Electronic Markets and Multiagent Systems

Alberto Sardinha
Instituto Superior Técnico, UTL, Portugal

Ruy L. Milidiú
Pontifícia Universidade Católica do Rio de Janeiro, Brazil

ABSTRACT

As the Internet helps mediate millions of transactions in electronic markets, research work on automated trading agents is helping humans improve their trading objectives (e.g., finding lower prices and improving delivery options). This chapter presents an overview of the research work on trading agents in the context of the Trading Agent Competition. The Trading Agent Competition is an annual event where researchers are generally interested in the following research questions: (i) how to design trading agents, (ii) how to evaluate these trading agents, and (iii) how do trading agents affect electronic markets. This research community has produced many research results that are based on state-of-the-art techniques from artificial intelligence, operations research, statistics, and other relevant fields.

INTRODUCTION

Agents [Wooldrige, 1999] are software programs that are capable of making decisions autonomously in order to achieve goals (normally defined at design time). The term "Multi-agent System" is generally used for a group of agents in an environment with resources, which have to collaborate or compete to achieve a common goal. Trading agents are software agents that pursue trading objectives and have to abide by the rules defined by the electronic markets.

As the Internet helps mediate an increasing number of electronic transactions, there is a growing interest in investigating the benefits of developing automated trading systems for electronic markets. Effective trading systems depend not only on services with reliable interfaces but also on decision making processes (such as which goods to buy, when to purchase such goods, and at what price). One of the main concerns of Artificial Intelligence (AI) [Russell & Norvig, 1995] is to develop techniques for decision making processes by autonomous agents. Hence, agent-based systems are probably one of the best paradigms for effectively automating the decision making processes of trading systems.

The Trading Agent Competition (TAC) [Arunachalam & Sadeh, 2005; Wellman et al., 2001] is an annual event that was created to promote and foster high quality research into the trading agent problem. The trading agent problem is a complex decision making problem where autonomous software agents have to negotiate goods to achieve trading objectives. Since 2000, the event has attracted over 120 entries from universities and research institutes, such as Carnegie Mellon

University, Harvard University, University of Michigan, University of Texas at Austin, Southampton University and SICS.

This chapter is organized as follows. Firstly, we introduce the main objectives of the Trading Agent Competition and short description of the top two scenarios. Secondly, we present an overview of the different architectures used by the trading agent developers. Finally, this section discusses some lessons learned and presents some concluding remarks.

TRADING AGENT COMPETITION

The Trading Agent Competition (TAC) [TAC, 2011] is an annual event that has been held since 2000. The main goal of this event is to encourage high quality research into the trading agent problem. The trading agent problem is a complex decision making problem where autonomous software agents have to negotiate goods to achieve trading goals. For example, a trading agent might have a trading goal to buy components in order to build a computer. Not only does the agent have to try to minimize the purchasing cost but also guarantee that all components are procured (since a computer with a missing component, such as memory or motherboard, is not ready to be assembled).

In the TAC community, the researchers are generally interested in the following research questions [TAC Association, 2011]:

(i) **How to design a trading agent?** Given an electronic market specification, a trading agent developer normally starts the design process with the identification of key modules that are going to help the agent pursue the trading goals. For example, almost every trading agent has a forecast module where future prices of goods are predicted. This module might help other modules such as a bidding module in order to buy goods and minimize procurement costs.

(ii) **How to evaluate a trading agent?** The evaluation of a trading agent is a key aspect of the development process of a successful agent. The primary motivation for organizing TAC was to create a common electronic market where developers can compare and evaluate their agents.

(iii) **How do trading agents affect electronic markets?** Another important goal of TAC is to evaluate how the competing agents affect an electronic market.

Since 2000, the competition has attracted research groups from around the world to put forth their best efforts at developing automated trading agents for a specific market scenario. The market scenarios in TAC are:

- **TAC Travel** [Wellman et al., 2001]– Trading agents have to assemble travel packages for customers, which express their preferences for various aspects of the trip. In order to assemble these travel packages, the agents bid on flight tickets, hotel reservations and entertainment tickets from simultaneous auctions. The agent must maximize the total satisfaction of the customers while minimizing the procurement costs.

- **TAC Supply Chain Management** [Arunachalam & Sadeh, 2005] – Trading agents have to manufacture PCs, win customer orders in a competitive market, and procure components in order to build and deliver PCs to customers. The agent's goal is to maximize its profit by generating revenue from the customer orders and minimizing costs from the procurement of components and other related costs.

- **TAC Ad Auctions** [Jordan & Wellman, 2010]- Trading agents have to bid to place ads in a sponsored search environment. The competing trading agents are advertisers that employ different bidding strategies, while the users and behavior of the search engine is simulated by a server. At the end of the competition, agents are evaluated based on the sales profits and click costs.

- **TAC Market Design** [Niu et al., 2008]– Agents compete in an electronic market by defining rules for matching buyers and sellers. By setting appropriate fees, the agent has the goal of maximizing its profit by attracting buyers and sellers.

- **Power TAC** [Ketter et al., 2010]– Trading agents are brokers in retail electric power markets. These brokers buy and sell energy through contracts with retail customers and wholesale market. The main goal of each agents is to earn a profit by negotiating these contracts.

The following sections are focused on the top two scenarios from TAC, namely TAC Travel and TAC SCM. These two scenarios have attracted over 120 entries from 60 teams in 21 countries (e.g., Carnegie Mellon University, Harvard University, University of Michigan, University of Texas at Austin, Southampton University and SICS).

TAC Travel

In TAC Travel, each agent has the goal of assembling travel packages from TACtown to Tampa during a 5-day period. The agent acts on behalf of eight clients, and the objective of the travel agent is to maximize the total satisfaction of its clients with the minimum expenditure. Travel packages consist of items, such as round-trip flight, a hotel reservation, and tickets to some entertainment events. Like most electronic markets, TAC Travel requires that agents make decisions under uncertainty, such as deciding to book hotel rooms without having flight tickets yet.

