JAgent: Strategic Demand Reduction and Performance Against Duplicate Agents

Jeremy Stober and Jeremy Holländer

December 7, 2006

Abstract

We discuss the design and performance of JAgent, a TAC SCM trading agent. JAgent strategically reduces component demand in hostile market conditions. By doing so, JAgent exhibits better performance when pitted against copies of itself than many other highly competitive TAC SCM agents. We tested three other agents, TACStarterAgent, TacTex06, and PhantAgent06 in competitions involving only duplicate copies of one agent. JAgent was the only agent out of this set to consistently finish with a positive bank balance. In addition, JAgent is able to dynamically increase component purchasing volume in response to favorable market conditions, yielding favorable performance in non-competitive, or strategically advantageous markets.

1 Introduction

The design of our agent is meant to explore the space of strategies that are robust when pitted against duplicate copies. In [6], the authors remark on the poor revenue performance in games involving duplicate agents. The authors speculate that duplicate agents crowd each other out of a preferred niche in the TAC SCM strategy space. Even adaptive agents, such as TacTex, suffer from negative revenue in competitions involving duplicate copies of itself. The dummy agents provided by the TAC SCM server, however, do not have this property, and so it is natural to assume that competitiveness is somehow inversely proportional to performance against duplicate copies.

JAgent is an attempt to design an agent that performs well in duplication games, while still remaining competitive in general games where the strategy mix is heterogeneous. To accomplish this, JAgent strategically reduces its demand for components, and consequently its overall trading volume, in negative margin scenarios. We give a technical description of our agent in the next section, present our experimental findings, and then conclude with a discussion of the various strategy issues JAgent addresses in the context of TAC SCM.

2 JAgent Description

2.1 Supply Manager

Component purchasing is done on a fixed schedule governed by two parameters, the supply interval (SI) and the lead time (LT). When the current date D is a multiple of the supply interval, JAgent places the same component order for each component starting on date $D + LT$ to date $D + LT + SI$. If the desired component has multiple suppliers, the order is split evenly among all suppliers of that component, essentially averaging out the cost difference between suppliers.

The amount of each component ordered, or inventory goal (IG), is increased or decreased by an inventory step size (IS) depending on whether the moving average profit over the previous n days is positive or negative. The scheduled purchase amount for each component is bounded by $[2, 202]$. Default values for these parameters are described in Table 1.

Parameter Name	Default Value
Supply Interval (SI)	5
Lead Time (LT)	10
Inventory Goal (IG)	52
Inventory Step (IS)	5
Profit Lookback Window (n)	10
Inventory Goal Range	$[2, 202]$

Table 1: Supply Manager Parameters

2.2 Demand Manager

Customer RFQs are handled by the same greedy scheduling algorithm employed by the TACStarterAgent[3]. The scheduler attempts to produce computers based on the existing and expected component inventory, current computer inventory, and available factory cycles.

Prices for responding to RFQs are set by multiplying the previous sale price $S_{i,d-1}$ for a computer type i by a factor $\alpha_{up|down}$ depending on whether the percent of offers accepted ($OA_{i,d}$) is above or below the target acceptance rate (TA). For acceptance rates within a target acceptance threshold (AT) of the desired acceptance rate, no change to the sale price is made.

To account for rapid market price volatility, the sale price update routine also accounts for the high and low market prices (M_{high} and M_{low} respectively) from the last available market report as in Equation 1.

$$S_{i,d} = \begin{cases} \max(M_{high}, S_{i,d-1} * \alpha_{up}) & OA_{i,d-1} > TA + AT \\ \min(M_{low}, S_{i,d-1} * \alpha_{down}) & OA_{i,d-1} < TA - AT \end{cases} \quad (1)$$

We summarize the default values for the JAgent demand manager in Table 2.

Parameter Name	Default Value
α_{up}	1.02
α_{down}	.98
Target Acceptance (TA)	.15
Acceptance Threshold (AT)	.05

Table 2: Demand Manager Parameters

3 Results

3.1 Heterogeneous Configuration

The following experiments were conducted over five games each. The intervals in Figures 1,2,3,4 indicate 95% confidence intervals around the average revenue. The experiments described below were run on http://landmine.cs.utexas.edu:8080, games 6-50.

