Strategic Issues in Trading Agent Competition: TAC-Classic

Francisco Oliveira, Yain-Whar Si
Faculty of Science and Technology, University of Macau, Macao
olifran@umac.mo, fstasp@umac.mo

Abstract

With the advancement of Internet Technology, increasing number of electronic marketplaces and online auction houses provides real-time transaction services for buyers and sellers regardless of their geographical locations. Easy access to these services has also fueled the need for automating the overall negotiation and decision making process. The annual Trading Agent Competition (TAC) is an international forum designed to provide a competitive benchmark for the trading agents developers around the world. The first TAC was held in July, 2000. In the TAC-Classic category, travel agents compete against each other to assemble travel packages on behalf of several clients. The objective of the travel agent is to maximize the total satisfaction of its clients. During the competition, travel agents may perform three crucial tasks: allocation of resources according to the clients' preferences, prediction of air ticket and hotel prices, and bidding in respective auctions for the required goods. In this paper, we review the algorithms, techniques and heuristics used by the agents which have the highest scores in the last six years of tournaments.

1. Introduction

The Trading Agent Competition (TAC) [1] is an international forum designed to promote and encourage high quality research into the trading agent problem. Two types of games are played in the competition, TAC-Classic and TAC-SCM (for Supply Chain Management). A number of reports have been published on the result of the competitions[1].

In the TAC-Classic shopping game [3], travel agents compete against each other to assemble travel packages (from TAC-town to Tampa, during a five day period). Each agent acts on behalf of eight clients and each client has specific preferences over various aspects of the trip. Travel packages consist of round-trip flights, hotel reservations, and entertainment tickets (alligator wrestling, amusement park, and museum). These goods can be traded in respective auctions during the simulated game. The objective of the agents is to maximize the total satisfaction of its clients. An overview of a TAC-Classic scenario is depicted in Figure 1.

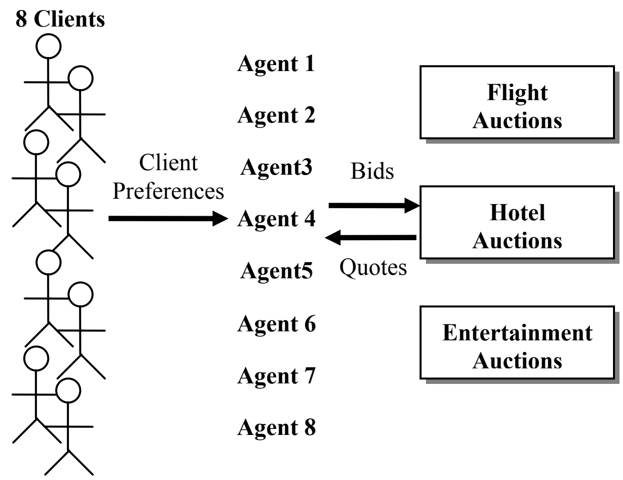

Figure 1. A TAC-Classic Game

During each tournament, trading agents may perform three crucial tasks: resource allocation, price prediction, and placing bids in appropriate auctions. Resource allocation is used to determine which goods to be acquired from the auctions or is used to assign purchased goods to clients in order to maximize the total client utility [4]. Price prediction is used to estimate the auction clearing prices. Price prediction is considered as one of the important task in TAC [5] since it has significant influence on the agent's bidding decisions. Price prediction can be divided into two phases: initial and interim [6]. Initial prediction is made early in the game whereas interim prediction is performed as the game progresses. Bidding heuristics are used to decide when to bid in the respective auctions. Based on the estimated prices and the bidding heuristics, the agent places appropriate bids in the auctions. Bidding can be done either at the beginning of the game or at the later stages of the game. *Early bird* heuristic agents [7] make resource allocation decisions at the beginning of the game and do not make any future changes whereas *deliberate* buyer heuristic agents make bidding decisions throughout the game and bid in the auctions when needed. Early bird heuristics and deliberate buyer heuristics are also called open-loop and close-loop [8].