There are clear interdependencies in the TAC Travel scenario, since a traveler needs a hotel for every night between arrival and departure of the flight, and can only attend entertainment events during that interval. The three types of goods (flights, hotels, entertainment) are purchased in separate markets with different rules:

- ***Flight Ticket Market*** – Only one airline company (TACAir) operates flights between TAC Town and Tampa. While it operates only one in-flight on day 1 and one out-flight on day 5, all the other days have an in and out flight. This is due to the fact that each client must spend at least one night in Tampa. Flight tickets are sold in single seller auctions, and there is an independent auction for each day and direction (in or out). The flight quotes are updated every 10 seconds by a random walk, and the flight auctions close at the end of the game. Agents can buy flight tickets at anytime from these auctions at the posted price, but are not allowed to resell or exchange these goods with other agents.

- ***Market for Hotel Reservations*** - The hotel market has 16 available rooms per night in each hotel: *Towers*, a premium hotel, and *Shanties*, a lower quality lodging option. Hotel auctions are Standard English ascending multi-unit, except that they close at randomly determined times. A randomly chosen hotel auction closes at minute one of the game, and the other hotel auctions close randomly at each minute thereafter. Price quotes are only generated once per minute as ask prices; the ask price (*ASK*) is calculated as the 16^{th} highest price among all bid units. Agents can place bids for at least one unit at price of *ASK* + 1 (while the auctions are open), but they cannot withdraw or resell a bid. When an auction closes, the 16 highest bid units are declared winners and each bidder gets a hotel reservation at a price equal to the ask price.

- ***Market for Entertainment Ticket*** – The agents can buy and sell tickets for three types of entertainment events: Amusement Park, Alligator Wrestling, and Museum. The entertainment auctions are standard continuous double auctions such as a stock market, and there is an auction to each type of entertainment event on each day within the 5-day period. Agents receive an initial endowment of 12 tickets (partitioned between the days and types), and may bid to buy or sell the tickets in these auctions. The entertainment auctions close at the end of the game, and price quotes are updated in response to new bids.

The market demand is determined by 64 clients' preferences, which are randomly generated by probability distributions. Each client preference has the following components:

- *Ideal arrival and departure dates* within the 5-day period;
- A *premium value* for staying in the higher quality hotel;
- An *entertainment value* for each type of entertainment ticket.

The value of a travel package depends on how the goods are bundled (also known as *trips*) for each client. However, a *trip* is only considered feasible for a client iff: (i) the arrival date is strictly before the departure date, (ii) there is a hotel reservation for all nights between arrival and departure dates (and must be the same hotel throughout the trip), and (iii) there is at most one entertainment event per night. A travel package for each client is accrued by a baseline value of 1000 for each feasible *trip*, minus 100 for each day of deviation from the ideal arrival and departure dates, plus the bonuses for staying at the higher quality hotel and attending the entertainment events.

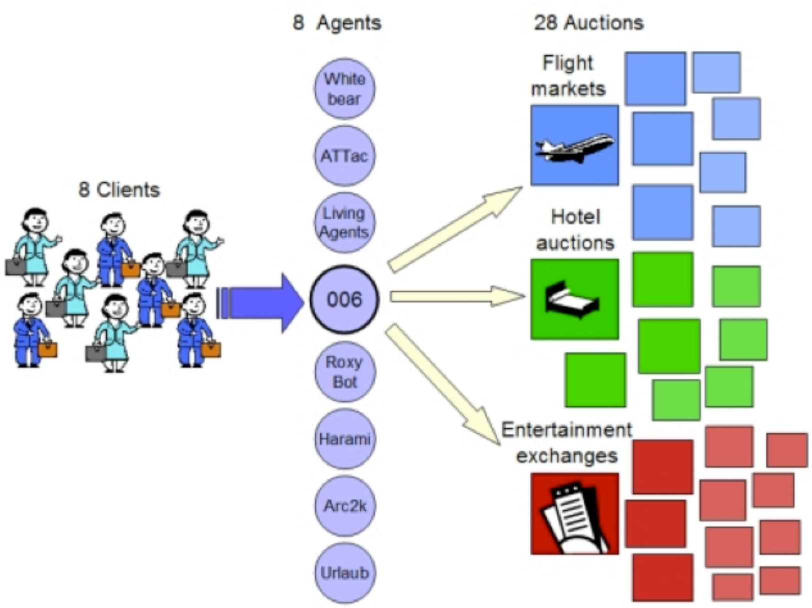

Figure 1: The TAC Travel Environment

Figure 1 shows the TAC Travel environment, where a server simulates the clients and 28 simultaneous auctions where the agents have to procure the travel goods. A game pits an entry against seven other trading agents developed by different research teams. The score of each agent in a game is the value of the travel packages minus the procurements costs. The game lasts nine minutes and several games are played during each round in order to evaluate each agent's average performance and to smooth the variations in client preferences. At the end of the competition, the agent with the highest average score is declared the winner.

TAC Supply Chain Management

TAC SCM is a simulation of a supply chain where six manufacturer agents compete with each other for both customer orders and components from suppliers. A server simulates the customers and suppliers, and provides banking, production, and warehousing services to the individual agents. Every game has 220 simulated days, and each day lasts 15 seconds of real time. The agents receive messages from the server on a daily basis informing them of the state of the game, such as the current inventory of components. The agents must send responses to the same server indicating their actions prior to the end of the day, such as requests for quotes from the suppliers. At the end of the game, the agent with the highest sum of money is declared the winner.

