agents, p301-308, Minneapolis, 1998.

[15]Wurman, P. R., William E. W. and M. P. Wellman, "Flexible Double Auctions for Electronic Commerce: Theory and Implementation", 1998.