The first experiment tests the performance of JAgent in a heterogeneous strategy environment. We used several agent binaries available from the TAC Agent Repository [5] along with TACStarterAgent.

Figure 1 shows the average revenue for six agents over five games each. During the five games, the revenue of TAC SCM agents TacTex06, Maxon06, Southampton06 and PhantAgent06 varied between 5 million and 22 million. Our agent on the other hand had a revenue just below zero but consistently above the TAC starter agent's revenue.

TACStarterAgent's poor performance may be attributed to its weak component buying strategy but also to some extent to its weak computer pricing strategy. In game 8 the starter agent's end result amounted to -17588873. This is first and foremost due to the agent's weak component buying strategy which is reflected in the fact that the cost for material was much higher than the revenue earned from computers sold. Finally, the starter agent holds too many components and computers at the end of the game. While TacTex06 and JAgent each hold 709 and 350 components respectively at the end of the game, the starter agent holds 5693 components.

3.2 Six Duplicate Agents

For the next experiment, we ran six duplicate copies of the same agent for five games each. We tested JAgent, TACStarterAgent, PhantAgent, and TacTex in this configuration.

Figure 2 shows the 95% confidence intervals around the average revenue of agents in this scenario. TacTex06, PhantAgent, and TACStarterAgent's revenues range between -5 and -55 million when playing against duplicates of themselves.

When the bank balance of our agent declines, so does our acquisition of components. JAgent manages to stay above zero by refraining from buying components when profit is negative. Although the final balance of each JAgent

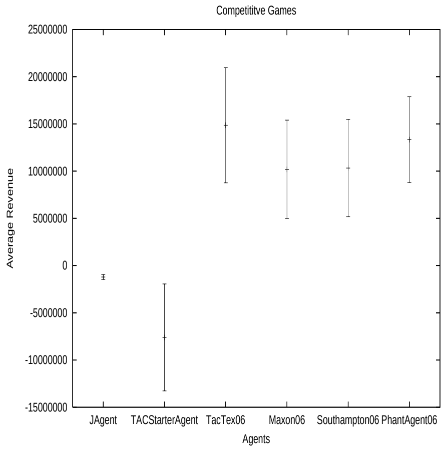

Figure 1: Competitive Games

Player	Revenue	Interest	Costs			Margin1	Margin2	Result
			Material	Storage	Penalty			
JAgent	6454255	-104915	7259893	94449	899529	-13%	-29%	-1904531
TacTex06	94206327	-658242	98896907	1365447	346819	-5%	-6%	-7061088
TacTex065	96036853	-652640	100497121	1351598	694241	-5%	-6%	-7158747
TacTex063	97184215	-641092	102132524	1397093	355804	-6%	-7%	-7342298
TacTex064	97946629	-651019	103291222	1360698	586635	-6%	-7%	-7942945
TacTex062	96668288	-676287	102496800	1450243	585612	-7%	-8%	-8540654

Table 3: Game 40 Results

binary in the duplicate agents competition is relatively small, JAgent is the only agent whose revenue is consistently above zero when playing against other copies of itself.

3.3 JAgent and Five Duplicate Agents

We also tested JAgent against five copies of another agent. We used the same agents as in the duplicate configuration above.

Figure 3 shows the average revenue with a confidence level of 95% when JAgent plays against five copies of another agent. In each case it wins against TACStarterAgent and TacTex and, in some cases, it also wins when playing against five copies of PhantAgent.

In the game theoretic analysis of TAC SCM found in [6], PhantAgent was found to have the most robust strategy when pitted against duplicate copies of itself. PhantAgent's good performance in this scenario supports that conclusion.

Table 3 shows the results of a typical game between JAgent and five copies of TacTex. While copies of the TacTex agent sell at a high volume, each computer sold incurs a small loss since the margin is negative. The duplication affects material costs since each copy of TacTex tries to occupy the same supply niche. TacTex also incurs higher storage costs (on average 1466% of JAgent's storage cost) and interest costs.