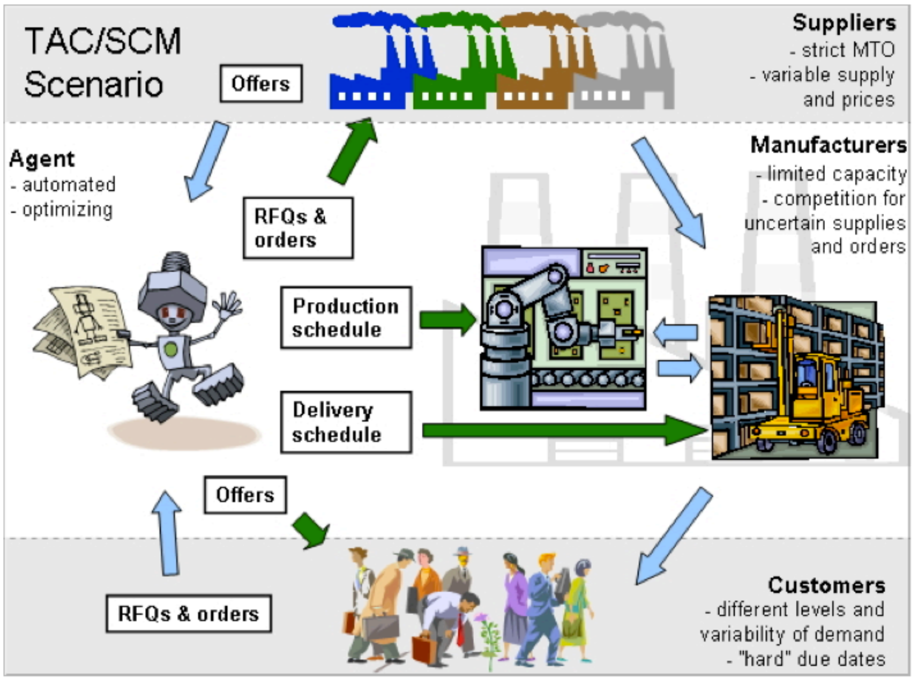

Figure 2: TAC SCM Environment

Typically, each manufacturer agent separates its decisions into the important sub-problems of a supply chain: procurement of components, production and delivery, and sales.

Procurement of Components

By using different combinations of components, each agent is able to produce and store 16 different computer configurations in its own production facility. These computers are made from four basic components: CPUs, motherboards, memory, and hard drives. There are a total of 10 different components: two brands and speeds of CPUs, two brands of motherboards, and two sizes of hard disks and memory. The game includes 8 distinct suppliers, and each component has a base price that is used as reference for suppliers making offers. Each PC type also has a base price equal to the sum of the base prices of its components.

Every day, agents can send requests for quotes (RFQs) to suppliers with a given reserve price, quantity, type and delivery date. A supplier receives all RFQs on a given day, and processes them together at the end of the day to find a combination of offers that approximately maximizes its revenue. On the following day, the suppliers send back to each agent an offer corresponding to each RFQ with a price, a quantity, and a delivery date. Due to capacity restrictions, the supplier may not be able to supply the entire quantity requested in the RFQ by the requested due date. In this case it responds by issuing up to two modified offers, each of which relaxes one of the two constraints:

- Quantity, in which case offers are referred to as partial offers.
- Due date, in which case offers are referred to as earliest offers.

The suppliers have a limited capacity for producing a component, and this limit varies throughout the game according to a mean reverting random walk. Moreover, suppliers limit their long-term commitments by reserving some capacity for future business. The pricing of components is based on the ratio of demand to supply, and higher ratios result in higher prices. Each day the suppliers estimate their free capacity by scheduling production of components previously ordered as late as possible. The manufacturer agents normally face an important trade-off in the procurement process: pre-order components for the future yielding lower prices but where customer demand is difficult to predict, or wait to purchase components at the last minute and risk being unsuccessful due to high prices or low availability.

A reputation rating is also used by the suppliers to discourage agents from driving up prices by sending RFQs with no intention of buying. Each supplier keeps track of its interaction with each agent, and calculates the reputation rating based on the ratio of the quantity purchased to quantity offered. If the reputation falls below a minimum value, then the prices and availability of components begin to deteriorate for that agent. Therefore, agents must carefully plan the RFQs that they send to suppliers.

Sales

The server simulates customer demand by sending customer requests for quotes (RFQ) to the manufacturer agents. Each customer RFQ contains a product type, quantity, due date, reserve price, and daily late penalty. Moreover, these customer requests are classified into three market segments: high range, mid range, and low range. Every day, the server sends a number of RFQs for each segment according to a Poisson distribution, with an average that is updated on a daily basis by a random walk. The total number of RFQs per day ranges between 80 and 320, and demand levels can change rapidly throughout the game. Thus, agents are limited in their ability to plan sales, production and procurement. The manufacturer agents respond to the customer RFQs by bidding in a first price sealed bid reverse auction: agent's cannot see competitors bids, and the lowest offer price wins the order. Agents do receive market reports each day that inform them of the highest and lowest winning bid prices for each PC type on the previous day.

Production and Delivery

Each manufacturer agent manages an identical factory, where it can produce any type of computer. The factory is simulated by the game server, and also includes a warehouse for storing components and finished computers. Each computer type requires a specified number of processing cycles, and the factory is limited to produce 2000 cycles (approx. 360 units) per day.

Each day the agent sends a production schedule to the game server. The simulated factory produces all the PCs in the schedule, as long as the required components are available. A delivery schedule is also sent to the server on a daily basis. It must specify the products and quantities of computers to be shipped to each customer on the following day. Only computers available in inventory can be shipped to customers. When a customer receives the PCs it ordered, the agent's bank account is credited with the payment equal to its bid price for the order times its quantity.

TRADING AGENTS

This section presents some of the top performing trading agents that have participated in the TAC Travel and TAC SCM events. We also present a generic bidding cycle for the two scenarios, which tackles sub-problems that all agents face when they are negotiating goods in the electronic markets. Some agents might frame the steps in this bidding cycle somewhat differently, but all agents tend to use most of these steps in the negotiation process.