[1] An extensive list of publications on TAC-Classic can be found in [2].

Proceedings of the 2006 IEEE/WIC/ACM International Conference on
Web Intelligence and Intelligent Agent Technology (WI-IAT 2006 Workshops)(WI-IATW'06)
0-7695-2749-3/06 $20.00

Table 1. TAC-Classic final results from 2000 to 2005

POS	2000	2001	2002	2003	2004	2005
1	ATTac	Livingagents	Whitebear	Attac-01	Whitebear04	Mertacor
2	RoxyBot	ATTac	SouthamptonTAC	PACKATAC	Walverine	Whitebear05
3	Aster	Whitebear	Thalis	Whitebear	LearnAgents	Walverine
4	UMBCTAC	Urlaub	UMBCTAC	Thalis	SICS02	Dolphin
5	ALTA	Retsina	Walverine	UMBCTAC	NNN	SICS02
6	DAIHard	SouthamptonTAC	Livingagents	NNN	UMTac-04	LearnAgents
7	RiskPro	CaiserSose	KavayaH	Walverine	Agent@CSE	e-Agent
8	T1	TacsMan	CUHK	RoxyBot	RoxyBot	RoxyBot
9				Zepp		

In this paper, we review the main strategies applied by 23 top scoring agents from the last six tournaments focusing on resource allocation, price prediction, and bidding heuristics. These agents are listed in Table 1 based on their final ranking in the competitions [2]. We excluded the discussion of some of the agents due to the unavailability of related information on their strategies and heuristics.

This paper is organized as follows: a brief introduction to the TAC-Classic game is given in section 2. Comparison of agents' strategies is given in section 3. Strategies for resource allocation, price prediction, and bidding heuristics are detailed in section 4, 5, and 6 respectively. Finally, we conclude our survey in section 7.

2. TAC-Classic Game Overview

There are eight flight auctions: TAC-town to Tampa (days 1 to 4) and Tampa to TAC-town (days 2 to 5). Flight auctions are continuous one-sided auctions, and close at the end of the game. Agents can only submit buy bids and only the TACAIR seller may submit sell bids. The quote is always equal to the current sell bid. Any buy bids that are at least as high as the current asking price will match immediately at the ask price. Any buy bids that do not match immediately remain in the auction as standing bids. A standing buy bid remains in the auction until it is matched by a sell bid with a price at or below the buy bid, or until the bid is withdrawn.

There are two hotels in the game: the up-market Tampa Towers (TT) and the less attractive Shoreline Shanty (SS). Hotel auctions are multi-unit English auctions. Every minute during the game, a randomly chosen auction will be closed. Price quotes are generated once per minute. The initial prices of all the auctions are set to zero. Overall, there are eight hotel auctions (one for each combination of hotel and night, apart from the last one) and 16 rooms are sold in each auction. When an auction closes, the 16 highest buy units will be matched and the winning agents will pay at the 16th highest price.

Each agent is randomly endowed with 12 entertainment tickets at the beginning of the game. There are 12 continuous double auctions for trading entertainment tickets (for each kind of entertainment for each day 1 to 4). A travel package is feasible if it contains rooms for every night between the arrival and departure dates (exclusive) and all rooms are in the same hotel. Entertainment tickets are not compulsory for a travel package. However, two identical entertainment events are not allowed within the same package.

Client's utility is given by the following equation:

$$\begin{aligned} U = 1000 - &\left[100 \times \left(|AA - PA| + |AD - PD|\right)\right] + \\ &\left(TTb \times HP\right) + \\ &\left[\left(AWb \times AW\right) + \left(APb \times AP\right) + \left(MUb \times MU\right)\right] \end{aligned} \qquad (1)$$

AA and *AD* are defined as the actual arrival and departure dates of the clients whereas *PA* and *PD* are the preferred dates of arrival and departure of the clients. *HP*, *AW*, *AP*, and *MU* are the premium values of securing a better hotel, and having entertainment tickets for alligator wrestling, amusement park, and museum. *TTb*, *AWb*, *APb*, and *MUb* are binary values used to indicate whether clients got a better hotel and tickets for entertainment events or not.