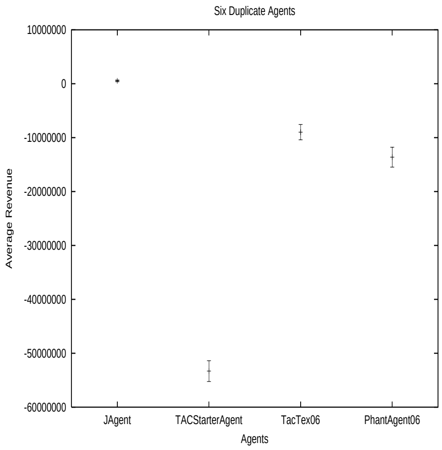

Figure 2: Duplicate Agents

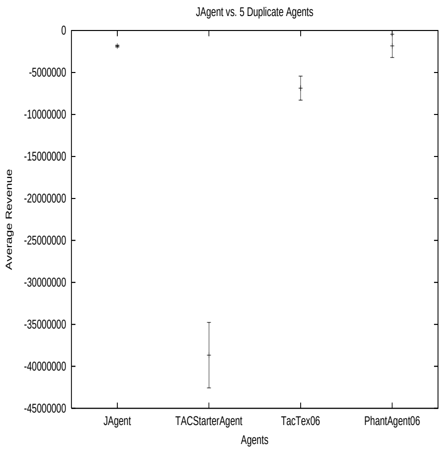

Figure 3: JAgent vs. 5 Duplicate Agents

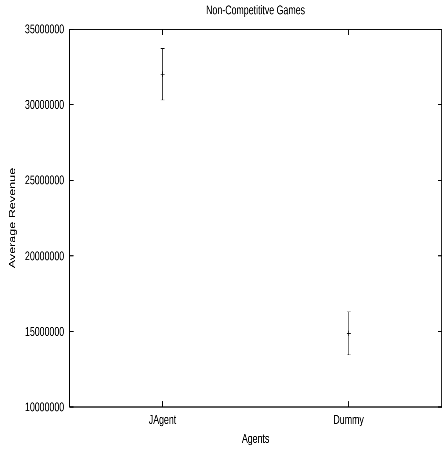

Figure 4: JAgent in Non-Competitive Games

3.4 Non-Competitive Configuration

Figure 4 shows that in non-competitive games, i.e. games against five copies of the dummy agent, our agent obtains a result ranging from 30 to 34 million, in comparison to 13 to 17 million for the dummy agents.

The average gross revenue of approximately 95 million for JAgent in non-competitive games is much higher than in the more competitive market situations described above. For example, JAgent's average gross revenue when facing five copies of PhantAgent was approximately 5 million, and in Table 3 JAgent had a gross revenue of 6.4 million. This decrease in gross revenue corresponds to a strategic decrease in component purchasing in response to consistently negative P&L.

4 Discussion

4.1 Typical Game Analysis

To fully understand JAgent's strategy it is worth noting the correlation between the account's balance, the component's holding and the number of RFQs sent. Figure 5 explores component purchases for JAgent on day 85 of game 36, a game pitting JAgent against 5 copies of TacTex. JAgent's account balance was -1.541 million. Since the moving average profit was negative, our agent strategically reduced its demand for components. As we mention in this report, our agent sends RFQs with a quantity in the range $[2, 202]$ for each component, depending on the profit being made. Because JAgent was losing money, only 240 components were requested in total.