TAC Travel Trading Agents

Agent developers in TAC Travel have proposed numerous different approaches to tackle sub-problems of trading interdependent goods in simultaneous auctions. Although each agent developer uses different techniques in the agent design, almost every agent is developed to solve two main sub-problems, namely the prediction of future prices and holdings of goods, and the problem of constructing and placing bids. In [Wellman et al., 2007], a generic agent cycle is presented to show the main decisions of an agent in the TAC Travel environment (as shown in Table 1).

```
While at least an auction of travel goods is open:
    Update prices of goods  and current holdings in the agent's knowledge base;
    Predict future prices and holdings of goods;
    Construct and place bids:
        Determine target holdings;
        Decide which target holding to bid on;
        Calculate bid prices;
```

Table 1: Trading Agent Main Cycle in TAC Travel

A typical bidding cycle lasts the time interval between the price quotes from the auctions (recall that flight price quotes are updated every 10 seconds, hotel quotes every minute, and entertainment quotes every time a new bid is received). Therefore, depending on the design, an agent might run the main cycle every 10 seconds. The main cycle starts with a step that collects prices of the goods in the market and current holdings. This important step updates the agent's knowledge with the current state of the market.

In TAC Travel, the agent receives price quotes and transaction notifications from the server upon request. The next step in the main cycle is to predict the prices and holdings of goods in the future. While some prices remain practically the same throughout the game, some prices might skyrocket (due to demand). The price prediction of quotes is an important task for agents that need to perceive the cost-benefit analysis of the goods.

With the information of current prices and holdings of goods together with the predictions, the agent now faces the problem of constructing and placing bids. This problem can be broken down into three decisions. First, the agent must decide what it wants to buy or sell (target holdings). This decisions is typically formulated as an optimization problem where agents maximize the difference between the value of the travel packages and the procurements costs. Second, the agent decides which of the target holdings it must bid now. For example, an agent might decide that it needs an in-flight on day 2, but will not buy the flight ticket now due to price predictions that indicate that prices are going down (so the agent might get a better deal if it postpones its action). Third, the agent must decide at what price it must place a bid for each good that it intends to buy now.

Agent	Affiliation	Prediction Strategy	Bidding Strategy
ATTac	AT&T Labs, Research	Boosting Algorithm	Integer Linear Programming (ILP) and marginal values
Walverine	University of Michigan	Competitive Equilibrium Analysis	Decision-theoretic formulation
Whitebear	Cornell University	Bayesian Analysis	Principled methods and Empirical knowledge
Roxybot	Brown University	Stochastic model	ILP formulation of SAA
Mertacor	Aristotle University Thessaloniki	Fuzzy reasoning and heuristic	Integer Linear Programming (ILP) and marginal values
SouthamptonTAC	University of Southampton	Fuzzy Logic	Integer Programming
LearnAgents	Pontifícia Universidade Católica	Historical and Maximum Likelihood technique	Integer Linear Programming (ILP) and marginal values

Table 2: Summary of Agent Strategies in TAC Travel

Table 2 summarizes the strategies employed by top performing agents in the TAC Travel competition. ATTac [Stone et al., 2001], a top performing agent in three of the first four TAC events, uses a boosting algorithm to predict probability distributions over prices, and an integer linear programming formulation to determine target holdings to bid on. This work also introduces the notion of marginal utility of goods, and used these values as a bidding strategy. Walverine [Cheng et al., 2005], a runner-up agent in 2004 and 2006, predicts point prices by utilizing an equilibrium analysis and a bidding strategy that uses a formulation that takes into account bids from other agents (assuming that they bid competitively) so that optimal bids can be constructed.

Roxybot [Lee et al., 2007], the top performing agent in the 2006 event, uses a stochastic model to predict probability distributions over prices, and a Integer Linear Programming formulation to solve a Sample Average Approximation (SAA) method of the bidding problem. Whitebear [Vetsikas et al., 2003], a top performing agent in the 2002 and 2004 events, predicts prices with a Bayesian Analysis, and the bidding strategy uses a combination of a randomized greedy strategy together with extensive empirical knowledge about the game. SouthamptonTAC [He et al., 2003], a runner-up agent in 2002, uses fuzzy logic to predict point prices and integer programming formulation in the bidding module.

Mertacor [Toulis et al., 2006], the top performing agent in the 2005 competition, uses a heuristic to predict flight prices and a fuzzy reasoning technique to predict hotel prices. LearnAgents [Sardinha et al., 2005], a third place entry in the 2004 event and finalist in three events, used historical data to predict the hotel prices and a maximum likelihood technique to predict flight prices. Both Metacor and LearnAgents use a integer linear programming formulation to determine target holdings to bid on, and also compute marginal utilities in the bidding strategy (using a similar strategy proposed by ATTac). LearnAgents also presents a multiagent architecture, as opposed to the single-agent strategy of the other agents, where each agent is responsible for solving the sub-problems presented in Table 1.

TAC SCM Trading Agents

Development teams of TAC SCM agents have proposed several different approaches for tackling important sub-problems in dynamic supply chains. Although each design presents different techniques, almost every agent developer tries to solve common sub-problems (this is not surprising, given the common goal that every agent pursues). Table 3 presents a generic agent cycle for TAC SCM that illustrates the main decisions of an agent in this supply chain environment.

At the start of each day, the agent must perform the following tasks:

Update information from the customer market and supplier market;
Predict future prices and demand trends of PCs and components;
Bid for customer orders;
Negotiate supply contracts;
Build a production schedule;
Ship orders to customers;

Table 3: Trading Agent Main Cycle in TAC SCM

An agent in TAC SCM has a day (15 seconds of real time) to perform the tasks presented in Table 3. The first task collects information about the market from the server, such as the request for quotes from the customers and supplier offers. This information is used by the next step, which predicts future prices of PCs and components, and forecasts demand trends of the business-to-consumer market (i.e., the consumer market) and the business-to-business market (i.e., the supplier market).