3. Comparison of Agent's Strategies

A summary of some of the agents who competed in TAC-Classic are listed in Table 2. It contains the name of the agents, corresponding affiliation, resource allocation techniques, price prediction approaches (some of the information is extracted from [6]), and the bidding heuristics applied (which can be classified either as Early Bird (E) or Deliberate Buyer (D)). Statistical information based on Table 2 is depicted in Figures 2, 3, and 4 respectively.

[2] Due to the space limitation, we have omitted the detailed review for each agent. However, we are in the process of releasing the extended version of this survey as a technical report.

Proceedings of the 2006 IEEE/WIC/ACM International Conference on
Web Intelligence and Intelligent Agent Technology (WI-IAT 2006 Workshops)(WI-IATW'06)
0-7695-2749-3/06 $20.00 © 2006 IEEE

Table 2. Summary of Agent Strategies

Agent Name	Affiliation	Resource Allocation	Price Prediction	Bidding Heuristics
ALTA	Artificial Life, Inc.	Genetic Algorithms	Historical	E
Aster	Intertrust Technologies Corp.	Heuristic Search	Historical	D
ATTac	ATT Labs, Research	Integer Programming	Machine Learning	E/D
CaiserSose	University of Essex	A* Search	Initial hotel bid price	D
CUHK	Chinese University of H.K.	Greedy Search	Historical	E/D
KavayaH	Oracle India	Genetic Algorithms	Neural Networks	D
LearnAgents	PUC-Rio, Rio de Janeiro, Brasil	Integer Programming	Historical	D
Livingagents	Living Systems, AG	Exhaustive search	Historical	E
Mertarcor	Aristotle University of Thessaloniki	Linear Programming	Historical	D
PACKATAC	North Carolina State University	Linear Programming & Heuristics	Historical	E/D
Retsina	Carnegie Mellon University	Markov Chain Monte Carlo	Historical	D
RiskPro	SICS	Heuristic search	Threshold values	D
RoxyBot	Brown University	A* Search	Machine Learning	D
SICS	SICS	Branch-Bound Search	Historical	D
SouthamptonTAC	University of Southampton	Integer Programming	Fuzzy Logic	D
T1	SICS & Industrilogik	Greedy Search	Threshold values	E/D
TACSMan	Stanford University	Integer Programming	Historical	D
Thalis	University of Essex	Exhaustive Search	Historical	D
UMBCTAC	U. Maryland at Baltimore County	Exhaustive Search	Historical	E/D
UMTac	University of Macau	Genetic Algorithms	Historical	D
Walverine	University of Michigan	Integer Linear Programming	Competitive Equilibrium	D
Whitebear	Cornell University	Randomized Greedy Algorithm	Bayesian analysis	E/D
Zepp	Politehnica University	Greedy Search	Risk management	D

In Figure 2, 53% of the agents apply searching algorithms, 13% of the agents apply machine learning algorithms, and 30% of the agents apply either Integer or Linear programming algorithms.

Although these approaches may not be able to provide an optimal solution, it is widely considered to be sufficient if a near optimal solution can be obtained.

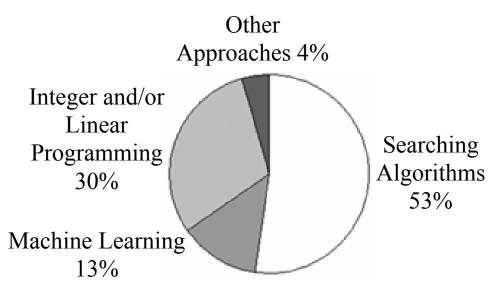

Figure 2. Resource Allocation techniques applied

In Figure 3, 57% of the agents use historical information in prediction, 13% of the agents use machine learning algorithms, 13% of them considered Threshold values, and 17% of the agents applied other approaches.