Figure 6 shows how many components TacTex wished to obtain on day 85 when its bank account was -9.548 million and the overall customer demand

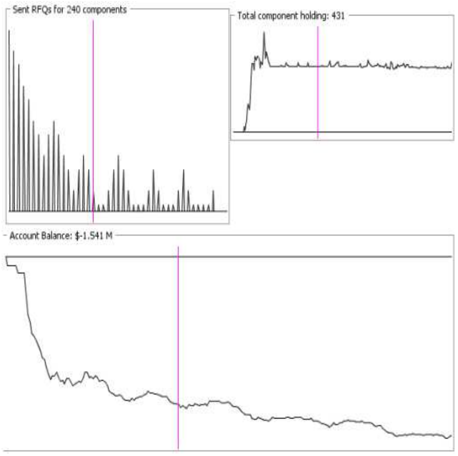

Figure 5: JAgent Component Purchasing on Day 85 of Game 36

was low. This example shows the difference in the type of adaptation between other agents and JAgent. While other agents follow a high volume strategy independent of profit, our agent stops buying components to minimize storage costs and additional interest to be paid on a negative account balance in negative market environments.

4.2 Strategic Demand Reduction

Our agent employs a simple strategy to avoid catastrophic losses often encountered when agents play duplicate copies of themselves. We modify our supply strategy by decreasing or increasing the scheduled purchase amount depending on whether the moving average profit over the previous five days was positive or negative.

JAgent never reduces the component purchase amount to 0 in order to maintain enough of a supply to continue to produce and ship computers even at a loss, for if our agent were to shut down production completely, it would lose the ability to track the relationship between offer price, acceptance rate, and component cost and so would never know when to re-enter the market should conditions improve.

4.3 Customer Demand Prediction

The Deep Maize [2] team provides a customer demand prediction package that uses a Bayesian network model to track the underlying trend and mean parameters that govern customer demand as indicated in the TAC SCM specification[1].

Figure 7 shows the variance of the distributions for each day untill the end of the game returned by the customer demand predictor on day 100. The variance increases markedly through day 100+20, achieving peak variance at

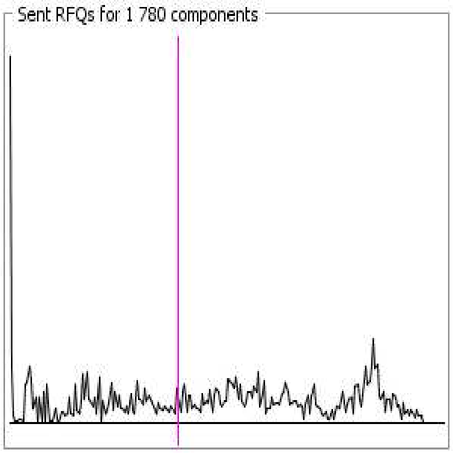

Figure 6: TacTex Component Purchases on Day 85 of Game 36

100+40 days. The sharp increase in variance corresponds to a flattening of the distribution for future days. Figure 8 shows how the distribution for day 100 demand evolves as the current day approaches day 100. Note in particular that a query for mean demand on day 100 made on day 79 would have differed markedly from a query for mean demand on day 100 made on day 99.

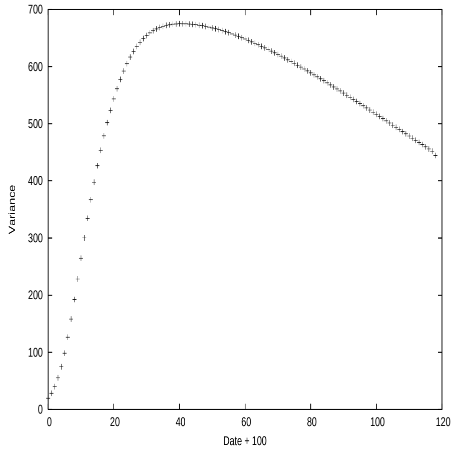

Figure 7: Customer Demand Variance on Day 100

To the extent that this prediction scheme can be considered optimal, any strategy that makes decisions more than 20 days in advance based on customer demand, must be robust in the face of high variance, and must consequently be able to handle actual demand differing significantly from predicted demand.