In the next step, the agent bids for customer orders in the business-to-consumer (B2C) market. Recall that customers send requests for quotes (RFQs) to the agents on a daily basis. All the agents respond to the customer RFQs by bidding in a first price sealed bid reverse auction, and the lowest offer price wins the order. Typically, the agent's bidding strategy in the B2C market tries to choose prices to maximize the agent's expected profit, while offering the amount of products that it can produce and deliver in the future.

The negotiation of supply contracts is an important task in the agent's main cycle, because there is a high interdependency between this task and other tasks. For example, the low availability of a given component might have a negative impact on the production of PCs (agents can only assemble PCs if all components are available) and consequently on the bidding strategy. Recall that the agent procures components by sending request for quotes (RFQs) to suppliers. On the following day, the suppliers send back to the agent the corresponding offers. Based on these offers, the agent can place orders for components. Typically, the agent's procurement strategy searches for offers with low prices, while trying to maintain high availability of components.

The scheduling strategy in the agent continuously maintains a production schedule over a horizon of several days. This schedule reflects current contracts, forecast contracts and projected component inventory levels. It helps drive other planning decisions including which customer RFQs to bid on and which RFQs to send to suppliers. A delivery schedule is created every day, and it must specify the products and quantities of computers to be shipped to each customer on the following day. Only computers available in inventory can be shipped to customers.

Table 4 presents the main strategy employed by the top performing agents in TAC SCM. Deep Maize [Kiekintveld et al., 2004], a top performing agent in the 2008 and 2009 events and a runner-up in the 2003 and 2010 events, uses a game theoretic analysis to factor out the strategic aspects of the environment and to define an expected profitable zone of operation. The agent uses market feedback to dynamically coordinate sales, procurement and production strategies in an attempt to stay in the profitable zone.

TacTex [Pardoe & Stone, 2006], a top performing agent in the 2005, 2006, and 2010 events, is built around machine learning techniques [Pardoe & Stone, 2005] for predicting the customer bid price probability distributions, and some heuristics for the procurement strategy. SouthamptonSCM [He et al., 2006], a runner-up agent in the 2005 event, uses fuzzy reasoning to compute bid prices on RFQs based on the current inventory levels, market demand, and current day in the game. RedAgent [Keller et al., 2004], a top performing agent in the 2003 event, uses an internal market architecture as the main decision mechanism with simple heuristic-based agents that individually handle different aspects of the supply chain process (such as the task of procuring components).

Agent	Affiliation	Main Strategy
Deep Maize	University of Michigan	Market feedback and game theoretic analysis
SouthamptonSCM	University of Southampton	Fuzzy reasoning
RedAgent	McGill University	Internal market heuristic
TacTex	University of Texas at Austin	Machine Learning + heuristics
CMieux	Carnegie Mellon University	Machine Learning + Greedy search
Botticelli	Brown University	Mathematical programming

Table 4: Summary of Agent Strategies in TAC SCM

CMieux [Benisch et al., 2009], a third place agent in the 2008 event and finalist in the 2007-2009 events, uses machine learning techniques to predict customer bid price probability distributions, component prices and demand trends in the B2B and B2C markets. The agent also uses a greedy procedure in the scheduling strategy and some heuristics in the procurement strategy. The Botticelli team [Benisch et al., 2009], a finalist in the 2003, 2004, 2006, 2008, 2009, and 2010 events, shows how the problems faced by TAC SCM agents can be modeled as mathematical programming problems, and used heuristic algorithms for bidding on RFQs and scheduling orders.

Competition and Agent Analyses

Several researchers have proposed different methods for analyzing data from the TAC markets. Such analyses give agent developers and market designers an interesting way to gain insights about agent performance and market rules. In [Wellman et al., 2007], an extensive analysis of the TAC Travel scenario is presented utilizing the game data from several competitions.

In order to perform analyses in the TAC SCM game data, there are several toolkits available on the Internet, such as the Analysis Instrumentation Toolkit [Benisch et al., 2005] and the Swedish Institute for Computer Science (SICS) Game Data Toolkit. These tools allow teams to analyze historical log files from a single TAC SCM game, and provide an in-depth view of the B2B and B2C interactions through graphical front-ends.

The team at the University of Michigan applied game theoretic analysis [Kiekintveld et al., 2006; Jordan et al., 2007a] to abstracted versions of the TAC SCM games. The results were generated empirically from simulations of different agent strategies, and reveals interesting best response and equilibrium relationships. For example, the analysis in [Kiekintveld et al., 2006] shows that the early procurement strategies of top performing agents in the 2004 TAC SCM competition had a positive contribution in the final results. In [Jordan et al., 2007b], a method is presented for estimating market efficiency and agent competency in the TAC SCM environment. The results show a significant increase in the overall market efficiency across competitions held on different years, but not across rounds in the same competition.

In [Borghetti et al., 2006], techniques are presented to manipulate the market environment of the TAC SCM simulator. By controlling various market factors, such as aggregate demand and supply, they suggest that TacTex loses its edge when market pressure is high. The analysis in [Andrews et al., 2009] presents a method for investigating the behavioral features that are associated with successful performance. The analysis was conducted on actual competition data, as opposed to offline controlled experiments in [Kiekintveld et al., 2006; Jordan et al., 2007a; Jordan et al., 2007b; Borghetti et al., 2006]. An interesting result from this work reveals that the top performing agents in the 2006 TAC SCM competition made purchases with longer lead times. The work in [Benisch et al., 2009] also analyzed actual competition data from the seeding rounds of the 2005 TAC SCM event. The results show that CMieux's strong performance is largely attributable to significantly cheaper component purchase prices than other agents.

