Taking Inventory Changes into Account While Negotiating in Supply Chain Management

Celal Ozan Berk Yavuz[1][a], Çağıl Süslü[1] and Reyhan Aydoğan[1,2][b]

[1]*Department of Computer Science, Özyeğin University, Istanbul, Turkey*

[2]*Interactive Intelligence Group, Delft University of Technology, The Netherlands*

Keywords: Agent-based Negotiation, Bidding Strategy, Supply Chain Management.

Abstract: In a supply chain environment, supply chain entities need to make joint decisions on the transaction of goods under the issues quantity, delivery time and unit price in order to procure/sell goods at right quantities and time while minimizing the transaction costs. This paper presents our negotiating agent designed for Supply Chain Management League (SCML) in the International Automated Negotiation Agents Competition (ANAC). Basically, the proposed approach relies on determining reservation value by taking the changes in the inventory stock into account. We have tested the performance of our bidding strategy in the competition simulation environment and compared it with the performance of the winner strategies in ANAC SCML 2019. Our experimental results showed that the proposed strategy outperformed the winner strategies in overall.

1 INTRODUCTION

A supply chain is a network in which suppliers, manufacturers, distributors, wholesalers and retailers interact with each other by procuring and processing intermediate products or distributing goods in order to provide final products to end customers (Mbang, 2011). The main goals of supply chain are increasing efficiency by prognosticating demand, decreasing overall cost, strengthening communication between entities, dealing with the dynamic nature of the environment and so forth. To achieve those goals, supply chain entities must be in cooperation in order to operate effectively (Lin and Lin, 2004). One of the main aspects of cooperation requires joint decision making in transactions, where sellers and buyers mutually determine the unit cost and amount of the goods to be sold as well as their delivery time. By and large, there is a conflict of interests among those stakeholders. For instance, a buyer prefers to buy a product at a low cost while a seller would like to sell the product at a high cost. In such cases, they need to negotiate to resolve their conflicts and come up with an agreement. Negotiations for supply chain management has a number of challenges due to the fact that supply chain entities form a complex ecosystem. First, stakeholders are uncertain about supply and demand because of the stochastic nature of market. They do not know what is acceptable or unacceptable for other side. Learning other side's interests and preferences over their interaction can help them make well-targeted offers, which are most likely to be accepted by their opponent (Hindriks et al., 2009; Aydoğan and Yolum, 2012). Second, the supply chain has a dynamic structure where new entities may join or leave the environment. In such a dynamic and open environment, they need to choose whom to negotiate to minimize the risks arising from contract violations. Furthermore, there are multiple concurrent negotiations between sellers and buyers and they are not independent at all. It is important to establish a coordination among multiple ongoing negotiations in order to procure and sell goods at right quantities.

In the last decades, researchers work on developing agent-based negotiation technologies to automate this process (Ito et al., 2007; Fujita, 2014; Sanchez-Anguix et al., 2014; Fatima et al., 2014; de Jonge et al., 2019; Mell et al., 2018). To address the aforementioned issues, the International Automated Negotiating Agents Competition (ANAC), introduced a new league called Supply Chain Management League (SCML) in which the participants are asked to develop a factory manager agent for a supply chain simulation environment to maximize the agent's profits. In the environment, the factory manager agents need

[a] https://orcid.org/0000-0002-7946-198X

[b] https://orcid.org/0000-0002-5260-9999

Yavuz, C., Süslü, Ç. and Aydoğan, R.
Taking Inventory Changes into Account While Negotiating in Supply Chain Management.
DOI: 10.5220/0008976900940103
In *Proceedings of the 12th International Conference on Agents and Artificial Intelligence (ICAART 2020) - Volume 1*, pages 94-103
ISBN: 978-989-758-395-7; ISSN: 2184-433X

to decide on what agents to negotiate, build a dynamic endogenous utility function for negotiations, which remains robust by adjusting itself as the supply chain environment state changes, deal with the concurrent negotiations, determine bidding and acceptance strategies, decide on reservation value, manage a production schedule to decide the level of production with respect to simulation steps and so on. In this paper, we present our factory manager agent developed for the SCML. The novel aspects of our agent are reservation value adjustment strategy based on the average inventory change, and procuring all types of products, apart from the products which can be processed in factory, so as to maximize the negotiation opportunities. We have tested our agent by running simulations with the top performing factory manager agents developed by participants in SCML 2019 and found that in overall, our agent outperformed the existing agents under the performance metrics average profit, number of simulation runs in which the agents went bankrupt, and have more profit than its opponent.

The rest of the paper is organised as follows: section 2 briefly describes the supply chain environment, section 3 introduces the strategy of the factory agent, section 4 explains the experiment setup and interprets the results, section 5 expresses the existing studies in the literature, and section 6 presents conclusion.

2 NEGOTIATION COMPETITION FOR SCM

The International Automated Negotiating Agents Competition (ANAC) has been organized since 2010 to facilitate agent-based negotiation research and introduces new research challenges every year (Jonker et al., 2017). In 2019, the challenge of designing negotiating agents for supply chain management has been introduced by the organizers under Supply Chain Management League (SCML) in cooperation with NEC-AIST.