This may shed light on the question presented in [4] as to why changing

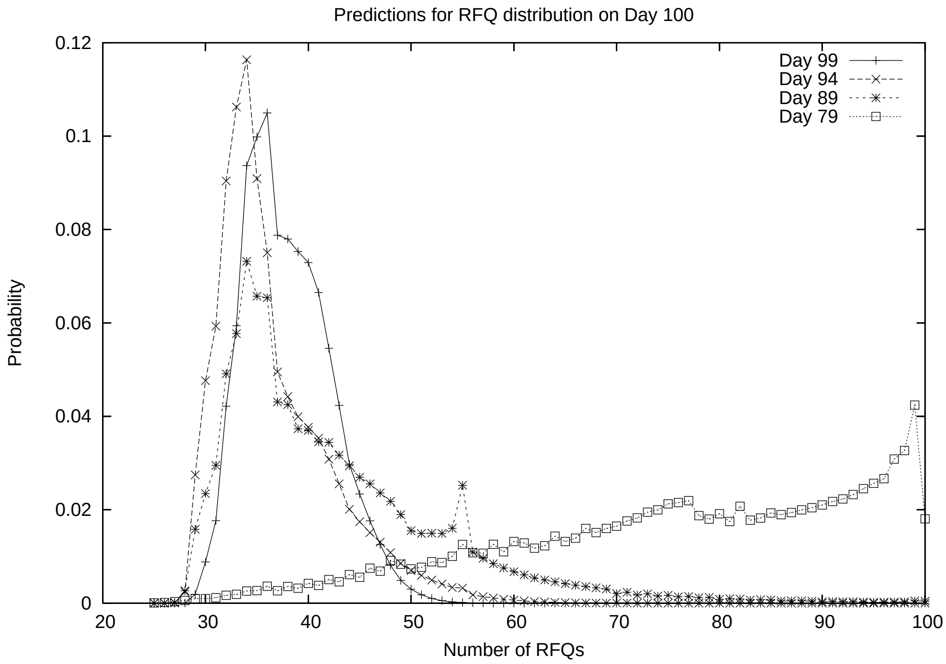

Figure 8: Deep Maize Predictions of Customer Demand

between a predictive and fixed model of customer demand has an insignificant affect on performance.

Instead of explicit demand prediction, JAgent relies on the offer acceptance rate for each computer type as a signal for updating sale prices. The offer acceptance rate encapsulates current customer demand and opposition pricing strategies. By not making forward predictions about the state of the market on future days, JAgent incurs a lag in reacting to customer demand changes.

5 Conclusion

By strategically limiting demand, JAgent performs well relative to other agents in many scenarios described above. JAgent's performance in heterogeneous strategy environments, however, is quite poor. It may be possible to improve the underlying supply and demand strategies, while still modulating supply chain volume according to moving average profit.

However, the results of the class competition and our own competitive experiments seem to indicate that reducing supply chain volume endows other participants with advantages in selecting what customer RFQs to fill and at what prices to fill them. Though this does not rule out the existence of a competitive agent strategy that performs well in both the duplicate agent and heterogeneous strategy case, this situation does indicate the possibility of a game-theoretic limitation on agent performance in both heterogeneous and homogeneous strategy environments.

References

[1] John Collins, Raghu Arunachalam, Norman Sadeh, Joakim Eriksson, Niclas Finne, and Sverker Janson. The supply chain management game for the 2006 trading agent competition. Technical Report CMU-ISRI-05-132, Cargegie Mellon University, 2005.

[2] Christopher Kiekintveld, Jason Miller, Patrick Jordan, and Michael P. Wellman. Controlling a supply chain agent using value-based decomposition. In *Seventh ACM Conference on Electornic Commerce*, Ann Arbor, MI, 2006.

[3] David Pardoe. http://www.cs.utexas.edu/ TacTex/starterAgent/.

[4] David Pardoe and Peter Stone. Predictive planning for supply chain management. In *Proceedings of the International Conference on Automated Planning and Scheduling*, Cumbria, UK, June 2006. ICAPS 06.

[5] TAC Agent Repository. http://www.sics.se/tac/showagents.php.

[6] Michael P. Wellman, Patrick R. Jordan, Christopher Kiekintveld, Jason Miller, and Daniel M. Reeves. Empirical game-theoretic analysis of the TAC market games. In *2006 Workshop on Game-Theoretic and Decision-Theoretic Agents*, Hakodate Japan, 2006.