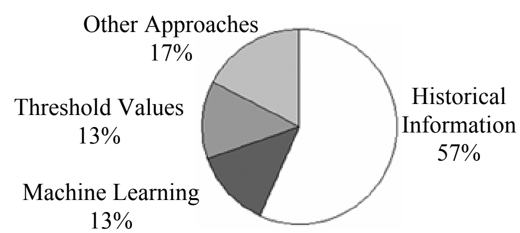

Figure 3. Price Prediction Strategies applied

In Figure 4, 65% of the agents apply deliberate buyer heuristics, 26% of the agents apply early bird heuristics, and 9% of the agents apply both early bird and deliberate buyer heuristics.

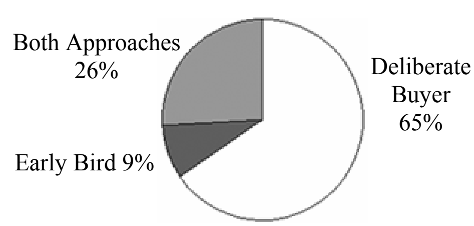

Figure 4. Bidding Heuristics applied

From our review, we find that agents from the early years of competition tend to use simple algorithms

Proceedings of the 2006 IEEE/WIC/ACM International Conference on
Web Intelligence and Intelligent Agent Technology (WI-IAT 2006 Workshops)(WI-IATW'06)
0-7695-2749-3/06 $20.00 © 2006 IEEE

whereas agents from recent competition adapt one or more sophisticated A.I. techniques.

4. Resource Allocation Techniques

Allocation strategies can be divided into three main categories: searching algorithms based strategies, integer and linear programming based strategies, and machine learning algorithms based strategies.

During the competition, various searching algorithms are used for constructing search trees where each node represents a possible travel package. Roxybot [9] applied A* search algorithm in its implementations which is divided into two stages: assignment of travel packages (flight tickets and hotel rooms), and the assignment of the entertainment packages. After dividing the search tree into two stages, the agent then applied appropriate heuristics to reduce the search time.

Aster [5] and RiskPro [5] applied heuristic search algorithms in its resource allocation. Aster first generates a tentative allocation based on the estimated costs and current holdings. It then uses a heuristic search for deriving the optimum allocation. In RiskPro, it generates an allocation of all available goods by regarding subsets of goods as a single bundle, and it will then search for these bundles for allocation.

Thalis [10] and LivingAgents [11] performed an exhaustive search that evaluates all the possible packages during each client's stay and selects an optimum package that can maximize the overall utility. Although UMBCTac [7] also performed an exhaustive brute force search to derive optimum packages for all the clients, UMBCTac tried to strike a balance between profitability and safety by deriving the most profitable yet less risky allocation.

The strategy of CUHK [5] and Whitebear [12] for deriving the best packages for the clients is based on a greedy search. CUHK divides the search into two phases, one for the set of hotel rooms and flight tickets, and the other for allocating entertainment tickets to the clients. Whitebear divides the whole TAC problem into several sub-problems, and for each one, it generates near optimal allocations based on a randomized greedy algorithm.

SICS [13] applied a branch-and-bound search based on the client preferences and the estimated costs of winning in each auction.

A number of agents applied Integer and/or Linear Programming techniques to allocate their resources, including ATTac [14], LearnAgents [15], Mertarcor [16], PackaTAC [17], Walverine [18], SouthamptonTAC [19], and TACSman [1].

Some of the agents use machine learning algorithms to derive optimal allocation based on historical and live TAC-Market data. ALTA [5], KavayaH [20], and UMTac [21] applied genetic algorithms to derive the optimal set of plans that maximizes the total client's utility. Each agent generates its own population of chromosome and performs crossover, mutation, and replacement techniques for new generations.