In the given environment, there are a variety of agents such as factories, miners, and consumers. The main aim is develop a factory manager agent maximizing its profit. The factory manager agents need some raw materials and intermediate products provided by miners and other factories respectively in order to produce their products which will be sold to the consumers and other factory manager agents. Consumer agents specify what products they want to buy on a bulletin board. Factory manager agents can see those requests and initiate a negotiation with consumer agents in a bilateral fashion on the unit price of their product, delivery time, grace period, quantity, and negotiated penalty. Note that grace period and negotiation penalty are optional. Furthermore, In order to produce their products, factory manager agents may also need to negotiate with miners and other factories to supply their needs.

Here, the main challenge is to design a factory agent, which decides with whom to negotiate and when to negotiate in a supply chain environment so as to maximize its profits. There are a number of challenges for designing such an agent. First, agents need to make their decisions across multiple concurrent negotiations. Second, they are not given a predefined utility function as in other negotiation environment such as Genius (Lin et al., 2014). An endogenous utility function, which dynamically estimates utilities of given offers based on environment states, should be defined by agent designers.

In the competition, the NEGMAS framework (Mohammad et al., 2019) is used to simulate the aforementioned negotiation environment. In the following part, we provide the details of this environment.

2.1 Environment Settings

In SCML environment, there is a publicly available bulletin board where agents post call for proposals (CFP) specifying what materials/products they want to buy and their constraints on the negotiation issues such as the limits for price and so on. In addition to call for proposals, the bulletin board also contains some public information such as the list of the bankrupted agents, breaches.

Based on the CFPs, other agents may initiate negotiation request. If the publisher of the underlying CFP accepts the request, negotiation starts. Each CFP is represented as a tuple as follows:

$$CFP = (p, j, q, d, c, g) \tag{1}$$

where p denotes the product type to be bought, j denotes the price interval (e.g.[0, 4]), q denotes the quantity interval, d denotes the delivery time interval, c denotes the negotiated penalty interval in case of contract violation, and g denotes the grace period interval, which states the time of signing contract.

Figure 1 depicts how the agents interact with the bulletin board. As seen below, customer agents only post CFPs to the bulletin board and factory manager agents may request for negotiation, if a customer agent accepts a negotiation request, a negotiation between these agents begin. Similar to customer agents, factory manager agents post CFPs but they can also read miners' and factory manager agents' CFPs and request for negotiations for those they are interested in. Miners on the other hand, only read CFPs and request negotiations for them.

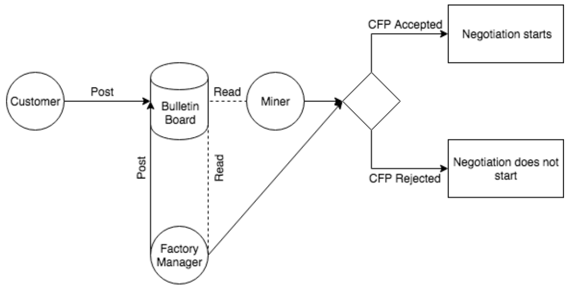

Figure 1: Agent Interactions with the bulletin board.

During the negotiation, they exchange offers based on a variant of Rubinstein's alternating offer protocol (Aydoğan et al., 2017). Different from the alternating offer protocol, both agents propose an initial offer and one of them is arbitrarily chosen as the opening offer (Mohammed et al., 2019). Afterwards, the agent receiving the offer can accept the offer, reject the offer by making a counter offer, or end the negotiation without an agreement. This process is repeated until the negotiation deadline is reached or an agreement is achieved. If an agreement is reached, the agreed offer becomes contract to be signed at the end of the grace period which as a default value of 1 step, if not negotiated. Note that agents can also refuse to sign the contract without incurring any penalty.

In order for a contract to be successfully executed, the seller party must transfer the products to the buyer's inventory and the buyer must pay the price for the products. In case of failure to execute the contract, either a breach report is imposed on the perpetrator or penalty cost should be paid. The breach information consisting of the breach type and breach level, a metric for the severity of the breach, is reported to the bulletin board. The breach types along with the breach level calculations are described below.

- **Insufficient Funds:** Reported for the buyer party failing to pay the cost of buying the products. The level of the insufficient funds breach is calculated in the following way:

$$s = \frac{a-b}{a} \tag{2}$$

where,

a : Cost of the contract for the buyer

b : Buyer's balance

- **Insufficient Products:** Reported for the seller party failing to transfer products to the buyer's inventory. The level of the insufficient products breach is calculated in the following way:

$$s = \frac{f-h}{h} \tag{3}$$

where,

f : Amount of products to be transferred to the buyer's inventory.

h : Amount of products in seller's inventory.