Several teams have also analyzed controlled experiments using different configurations of their own agent and publicly available agent binaries. For example, the work in [He et al., 2006] presents experiments with variants of their own agent that are more or less risk seeking in choosing selling prices. This work also provides similar analysis with respect to lead times on component orders. In [Pardoe et al., 2006], a controlled experiments evaluates variants of the TacTex agent against publicly available binaries of other agents. They used the results of their experiments to fine-tune various parameters in their final agent and guide future development.

CONCLUSION

In this concluding section, we present some observations about the development process of trading agents in the TAC competition, and some lessons learned from the competition. The chapter starts with an argument that the TAC competition provides a unique insight into the problem of creating trading agents in electronic markets. Since 2000, a research community commenced an in-depth exploration of this topic in many different scenarios, which has led to many research results that are based on state-of-the-art techniques from artificial intelligence, operations research, statistics, and other relevant fields. Moreover, many researchers have proposed methods for analyzing the TAC markets and trading agents, so that agent developers and market designers can gain interesting insights about agent performance and market rules.

The development process of trading agents in the TAC markets share some key elements (as do most market domains). Trading agent have the main goal of negotiating goods to achieve trading objectives, so its not surprising that most trading agents have modules for predicting prices and optimizing bids. Several research innovations have been the direct result of the development of trading agents in TAC. For example, generic bidding heuristics presented in [Wellman et al., 2007] are a direct consequence of the development of agents such as ATTac and RoxyBot.

However, research based competitions can present some pitfalls, such as the problem of having entrants that try to win at the expense of producing scientific results (especially if a monetary prize is awarded to top performing agents). In order to avoid hazards, the TAC events are organized with many rules that try to prevent these pitfalls. For example, the winning agents are not awarded any monetary prizes. In the TAC event, entrants are encouraged to disseminate their ideas rapidly. The final round includes a forum for poster presentations about each agent, where agent developers can share ideas and present their techniques.

In our opinion, the benefits of a trading agent competition outweigh the pitfalls. These are some of the benefits that we believe can yield positive outcomes:

- ***Complete Autonomous Agents***. In the tournaments, agent developers are confronted with the challenge of building a complete autonomous agent, instead of focusing in a specific sub-problem. The fact that agents are made to work in these electronic markets (with the high-level design and the low-level issues) lends substantial credibility to the research result.

- ***Agent that can Negotiate in a Broad Range of Market Conditions.*** The TAC community aims at creating trading agents that can perform in many different markets conditions. In order to evaluate the performance of these agents, the structure of the tournament includes many rounds with sufficient games, so that agents are confronted with many markets conditions. This avoids the development of agents that are risk-seeking (i.e., only perform well across a limited set of market conditions).

- ***Robust Agents.*** Competitions are one-shot events, so any type of failure can lead to frustration (due to hard work that has been wasted). Therefore, agent developers are required to test their agents thoroughly before the event starts. This fact encourages developers to create trading agents that are robust under as many conditions as possible.

- ***Flexible software.*** Since the competition rules may change slightly from one year to the other, agent developers must create agent designs that are flexible and adaptable.

Moreover, agent developers typically test different techniques to try to solve sub-problems of trading, so it is always good to build agent designs that can be adapted easily to such changes.

- ***Pool of Agents.*** At the end of each event, many agent developers upload their agent binary to a repository in the TAC website. By making the agents available to other members, the research community can use these agents in controlled experiments, such as the work conducted by [Jordan et al., 2007a], where the TAC SCM market is empirically tested to reveal interesting best response and equilibrium relationships.

However, the TAC markets are not entirely realistic (i.e., not perfect models of real-world markets). This is due to the fact that the main goal of TAC is to foster research into the trading agent problem, so the game design has to balance some factors: (i) making it interesting enough so that it captures elements of the real-world markets, (ii) making the rules simple so that new entrants can develop their trading agents, (iii) but, most importantly, making it challenging enough to push the boundaries of science. Therefore, the market designers are always trying to balance these factors, in order to create motivating research questions, such as: (i) how to design trading agents, (ii) how to evaluate these trading agents, and (iii) how do trading agents affect electronic markets.

We believe that the body of work from this research community can provide an engineering foundation to the real-world markets. This community has generated many novel techniques that can be applied to many different electronic markets, such as bidding heuristics, price prediction algorithms and methods, learning techniques, and optimization models. All these techniques have been tested extensively and integrated into the agents. Such techniques and agent designs can shape the future of automated trading in electronic markets.

REFERENCES

Andrews, J., Benisch, M., Sardinha, A., Sadeh, N. (2009). Using Information Gain to Analyze and Fine Tune the Performance of Supply Chain Trading Agents. In Collins, J., Faratin, P., Parsons, S., Rodriguez-Aguilar, J., Sadeh, N., Shehory, O., Sklar, E. (Eds.), *Agent-Mediated Electronic Commerce and Trading Agent Design and Analysis*, Lecture Notes in Business Information Processing, Springer Berlin Heidelberg.

Arunachalam, R. & Sadeh, N. (2005). The supply chain trading agent competition. *Electronic Commerce Research Applications*, 4(1), Elsevier, 66-84.

Benisch, M., Greenwald, A., Grypari, I., Lederman, R., Naroditsky, V., Tschantz, M. (2004). Botticelli: A supply chain management agent. In *Third International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'04)*.

Benisch, M., Andrews, J., Bangerter, D., Kirchner, T., Tsai, B., Sadeh, N. (2005). CMieux analysis and instrumentation toolkit for TAC SCM. *Technical Report CMU-ISRI-05-127*, School of Computer Science, Carnegie Mellon University.

Benisch, M., Sardinha, A., Andrews, J., Ravichandran, R., Sadeh, N. (2009). CMieux: Adaptive strategies for competitive supply chain trading. *Electronic Commerce Research and Applications*, 8 (2), 78-90.