Retsina [22] is one of the agents that applied a different technique for its resource allocation. It uses a Markov Chain Monte Carlo approach which is a variation of simulated annealing that starts with a random allocation. Retsina then performs a random biased walk on the set of all possible allocations.

5. Price Prediction Strategies

Price prediction algorithms play an important role in the success of trading agents in the competition.

A majority of the agents make use of previous historical averages to estimate the expected hotel clearing price. These agents include Alta [5], Aster [5], CUHK [5], LearnAgents [15], LivingAgents [11], Mertarcor [16], PackaTac [17], Retsina [22], SICS [13], TacsMan [1], Thalis [10], UMBCTAC [7], and UMTac [21].

Although most of the agent's price prediction is based on the historical averages in order to make suitable bids, agents participated for the first time in 2000 TAC competition, like RiskPro [5], T1 [5], and CaiserSose [23] used current ask quotes, holdings, etc. to make bidding decisions since they do not have any historical data.

Machine learning techniques are also used to derive relationships between observable parameters and resulting hotel prices [4]. For instance, ATTac [14] agent constructs a model of the probability distribution over clearing prices. In KavayaH [20] agent, a neural network was used to predict the prices. SouthamptonTAC [19] estimates the hotel closing prices based on fuzzy reasoning methods.

Some agents avoid Artificial Intelligence approaches in predicting the prices. For instance, Walverine [18] places bids optimally based on a competitive analysis of the TAC travel economy. The prediction in Zepp [24] is done by using a risk-management strategy to account for potentially expensive hotel prices

6. Bidding Heuristics

Choosing the right time to bid in the TAC auctions is also one of the crucial factors considered by the agents' designers. Agents with early bird heuristics try to obtain goods with relatively low prices. However, the benefit can be offset by the cost of imperfect price prediction. Although agents with deliberate buyer heuristics are flexible in acquiring goods, they are often forced to buy flight tickets at the final moment of the game. These agents may spend more since the flight tickets are getting expensive as the time decay. Although these two

Proceedings of the 2006 IEEE/WIC/ACM International Conference on
Web Intelligence and Intelligent Agent Technology (WI-IAT 2006 Workshops)(WI-IATW'06)
0-7695-2749-3/06 $20.00 © 2006 IEEE

heuristics are complementary, some agents also adapt both strategies in their design.

7. Conclusion

In this paper, we detailed a comprehensive review of the strategies and heuristics applied by 23 agents from Trading Agent Competition (Classic Category). Our survey reveals that, in order to achieve good results in the competition, the design of these agents cannot only rely on complex and efficient algorithms, but economic issues are also needed to be considered. Although most of the agents in this survey apply sophisticated A.I. techniques, they are also designed to address crucial economic issues such as marginal utility, relationships between supply and demand, and market dynamics.

8. References

[1] M. P. Wellman, A. Greenwald, P.Stone, and P. R. Wurman, “The 2001 trading agent competition”, Proceedings of the 14th Annual Conference on Innovative Applications of Artificial Intelligence (IAAI-02), Edmonton, Alberta, Canada, 2002, pp. 935-941.

[2] Trading Agent Competition (TAC) Research Reports, http://tac.eecs.umich.edu/researchreport.html, accessed July 2006.

[3] Trading Agent Competition, TAC-Classic, http://www.sics.se/tac/page.php?id=3, accessed July 2006.

[4] A. Greenwald and J. Boyan, “Bidding algorithms for simultaneous auctions: A case study”, Third ACM Conference on Electronic Commerce, Tampa, FL., 2001, pp. 115-124.

[5] P. Stone and A. Greenwald, “The first international trading agent competition: Autonomous bidding agents”, Electronic Commerce Research: Special Issue on Dynamic Pricing, Springer, vol. 5, 2005, pp. 229-265.

[6] M. P. Wellman, D. M. Reeves, K. M. Lochner, and Y. Vorobeychik, “Price prediction in a trading agent competition”, Journal of Artificial Intelligence Research, vol. 21, 2004, pp. 19-36.