- **Insufficient Funds for the Penalty:** Reported for the seller party when after getting an insufficient products breach, it fails to pay the negotiated penalty in case of breach. The level of the insufficient funds for penalty breach is calculated in the following way:

$$s = \frac{z-b}{b} \tag{4}$$

where,

z : Amount of negotiated penalty to be paid by the buyer.

- **Refusal to Execute:** May be reported for either party which dishonors the contract by a refusal to execute. The breach level of refusal to execute is always 1.

In case of insufficient funds breach, the agents are given an opportunity to renegotiate, if both parties accept the renegotiate the breach may be avoided; otherwise, the opportunity is lost. If insufficient products breach is occurred, the perpetrator is required to pay the global penalty, which is 2% of the money the buyer would supposed to pay to the seller.

In the simulation environment, the production graph states the kinds of available products in the simulation environment, and manufacturing processes showing how products are processed to generate other products. The production graph is generated randomly at the beginning of the simulation and disclosed to all agents.

Each product in the production graph is classified as raw materials, intermediate products and final products. A product is said to be a raw material if it is not an output of any manufacturing process, an intermediate product if it can be both an input and output of manufacturing processes, a final product if it is not an an input to any manufacturing processes but merely an output of manufacturing processes. While the types of products and manufacturing processes available in the simulation is a public information, the agents cannot access other agents' inventory and production profile (e.g. the quantity and the type of product).

2.2 Simulation Entities

The simulation environment consists of miners, factory managers and consumers forming a supply chain in which miners supply raw materials and sell those to factory managers, which process the raw materials to produce intermediate and final products. Consumer agents drive demand for the final products and acquire them from the factory managers.

2.2.1 Miner Agents

Miner agents sell raw materials for the factory managers through negotiations. During a negotiation, they aim to maximize the quantity of raw materials supplied with a high unit price in a short amount of delivery time in order to maximize the supply chain throughput. The miners can request negotiations based on the CFPs posted by factory managers.

2.2.2 Factory Manager Agents

Factory manager agents process raw materials or intermediate products to produce another intermediate products or final materials. They have warehouses in which the products are stored and factories where there are production lines which run specific manufacturing processes. The factory manager agents have random private manufacturing process profile (i.e., input/output products, cost of processing, processing time).

The factory manager agents can post CFP to buy intermediate products or raw materials. They can negotiate with buyers by responding to the CFP posted by buyers on the bulletin board. For each negotiation threads, they introduce a utility function in order to maximize their final score, which is calculated as follows:

$$\frac{(B_n - B_0)}{B_0} \tag{5}$$

where B_n and B_0 denote the final balance and initial balance respectively.

2.2.3 Consumer Agents

Consumer agents has a consumption schedule and they purchase final products from the factory manager agents by posting buy CFPs to the bulletin board. For each negotiation thread, the utility function is determined by taking into account the deviation in the consumption schedule and unit price.

3 PROPOSED FACTORY AGENT STRATEGY

We present a new factory manager agent, namely **Adaptive Reservation Value Agent** (*ARV Agent*), which evaluates the offers with respect to inventory changes and negotiate accordingly. This agent consists of the following decision modules:

- **Deciding which Negotiations to Enter:** Our agent checks all CFPs irrespective of the required materials for its manufacturing process, and its final products and requests/accepts negotiations with the non-bankrupted agents. The main motivation for negotiating materials apart from the ones the agent can process in its factory is that our agent can enter additional negotiations and resell those to maximize profit.
- **Deciding Unit Cost of the Product to be Bought:** Unit cost of a product is the cost incurred by obtaining/producing a unit of product. Note that for the rest of the paper we refer "selling negotiation" when our agent is negotiating in order to sell the other party its products. Similarly, we refer "buying negotiation" when our agent is negotiating in order to buy products/materials from the other party. The unit cost of product p is used in utility value calculations during *selling negotiations* as follows:

$$V_p = \begin{cases} c_{price} & \eta_p = \beta_p = 0 \\ \kappa_p & (\beta_p \neq 0) \wedge (\eta_p = 0) \\ \frac{((V_i+\phi_i)*\eta_p)+(\kappa_p*\beta_p)}{\eta_p+\beta_p} & (\beta_p \neq 0) \wedge (\eta_p \neq 0) \end{cases} \tag{6}$$

where V_p denotes the unit cost of product p while V_i denotes the unit cost of input product, which is needed to produce p.

When the factory manager has not produced ($\eta_p = 0$) or bought ($\beta_p = 0$) any product p since the beginning of the simulation, the unit cost of p is equal to the catalog price of p (c_{price}).

When the factory manager produced some product p ($\eta_p \neq 0$) but not bought any product p ($\beta_p = 0$) since the beginning of the simulation, the unit cost of p is equal to average cost of buying the product p through former negotiations (κ_p).

When some p is produced in factory ($\eta_p \neq 0$) and obtained through negotiations ($\beta_p \neq 0$), the unit cost of p is equal to the weighted average of the total production cost of p $((V_i + \phi_i) * \eta_p)$ where i is the input product to produce p and ϕ_i is the processing cost of the input product and total cost of buying product p through former negotiations ($\kappa_p * \beta_p$).