Borghetti, B., Sodomka, E., Gini, M., Collins, J. (2006). A market-pressure-based performance evaluator for TAC-SCM. In *Proceedings of AAMAS Workshop on Trading Agent Design and Analysis (TADA)*.

Cheng, S.-F., Leung, E., Lochner, K., O'Malley, K., Reeves, D., Schvartzman, J., Wellman, M. (2005). Walverine: a Walrasian trading agent. *Decision Support Systems*, 39(2), 169-184.

He, M. & Jennings, N. (2003). SouthamptonTAC: An adaptive autonomous trading agent. *ACM Transactions on Internet Technology*, 3, 218-235.

He, M., Rogers, A., Luo, X., Jennings, N.(2006). Designing a successful trading agent for supply chain management. In *Proceedings of the 5th international joint conference on Autonomous agents and multiagent systems (AAMAS'06)*.

Jordan, P., Kiekintveld, C., Wellman, M. (2007a). Empirical game-theoretic analysis of the TAC Supply Chain game. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems (AAMAS '07)*.

Jordan, P., Kiekintveld, C., Miller, J., Wellman, M. (2007b). Market Efficiency, Sales Competition, and the Bullwhip Effect in the TAC SCM Tournaments. In Fasli, M., Shehory, O. (Eds.), *Agent-Mediated Electronic Commerce. Automated Negotiation and Strategy Design for Electronic Markets*, Lecture Notes in Computer Science, Springer Berlin / Heidelberg.

Jordan, P. & Wellman, M. (2010). Designing an Ad Auctions Game for the Trading Agent Competition. In David, E., Gerding, E., Sarne, D., Shehory, O., Jordan, P., Wellman, M. (Eds.), *Agent-Mediated Electronic Commerce. Designing Trading Strategies and Mechanisms for Electronic Markets* (pp. 147-162). Lecture Notes in Business Information Processing, Springer Berlin Heidelberg.

Keller, P., Duguay, F.-O., Precup, D. (2004). RedAgent-2003: An Autonomous Market-Based Supply-Chain Management Agent. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems* (AAMAS '04), 1182-1189.

Ketter, W., Collins, J., Block, C. (2010). Smart Grid Economics: Policy Guidance Through Competitive Simulation. *ERIM Report Series Reference No. ERS-2010-043-LIS*. Available at SSRN: http://ssrn.com/abstract=1707913

Kiekintveld, C., Wellman, M., Singh, S., Estelle, J., Vorobeychik, Y., Soni, V., Rudary, M. (2004). Distributed feedback control for decision making on supply chains. In *Proceedings of the Fourteenth International Conference on Automated Planning and Scheduling*.

Kiekintveld, C., Vorobeychik, Y., Wellman, M. (2006). An analysis of the 2004 supply chain management trading agent competition. In Poutr, H. L.; Sadeh, N.; and Janson, S., (Eds.), *Agent-Mediated Electronic Commerce: Designing Trading Agents and Mechanisms* (pp. 99-112), Number 3937 in Lecture Notes in AI, Springer-Verlag.

Lee, S., Greenwald, A., Naroditskiy, V. (2007). RoxyBot-06: An $(SAA)^2$ TAC Travel Agent. In *Twentieth International Joint Conference on Artificial Intelligence (IJCAI'07)*.

Niu, J., Cai, K., Parsons, S., Gerding, E., McBurney, P., Moyaux, T., Phelps, S., Shield, D. (2008). JCAT: a platform for the TAC market design competition. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems: demo papers (AAMAS '08)*. International Foundation for Autonomous Agents and Multiagent Systems.

Pardoe, D., & Stone, P. (2005). Bidding for customer orders in TAC SCM. In *Agent Mediated Electronic Commerce VI: Theories for and Engineering of Distributed Mechanisms and Systems (AMEC 2004)*.

Pardoe, D., & Stone, P. (2006). TacTex-05: A Champion Supply Chain Management Agent. In *Proceedings of the twenty-first national conference on artificial intelligence (AAAI'06)*.

Russell, S. & Norvig, P. (1995). *Artificial Intelligence: A Modern Approach*. Prentice-Hall.

Sardinha, J., Milidiú, R., Paranhos, P., Cunha, P., Lucena, C. (2005). An Agent Based Architecture for Highly Competitive Electronic Markets. In *Proceedings of The 18th International Florida AI Research Society Conference (FLAIRS'05)*.

Stone, P., Littman, M., Singh, S., Kearns, M. (2001). ATTac-2000: an adaptive autonomous bidding agent. *Journal of Artificial Intelligence Research,* 15, 189–206.

TAC (2011). Trading Agent Competition Web Site. Retrieved March 25, 2011, from http://www.sics.se/tac

TAC Association (2011). Association for Trading Agent Research Web Site. Retrieved March 25, 2011, from http://tradingagents.org/

Toulis, P., Kehagias, D., Mitkas, P. (2006). Mertacor: a successful autonomous trading agent. In *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems (AAMAS '06)*. ACM, New York, NY, USA, 1191-1198.

Vetsikas, I. & Selman, B. (2003). A principled study of the design tradeoffs for autonomous trading agents. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems (AAMAS '03)*. ACM, New York, NY, USA, 473-480.

Wellman, M., Wurman, P., O'Malley, K., Bangera, R., Lin, S.-D., Reeves, D., Walsh, W. (2001). Designing the market game for a trading agent competition. *IEEE Internet Computing.* 5 (2), 43–51.

Wellman, M., Greenwald, A., Stone, P. (2007). *Autonomous Bidding Agents: Strategies and Lessons from the Trading Agent Competition*. The MIT Press.

Wooldridge, M. (1999). Intelligent Agents. In G. Weiss (Ed.), *Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence* (pp. 1-51). MIT Press.