[7] L. Ding, T. Finin, Y. Shi, Y. Zou, Z. Ding, and R. Pan, “Strategies and heuristics used by the UMBCTAC agent in the third Trading Agent Competition”, IJCAI-03 Workshop on Trading Agent Design and Analysis, Acapulco, 2003.

[8] P. Stone, R. E. Schapire, J. Csirik, M. L. Littman, D. A. McAllester, “ATTac-2001: A Learning, Autonomous Bidding Agent”, Proceedings of Workshop on Agent Mediated Electronic Commerce IV: Designing Mechanisms and Systems, 2002, pp. 143-160.

[9] A. Greenwald, “The International Trading Agent Competition: Focus on RoxyBot”, Computing Research News, vol. 14, no. 5, 2002, pp. 3-10.

[10] M. Fasli and N. Poursanidis, “Thalis: An autonomous trading agent description”, http://www.essex.ac.uk/isf/Seminarpapers/Fasli.pdf, accessed January 2005.

[11] C. Fritschi and K. Dorer, “Agent-oriented software engineering for successful TAC participation”, First International Joint Conference on Autonomous Agents and Multi-Agent Systems, Bologna, 2002.

[12] I. A. Vetsikas, “Whitebear: An adaptive agent description”,http://tac.eecs.umich.edu/research_reports/whitebear.pdf, accessed January 2005.

[13] P. Hultman, “SICS Agent description”, http://tac.eecs.umich.edu/research_reports/SICSAgent.pdf, accessed January 2005.

[14] P. Stone, M. L. Littman, S. Singh, and M. Kearns, “ATTac-2000: An adaptive autonomous bidding agent”, Journal of Artificial Intelligence Research, vol. 15, 2001, pp. 189-206.

[15] J. A. R. P. Sardinha, R. L. Milidiú, C. J. P. Lucena, P. M. Paranhos, and P. M. Cunha, “Learnagents agent description”, http://www.teccomm.les.inf.puc-rio.br/tac/LearnAgents.html, accessed January 2005.

[16] P. Toulis, D. Kehagias, and P. A. Mitkas, "Mertacor: A Successful Autonomous Trading Agent", Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS’06), 2006, pp. 1191–1198.

[17] P. Wurman, G. Cai, L. Johnson, S. Somanchi, and A. Sureka, "PackaTAC agent description", http://tac.eecs.umich.edu/research_reports/PackaTAC-2.pdf, accessed January 2005.

[18] S. Cheng, E. Leung, K. Lochner, K. O'Malley, D. Reeves, and M. Wellman, “Walverine: A Walrasian trading agent”, Proceedings of the Second International Joint Conference on Autonomous agents and Multi-agent Systems, 2003, pp. 465-472.

[19] M. He and N. R. Jennings, “SouthamptonTAC: An adaptive autonomous trading agent, ACM Transactions on Internet Technology”, vol. 3, 2003, pp. 218-235.

[20] R. P. Putchala, V. N. Morris, R. Khazhanchi, L. Raman, and S. Shekhar, “kavayaH agent description”, http://tac.eecs.umich.edu/research_reports/kavayaH02.pdf, accessed January 2005.

[21] Lei Man Kin, “Trading Agent Design TAC-Classic UMTac-04”, Final Project report, University of Macau, Macao, 2004.

[22] K. Sycara, Y. Nevmyvaka, P. Huang, and G. Economou, "Retsina intelligent trading agent description", http://tac.eecs.umich.edu/research_reports/Retsina.pdf, accessed January 2005.

[23] M. Fasli, I. Korres, G. Rallidis, and A. Zakas, “Caisersose: Trading agent description”, http://tac.eecs.umich.edu/research_reports/CaiserSose.pdf, accessed January 2005.

[24] O. Trascu, “Zepp: An adaptive multi-agent system approach”,http://tac.eecs.umich.edu/research_reports/Ovidiu-tac.pdf, accessed January 2005.