- **Determining the Utility Function for Evaluating the Given Offers:** Different utility functions are defined with respect to the agent's role in the negotiation. When our agent is selling a product, the utility of an offer o is calculated as follows:

$$U_s(o) = max(((o[s_p] - V_p) * o[q_o]), 0) \quad (7)$$

where $o[s_p]$ denotes the unit price of the product specified in the offer and $o[q_o]$ denotes the quantity of the product in the offer. The utility of the offer o is equal to the difference between the total price specified in the offer and the total cost of the underlying product calculated by our agent. The utilities of the offers are normalized during negotiations.

When our agent is buying product p, the utility of the offer during the negotiation is calculated as follows:

$$U_B(o) = \begin{cases} k * o[q_o] & c_{price} = o[s_p] \\ (c_{price} - o[s_p]) * o[q_o] & otherwise \end{cases} \quad (8)$$

where $o[s_p]$ denotes the unit price of the product as stated in the offer, c_{price} catalog price of the underlying product and $o[q_o]$ denotes the quantity of the product as stated in the offer. Here, k is a coefficient equals to 0.01 in order to assign a non zero utility for the offers equal to the catalog price of the underlying product. The motivation behind this is the fact that no storing cost of products incurred by the agent and there is no inventory storage capacity, hence it is not undesirable to buy products equal to catalog price.

- **Preparing CFP:** At each simulation step, our agent posts CFPs on the bulletin board to buy products. Thus, other factory managers or miners interested in our agent's CFP can request negotiation. The CFPs are constructed as follows:

Algorithm 1.

$\alpha \leftarrow 16$
$\theta \leftarrow 10$
$c \leftarrow 10.5$
for p *in products* **do**
  for s *in range*(θ) **do**
    $q \leftarrow (1, s+\alpha)$
    $d \leftarrow min(c_{step} + s, max\_steps)$
    $j \leftarrow (0.5, cprices[p])$
    $post(CFP(p, j, q, d, c))$
  end
end

In this procedure q and j denote the lower and upper boundaries for the quantity of the product (e.g., (1, 3) shows the negotiable quantities are between 1 and 3.) and for the price of the product respectively. d is the delivery time of the product, c_{step} is the current time step of the simulation, max_steps is the length of the simulation in terms of simulation steps, p is the product to be bought and c is the penalty incurred per product in case of insufficient products breach is committed by the seller.

- **Determining Reservation Value for Negotiation Strategy:**

During negotiations, the reservation value is the minimum acceptable utility for an offer. For selling negotiations, the reservation value of a particular product is updated based on the average inventory change of the product per simulation step. At the end of simulation step, when our agent acts as a seller, the reservation value of the product is updated as denoted in Algorithm 2.

Algorithm 2: Reservation value when our agent acts as the seller during negotiations.

1 $\delta \leftarrow 0.01$
2 **for** p *in products* **do**
3   $\Delta \leftarrow avg\_supplies[p] - avg\_demands[p]$
4   **if** $\Delta \neq 0$ **then**
5   **if** $\Delta < 0$ *and* $|amount[p]/\Delta| > r_{step}$ **then**
6       $r[p] \leftarrow min(r[p] - \delta * \Delta, 0)$
7   **end**
8   **else**
9       $r[p] \leftarrow max(r[p] + \delta * \Delta, 1)$
10   **end**
11   **end**
12 **end**

where for a product p, Δ denotes the difference between the average inflow and outflow of product p per step, which is equal to the average inventory change. In line 5, it is checked if the amount of product p in inventory is in decrease and is expected to be depleted before the end of the simulation (i.e when $|amount[p]/\Delta| > r_{step}$ where $amount[p]$ is the amount of product p in inventory and r_{step} is the remaining simulation steps left to the end of the simulation). The reservation value of product p ($r[p]$) for selling negotiations is decreased by $\delta * \Delta$ if the condition in line 5 is satisfied; otherwise it is increased (Line 9). The motivation behind decreasing the reservation value is to allow agents to concede more in order to increase the number of successful selling negotiations (i.e. minimizing the scrap products at the end of the negotiation). If there are no products expected to remain at the end of the negotiation, the reservation value is increased. Hence, the agent can maximize the utility by making more profit, at a reasonable risk of more negotiation failures.

The reservation value for buying negotiations for product p is determined as follows:

$$r_{p,t+1} = \begin{cases} r_{p,t} + \gamma & \text{Failure} \\ r_{p,t} - \theta & \text{Otherwise} \end{cases} \tag{9}$$

where $r_{b,t+1}$, and $r_{b,t}$ denote the updated and former reservation value respectively. In case a negotiation fails, the reservation value is increased by $\gamma = 0.01$; otherwise, decreased by $\theta = 0.001$. The adaptive structure of the reservation value prevents the agent to act too greedy or generous during the negotiations.