ADDITIONAL READING SECTION

Benisch, M., Greenwald, A., Grypari, I., Lederman, R., Naroditskiy, V., Tschantz, M. (2004). Botticelli: A Supply Chain Management Agent Designed to Optimize under Uncertainty. *SIGecom Exchanges* 4(3), 29–37.

Burke, D., Brown, K., Tarim, S., Hnich, B. (2008). Learning market prices in real-time supply chain management. *Computers & Operations Research* 35, 3465–3478.

Collins, J., Ketter, W., Gini, M. (2009) Flexible decision control in an autonomous trading agent. *Electronic Commerce Research and Applications* 8, 91-105.

Collins, J., Ketter, W., Sadeh, N. (2010). Pushing the limits of rational agents: The Trading Agent Competition for supply chain management. *AI Magazine* 31(2), 63–80.

Dahlgren, E., Wurman, P. (2004). PackaTAC: A Conservative Trading Agent. *SIGecom Exchanges* 4(3), 38-45.

Eriksson, J., Janson, S. (2002). The Trading Agent Competition – TAC 2002. *ERCIM News* 51.

Eriksson, J., Finne, N., Janson, S. (2006) Evolution of a Supply Chain Management Game for the Trading Agent Competition. *AI Communications* 19(1), 1-12.

Greenwald, A. (2002). The International Trading Agent Competition: Focus on RoxyBot. Computing Research News 14(5), 2002.

Greenwald, A. (2003). The 2002 trading agent competition: An overview of agent strategies. *AI Magazine* 24(1), 83-91.

Greenwald, A., Boyan, J. (2005). Bidding algorithms for simultaneous auctions: A case study. *Journal of Autonomous Agents and Multiagent Systems* 10(1), 67-89.

He, M., Jennings, N. (2004). Designing a successful trading agent: A fuzzy set approach. *IEEE Transactions on Fuzzy Systems* 12, 389-410.

Keller, P., Duguay, F., Precup, D. (2004). RedAgent: Winner of TAC SCM 2003. *SIGecom Exchanges* 4(3), 1-8.

Ketter, W., Collins, J., Gini, M., Gupta, A., Schrater, P (2009). Detecting and forecasting economic regimes in multi-agent automated exchanges. *Decision Support Systems* 47, 307–318.

Kiekintveld, C., Wellman, M., Singh, S., Soni, V. (2004) Value-Driven Procurement in the TAC Supply Chain Game. *SIGecom Exchanges* 4(3), 9-18.

Kiekintveld, C., Miller, J., Jordan, P., Callender, L., Wellman, M. (2009) Forecasting market prices in a supply chain game. *Electronic Commerce Research and Applications* 8, 63-77.

Pardoe, D., Stone, P. (2004). TacTex-03: A Supply Chain Management Agent. *SIGecom Exchanges* 4(3), 19-28.

Sadeh, N., Arunachalam, R., Eriksson, J., Finne, N., Janson, S. (2003). TAC-03: A supply-chain trading competition. *AI Magazine* 24(1), 92-94.

Sardinha, A., Benisch, M., Sadeh, N., Ravichandran, R., Podobnik, V., Stan, M. (2009). The 2007 Procurement Challenge: A Competition to Evaluate Mixed Procurement Strategies. *Electronic Commerce Research and Applications* 8, 106-114.

Si, Y., Edmond, D., Dumas, M., Chong, C. (2007). Strategies in supply chain management for the Trading Agent Competition. *Electronic Commerce Research and Applications* 6, 369–382.

Stone, P., Schapire, R., Littman, M., Csirik, J., McAllester, D. (2003). Decision-theoretic bidding based on learned density models in simultaneous, interacting auctions. *Journal of Artificial Intelligence Research* 19, 209-242.

Stone, P. (2003). Multiagent competitions and research: Lessons from RoboCup and TAC. In G. Kaminka, P. Lima, R. Rojas (Eds.), *RoboCup-2002: Robot Soccer World Cup VI*, 224-237, Springer-Verlag.

Wellman, M., Greenwald, A., Stone, P., Wurman, P., (2003). The 2001 Trading Agent Competition. *Electronic Markets* 13(1), 4-12.

Wellman, M., Cheng, S., Reeves, D., Lochner, K. (2003). Trading agents competing: Performance, progress, and market effectiveness. *IEEE Intelligent Systems* 18(6), 48-53.

Wellman, M., Reeves, D., Lochner, K., Vorobeychik, Y. (2004). Price prediction in a trading agent competition. *Journal of Artificial Intelligence Research* 21, 19-36.

Wellman, M., Estelle, J., Singh, S., Vorobeychik, Y., Kiekintveld, C., Soni, V. (2005). Strategic interactions in a supply chain game. *Computational Intelligence* 21, 1-26.

Wellman, M., Reeves, D., Lochner, K., Suri, R. (2006). Searching for Walverine 2005. In H. La Poutre, N. Sadeh, S. Janson (Eds.), *Agent-Mediated Electronic Commerce: Designing Trading Agents and Mechanisms*, LNAI 3937, Springer-Verlag.

Zhang, D., Zhao, K., Liang, C., Huq, G., Huang, T. (2004). Strategic Trading Agents via Market Modelling. *SIGecom Exchanges* 4(3), 46-55.

KEY TERMS AND DEFINITIONS

Software Agents: Software programs that are capable of making decisions autonomously in order to achieve goals.

Multi-Agent System: A group of software agents in an environment with resources, which have to collaborate or compete to achieve a common goal.

Trading Agents: Autonomous software agents that negotiate goods to achieve trading objectives.

Trading Agent Competition (TAC): An annual event to foster high quality research into the trading agent problem.

Agent Developers: A research team that develops a trading agent for the TAC scenarios.

TAC Travel: A TAC scenario where trading agents negotiate travel goods.

TAC Supply Chain Management: A TAC Scenario where trading agents negotiate PCs and components in a supply chain.