

JCAT: A Software Tool for Agent-Based Computational Economics

Jinzhong Niu

City University of New York

1 Introduction

Auctions, when well designed, result in desirable economic outcomes and have been widely used in solving real-world resource allocation problems, and in structuring stock or futures exchanges. The field of auction mechanism design has drawn much attention in recent years from economists, mathematicians, and computer scientists. In traditional auction theory, auctions are viewed as games of incomplete information and traditional analytic methods from game theory have been successfully applied to some simple types of auctions. However, the assumption of prior common knowledge in the incomplete information approach may not hold in some auctions, and computing analytic solutions may be infeasible in other auctions. Both of these problems hold in the case of *continuous double auctions*¹.

As a result of these problems, researchers often use computer simulation of auctions in which traders are software agents. Such agents, armed with various learning algorithms and optimization techniques, have been shown to produce outcomes similar to those observed in auctions with human subjects [6]. Indeed, software traders are capable of outperforming human traders [3]. Along with the automation of traders, computer scientists have started to take evolutionary and adaptive approaches to automatically creating auction mechanisms [1, 10]. Although this work has produced promising results, it has one common theme — the only comparisons that are made are indirect. The results from one lone market are compared with those of another lone market. In contrast, in real markets not only do traders in an auction compete against each other, but real market institutions compete against each other. In addition, existing work usually compares auction mechanisms in different settings which vary according to the availability of information, computational resources, and so on. The conclusions of these studies are thus difficult to compare and cumulate. It is therefore desirable to have a platform that allows multiple markets to compete against

each other, and allows market mechanisms to be evaluated in a uniform way. The JCAT² system that we introduce in this paper addresses these concerns.

JCAT extends an earlier version of Java Auction Simulator API, (JASA)³, adding support for multiple parallel markets with trading agents moving between them. It has been used to conduct research on computational auction design [8, 7, 9] and has been successfully used as the game server in the Trading Agent Competition (TAC) Market Design Competition (CAT) 2007–2011.

2 What JCAT provides

JCAT provides the ability to run what we will call “CAT games”, each of which is an interaction between markets and traders. A typical CAT game consists of a CAT server and several CAT clients, which may be trading agents or specialists (markets). The CAT server works as a communication hub between CAT clients. A registry component records all game events and validates requests from traders and specialists. Various game report modules are available to process game events, calculate and output values of different measurements for post-game analysis.

A CAT game lasts a certain number of *days*, each day consists of *rounds*, and each round lasts a certain number of *ticks*, or milliseconds. The game clock in the game server fires events to notify clients of opening and closing of each day and round intervals.

Each trading agent is assigned private values for the goods it will trade. For buyers the private value is the most it will pay for a good. For sellers, the private value is the least it will accept for a good. The private values and the number of goods to buy or sell make up the demand and supply of the markets. Private values remain constant during a day, but may change from day to day, depending upon the configuration of the game server.

Each trading agent is endowed with a *trading strategy* and a *market selection strategy*. The first specifies how to

¹A continuous double auction involves both buyers and sellers, and both kinds of trader are allowed to make or accept an offer at time during the auction.

²<http://jcat.sourceforge.net/>.

³<http://jasa.sourceforge.net/>.

make offers, the second specifies which market to choose to make offers in. Trading strategies provided in JCAT include those that have been extensively researched in the literature and some of them have shown to work well in practice, e.g., ZI-C [6], RE [4], ZIP [2], and GD [5]. A typical class of market selection strategies treats the choice of market as an n -armed bandit problem where daily profits are used as rewards when updating the value function.

Specialists facilitate trade by matching offers and determining the trading price in an exchange market. Each specialist operates its own exchange market and may choose its own auction rules — the aim of the CAT competition is to create a specialist that optimizes a particular set of measures. Specialists may have adaptive strategies such that the policies change during the course of a game in response to market conditions for desired outcomes. JCAT provides a reference implementation of a parameterizable specialist that can be easily configured and extended to use policies regulating different aspects of an auction.

A specialist typically includes components that regulate aspects of its market. *Matching policies* define the set of matching offers in a market at a given time. *Quoting policies* determine the ask quote and bid quote, which respectively specify the upper bound for offers to sell and the lower bound for offers to buy that may be placed in the market at a given time. *Shout accepting policies* judge whether a request by a trader to place an offer in the market should be accepted or rejected. *Clearing conditions* define when to clear the market and execute transactions between matched offers. A *pricing policy* is responsible for determining transaction prices for matched ask-bid pairs. The decision may involve only the prices of the matched offers, or more information including market quotes. *Charging policies* determine the charges a specialist imposes on a trading day. A specialist can set its fees, or *price list*, which are charged to traders and other specialists who wish to use the services provided by the specialist. Each specialist is free to set the level of the charges for registration with a specialist, for making an offer, for completing a transaction, and so on.

References

- [1] D. Cliff. Evolution of market mechanism through a continuous space of auction-types. Technical Report HPL-2001-326, Hewlett-Packard Research Laboratories, Bristol, England, Dec. 2001.
- [2] D. Cliff and J. Bruten. Minimal-intelligence agents for bargaining behaviours in market-based environments. Technical Report HPL-97-91, Hewlett-Packard Research Laboratories, Bristol, England, Aug. 1997.
- [3] R. Das, J. E. Hanson, J. O. Kephart, and G. Tesauro. Agent-human interactions in the continuous double auction. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, Seattle, U.S.A., August 2001.
- [4] I. Erev and A. E. Roth. Predicting how people play games: Reinforcement learning in experimental games with unique, mixed strategy equilibria. *The American Economic Review*, 88(4):848–881, September 1998.
- [5] S. Gjerstad and J. Dickhaut. Price formation in double auctions. *Games and Economic Behavior*, 22(1):1–29, 1998.
- [6] D. K. Gode and S. Sunder. Allocative efficiency of markets with zero-intelligence traders: Market as a partial substitute for individual rationality. *Journal of Political Economy*, 101(1):119–137, 1993.
- [7] J. Niu, K. Cai, and S. Parsons. A grey-box approach to automated mechanism design. In van der Hoek, Kaminka, Lespérance, Luck, and Sen, editors, *Proceedings of the Ninth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, Toronto, Canada, 2010. Extended Abstract.
- [8] J. Niu, K. Cai, S. Parsons, E. Gerding, and P. McBurney. Characterizing effective auction mechanisms: Insights from the 2007 TAC Mechanism Design Competition. In Padgham, Parkes, Müller, and Parsons, editors, *Proceedings of the Seventh International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, pages 1079–1086, Estoril, Portugal, May 2008.
- [9] J. Niu and S. Parsons. Maximizing matching in double-sided auctions. In Ito, Jonker, Gini, and Shehory, editors, *Proceedings of the Twelfth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013)*, Saint Paul, Minnesota, USA, 2013. Extended Abstract.
- [10] S. Phelps, S. Parsons, E. Sklar, and P. McBurney. Using genetic programming to optimise pricing rules for a double auction market. In *Proceedings of the Workshop on Agents for Electronic Commerce*, Pittsburgh, PA, 2003.