- **Making an Offer (Offering Strategy):** Our agent adopts a time-based concession strategy (Faratin et al., 1998) to make its offers during the negotiation. According to the time based concession strategy, the agent monotonically concedes over time. The target utility of the current offer is calculated as follows:

$$t_u = 1 + (r - 1) * t^z \tag{10}$$

where,

t_u : Target utility

r : Reservation value

t : Normalized timeline. It takes a value between 0 and 1, where 0 represents the beginning time and 1 denotes the timeline reaching the deadline.

z : Concession coefficient, z=10 for our agent

When our agent makes an offer, it first calculates the target utility. Among all possible offers, the offer with the smallest absolute value difference between the utility of the offer and target utility, is offered.

- **Deciding whether or not to Accept Opponent's Offer:** During the negotiation, our agent adopts AC_{next} acceptance strategy (Baarslag et al., 2014). If the utility value of the given offer is greater than or equal to utility of its next offer, opponent's offer is accepted; otherwise, it is rejected.
- **Scheduling Production:** At each simulation step, all idle production lines are scheduled to produce output products when there are enough input products in the inventory. When there are less input products than the number of idle production lines, all input products in the inventory is used for production.

4 EVALUATION

In the experiment, we have evaluated the performance of our agent based on the score (see Equation 5) gained at the end of simulations. The performance of our agent is compared with the performance of the SCML league winner agents in ANAC namely *SAHA*, *F2J*, *IFFM*, and the greedy factory manager agent provided by the ANAC organizers. In the simulation environments, there are multiple factory agents. In the current set up, we can specify agents strategies to be compared for only two factory agents and the rest of the factory agents by default are played by the greedy factory agents whose score is not taken into account. In our evaluation, we use the same simulation parameters with the ANAC setup except the number of simulation. We set the number of simulation steps as 150 in order to analyse more interactions while it is a random number between 50 and 100 in the competition.

4.1 Simulation Parameters

The simulation parameters determine the initial setup of the supply chain environment. The values of the simulation parameters in the experiments are specified below:

- Type of raw materials : 1
- Number of intermediate products : uniform(1,4)
- Number of final products : 1
- Number of miners : 5
- Number of consumers : 5
- Starting balance : 1000
- Production line count : 10
- Production cost : uniform(1,4)
- Amount of manufacturing process inputs : 1
- Amount of manufacturing process outputs : 1
- Time required for manufacturing process : 1 step

4.2 World Parameters

The world parameters determine the rules for negotiations, simulation length and several other rules for the simulation. The parameters for the simulation in our experiments is shown below:

- Number of simulation steps : 150
- Simulation time limit : 7200 Secs
- Negotiation time limit : 120 Secs
- Negotiation rounds limit : 20

- Negotiation time limit for each round : 10 Secs
- Negotiation speed multiplier : 21
- Immediate negotiations : No
- Default grace period for contract signing : 1
- Transportation delay : 0
- Negotiable penalties : Yes
- Allow breach renegotiations : Yes
- Global breach penalty : 0.02
- Base insurance premium : 0.1

4.3 Experiment Results

As we mentioned before, we have tested the performance of our agent by running simulations with the agent provided by the organizing committee namely, greedy factory manager agent (GFM), and top performing agents in the competition specifically, *SAHA agent*, *IFFM agent*, and *F2J agent*.

To evaluate the performance of our agent, we have compared the mean scores of each agent, the number of times they outperformed their opponents, and the number of times each agent bankrupted at the end of the negotiation. For each agent pairs, we ran 10 different simulations (e.g. different product costs, catalogue costs, and number of intermediate products etc.) and calculate the mean score for each agent. Furthermore, we applied statistical significance test on the score data to check whether the medians of data are significantly different. It is worth noting that we applied the Smirnov-Kolmogorov test to see whether the data follows a normal distribution - which is a requirement for t test. Since our data is not distributed normally, we adopt a non-parametric statistical significance test namely Wilcoxon signed-rank test with a significance level of 0.01.

Table 1 shows the average scores of each agent with their standard deviations, and the number of times they won and bankrupted for each agent pair out of 10 simulations. It can be obviously observed that *SAHA* agent outperforms *IFFM* and *F2J* significantly according to the average score although its performance varies a lot (high standard deviation). It also outperformed our agent in pairwise comparison but the performance difference is not as much as others. As far as the number of wins are concerned, our agent outperformed *SAHA* agent (7 versus 3). When we analyzed the results in a detailed way, we observed that SAHA performed better with respect to the average score because when it beats our agent, the score difference is tremendous compared to ours. That is why it has a higher average score although we won more negotiations. It earns a lot due to its strategic pricing approach aiming at exploiting consumers while we stick on the catalogue price and aim to earn from the demand.

In pairwise comparison, our agent outperformed all other agents except *SAHA* agent according to average scores. It is seen that we are the winner when we negotiate with them (10 wins out of 10 runs). In the following sections, we elaborately presents the results of 10 negotiations in which our agent negotiate with each agent separately.

4.3.1 Greedy Factory Manager Agent

Our agent against greedy factory manager agent (GFM) achieved a mean score around 13.5 meaning that our agent's funds at the end of simulation was on average 13.5 times higher than the initial, while the greedy factory manager agent has an average score of -1, which is the possible lowest score in the simulation. Figure 2 shows the score of each agent per each simulation runs. As seen from the bar chart, our agent outperformed the opponent in all simulation runs. It is not a surprising outcome since that agent is not very sophisticated agent.

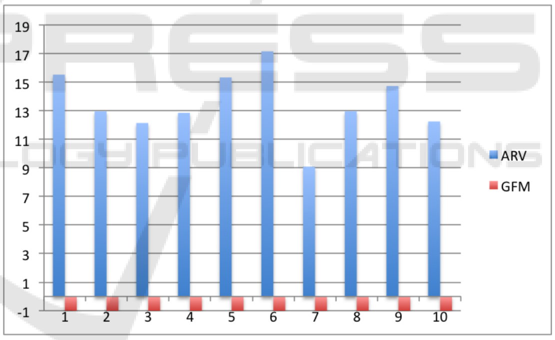

Figure 2: Scores of ARV and GFM for 10 negotiations.

When we apply the statistical tests, p value is 0.00512 and w value is 0. Because the p value is less than the significance level 0.01 and w score is less than the critical w score 5, the null hypothesis is rejected. That means the medians of the distributions differ significantly under the significance level 0.01.

4.3.2 SAHA Agent

The average score of our agent against SAHA agent was 10.94 while the SAHA agent got the average score of 22.24. Figure 3 depicts the score of each agent per simulation runs. We can observe that our agent outperformed the SAHA agent 7 times out of 10 runs. We have found out that the cases SAHA agent outperformed our agent when the SAHA agent

Table 1: Performance Comparison of Each Agent Pairs.

Agents	Avg Score	St dev	Bankrupts	Wins
SAHA/IFFM	**183.37** / 0	296.43 / 0.12	0/0	5/5
SAHA vs F2J	**32.94** / -0.02	95.36 / 0.07	0/0	4/6
F2J/IFFM	**0.76** / -0.06	0.19 / 0.88	0/0	3/7
ARV/GFM	**13.5**/-1	2.27/0	0/10	10/0
ARV/SAHA	10.94/**22.24**	5.26/53.49	0/5	7/3
ARV/IFFM	**8.85**/-0.06	2.48/0.04	0/0	10/0
ARV/F2J	**10.65**/-1	4.01/0	0/10	10/0

could run the manufacturing process for producing final products to sell to the consumers. SAHA agent exploited consumers to sell many products at a high price while our agent charged consumers at the catalog price of products at most even when the selling negotiation success rate was high, SAHA agent performed better in this scenario.

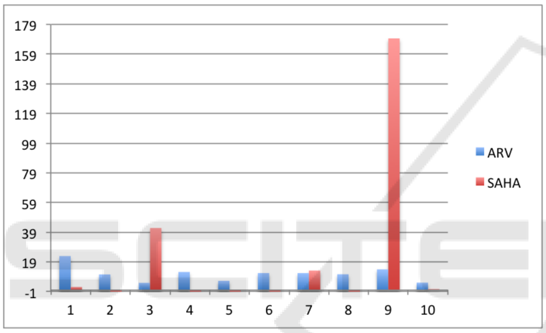

Figure 3: Scores of ARV and SAHA in 10 negotiations.

When we apply a statistical test, we see p = 0.44726 and w=20. Because the p value is greater than the significance level 0.01, the null hypothesis is failed to be rejected.

4.3.3 IFFM Agent

Our agent got an average score of 8.85 while the insurance fraud agent got an average of -0.06. Figure 4 depicts the score of each agent per simulation runs. For all of the simulation runs, our agent got higher score compared to the opponent.

Unlike SAHA agent, *IFFM* agent did not bankrupt at all. The test statistic values of Wilcoxon signed-ranks test is p = 0.00512 and w = 0 with a critical w value 5. Since p value is less than the significance level 0.01, the null hypothesis is rejected.

4.3.4 F2J Agent

Our agent got an average score of 10.65 while F2J agent got -1 and bankrupted in all of the simulation. Figure 5 shows the score of each agent per simulation runs. As seen from the chart, our agent beats in all runs.

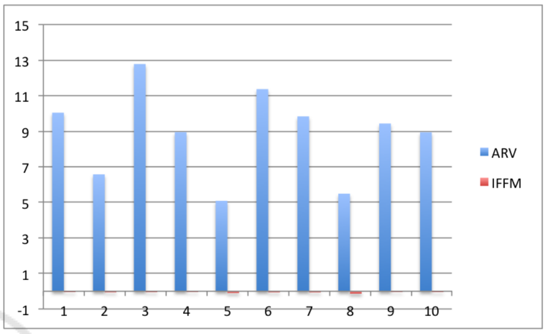

Figure 4: Scores of ARV and IFFM in 10 negotiations.

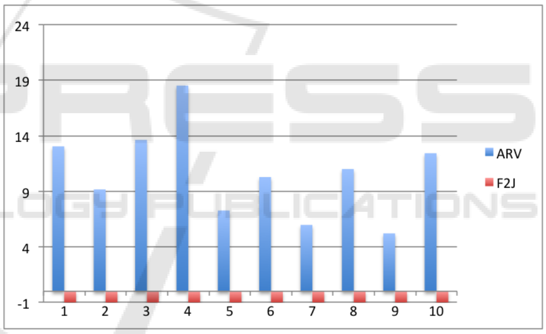

Figure 5: Scores of ARV and F2J in 10 negotiations.

The test statistic scores for the final scores was p=0.00512 and w=0 where the critical value of w is 5. Because p value is smaller than the significance level 0.01 and w is smaller than the critical value, the null hypothesis is rejected.

5 RELATED WORK

In the recent years, plenty of researches have been conducted for concurrent bilateral negotiations in supply chain management. To do so, researchers have worked on various supply chain models which have some similarities and differences with our study.

A negotiation-based multi-agentsystem for supply chain managementForget *et al.* developed a system that can coordinate agents in complex supply chain management environments with multi-behavior agents (Forget and CIRRELT., 2008). In their study, agents can negotiate on quantity, price, and delivery time similar to the SCML. To simulate the supply chain environment, they developed an agent based platform by emulating a lumber supply chain. In their study, different types of negotiations namely collaborative one-to-one, collaborative one-to-many, adversarial one-to-one, and adversarial one-to-many are analyzed. In the collaborative negotiation case, the mutual benefit of both parties is the concern while in adversarial case (i.e. the individual utility case), the agents try to maximize their own utility only. In our case, all negotiations in the simulation are one-to-one negotiations where agents have the discretion to act adversarial or collaborative by defining utility functions and negotiation strategies accordingly. We designed utility functions for buying and selling negotiations which remain robust because they are adjusted based on the inventory changes and negotiation results (i.e success or failure of the negotiations).

Lin *et al.* developed a Multi-Agent system to improve the order fulfillment process (OFP) in the Supply Chain System (Lin and Lin, 2004). An OFP is the process of receiving the order, producing it, and delivering the product to the customer. The OFPs are assumed to be given in their study while in our study, OFPs arise after the agents reach agreements through negotiations and then sign contracts. They modelled the order fulfillment process (OFP) as a distributed constraint satisfaction problem (DCSP) in which the constraints are distributed in all agents and to solve DCSP. Their contribution was combining DCSP with peer to peer negotiation approach in which the agents negotiated on the constraints in order to find a solution for their inter-agent constraints. In other words, peer to peer negotiation is the approach they used to solve the DCSP which represents the OFP in supply chain. In our study, peer to peer negotiations are used to sell/buy goods while maximizing at the profits at the end of the simulation. To test the performance of their system, they used performance metrics such order fulfillment rate and cycle time, which are the main concerns of OFP. While in our study, we evaluated the performance based on the final profits.

Chen *et al.* designed a negotiation based dynamic multi-agent system for supply chain environments in which the entities, represented as agents, can join or leave the environment and there are multiple final products where agents do negotiations for transactions (Chen et al., 1999). The agents have constraints such as delivery time, quantity and price and constraint resolution forms their acceptance strategy during one-to-many negotiations, where the agent negotiates for buying goods from many suppliers in the same thread and offer(s) of other party is accepted if the constraints are satisfied. In our study on the other hand there is one final product which are obtained by processing input products, the factory managers don't have constraints but has the aim to maximize their profit and therefore design their utility function dynamic to changes in market to make smart transactions.

Fink developed a multi agent collaboration system in supply chain management (Fink, 2004). In their work, a set of potential contracts between both parties are assumed to be given and during negotiations a mediator agent generates candidate contracts transparent to both parties which can accept or reject the mediator's offer and the agreement is reached when both parties accept the offer. The motivation behind this study is to reach mutual agreements for both agents while in our study the parties do bilateral negotiations to exchange offers where the main concern is maximizing the individual benefit.

Williams *et al.* introduced a novel strategy that enables agents to negotiate concurrently with multiple unknown opponents (Williams et al., 2012). In their work, they implement a coordinator entity which records the observations of ongoing negotiations and define a concurrent negotiation strategy based on the opponents' probabilistic actions. In our work, instead of coordinating multiple negotiations, we update the reservation values for the further negotiations based on the negotiation results and inventory changes in order to define a robust strategy in supply chain environment.

6 CONCLUSION

We developed a factory manager agent for Supply Chain Management League in ANAC where suppliers, factories and consumers interact with each other through negotiations. The proposed agent adopts adaptive adjustment of reservation value with respect to changes in its inventory and negotiates accordingly. Moreover, it seeks more negotiation opportunities to make some profits. To do so, it buys some products which are not processed in its manufacturing lines and sells them to customers with profit. We have evaluated the performance of the proposed agent by comparing it with the top performing agents in ANAC with respect to a number metrics. The experimental results showed that our agent outperformed them.

In the current work, we have not designed strategic pricing approaches; instead stick on the catalog price of products and aimed to gain from demands. As for future work, we are planning to incorporate strategic pricing as SAHA agent did. Furthermore, it would be interesting to predict consumer's demand in advance based on the past interactions. In the current set up, there is only one production line in which one input material is processed to produce a single product. It would be more challenging if the factory agent had production lines producing different products and decided on which products it should invest more.

REFERENCES

Aydoğan, R., Festen, D., Hindriks, K. V., and Jonker, C. M. (2017). Alternating offers protocols for multilateral negotiation. In *Modern Approaches to Agent-based Complex Automated Negotiation*, pages 153–167. Springer.

Aydoğan, R. and Yolum, P. (2012). Learning opponent's preferences for effective negotiation: an approach based on concept learning. *Autonomous Agents and Multi-Agent Systems*, 24(1):104–140.

Baarslag, T., Hindriks, K., and Jonker, C. (2014). Effective acceptance conditions in real-time automated negotiation. *Decision Support Systems*, 60:68–77.

Chen, Y., Peng, Y., Finin, T., Labrou, Y., Cost, R., Chu, B., Yao, J., and Sun, R. (1999). A negotiation-based multi-agent system for supply chain management. *Proc. Agents'99 Workshop Agent- Based Decision-Support for Managing Internet-Enabled Supply Chain*, pages 15–23.

de Jonge, D., Baarslag, T., Aydoğan, R., Jonker, C., Fujita, K., and Ito, T. (2019). The challenge of negotiation in the game of diplomacy. In *Agreement Technologies*, pages 100–114, Cham. Springer International Publishing.

Faratin, P., Sierra, C., and Jennings, N. R. (1998). Negotiation decision functions for autonomous agents. *Robotics and Autonomous Systems*, 24(3):159 – 182. Multi-Agent Rationality.

Fatima, S., Kraus, S., and Wooldridge, M. (2014). *Principles of automated negotiation*. Cambridge University Press.

Fink, A. (2004). Supply chain coordination by means of automated negotiations. *37th Hawaii International Conference on System Sciences*, pages 243–265.

Forget, P. and CIRRELT. (2008). *Collaboration Agent-based Negotiation in Supply Chain Planning Using Multi-behaviour Agents*. CIRRELT (Collection). CIRRELT.

Fujita, K. (2014). Automated strategy adaptation for multi-times bilateral closed negotiations. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pages 1509–1510. International Foundation for Autonomous Agents and Multiagent Systems.

Hindriks, K., Jonker, C., and Tykhonov, D. (2009). The benefits of opponent models in negotiation. In *2009 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, pages 439–444.

Ito, T., Hattori, H., and Klein, M. (2007). Multi-issue negotiation protocol for agents: Exploring nonlinear utility spaces. In *IJCAI*, volume 7, pages 1347–1352.

Jonker, C. M., Aydoğan, R., Baarslag, T., Fujita, K., Ito, T., and Hindriks, K. V. (2017). Automated negotiating agents competition (ANAC). In *AAAI*, pages 5070–5072.

Lin, F. and Lin, Y. (2004). Integrating multi-agent negotiation to resolve constraints in fulfilling supply chain orders. *Electronic Commerce Research and Applications*, 5:313–322.

Lin, R., Kraus, S., Baarslag, T., Tykhonov, D., Hindriks, K., and Jonker, C. M. (2014). Genius: An integrated environment for supporting the design of generic automated negotiators. *Computational Intelligence*, 30(1):48–70.

Mbang, A. (2011). A new introduction to supply chains and supply chain management: Definitions and theories perspective. *International Business Research*, 5:194–207.

Mell, J., Gratch, J., Baarslag, T., Aydoğan, R., and Jonker, C. M. (2018). Results of the first annual human-agent league of the automated negotiating agents competition. In *Proceedings of the 18th International Conference on Intelligent Virtual Agents*, pages 23–28. ACM.

Mohammad, Y., Areyan Viqueira, E., Ayerza, N. A., Greenwald, A., Nakadai, S., and Morinaga, S. (2019). Supply chain management world - A benchmark environment for situated negotiations. In *PRIMA 2019: Principles and Practice of Multi-Agent Systems - 22nd International Conference, Turin, Italy, October 28-31, 2019, Proceedings*, pages 153–169.

Mohammed, Y., Greenwald, A., Fujita, K., Morinaga, S., and Nakadai, S. (2019). *SCM World:Automated Negotiating Agents Competition*.

Sanchez-Anguix, V., Aydogan, R., Julian, V., and Jonker, C. (2014). Unanimously acceptable agreements for negotiation teams in unpredictable domains. *Electronic Commerce Research and Applications*, 13(4):243–265.

Williams, C. R., Robu, V., Gerding, E., and Jennings, N. R. (2012). Negotiating concurrently with unknown opponents in complex, real-time domains. In *ECAI*, pages 834–839.