Supply Chain Management League

Merve Dogan
Computer Science
Ozyegin University
Istanbul, Turkey
merve.dogan@ozu.edu.tr

Aysenur Emel Yargic Oksuz
Industrial Engineer
Ozyegin University
Istanbul, Turkey
aysenur.oksuz@ozu.edu.tr

Neva Yaren Bulut
Electrical and Electronics Engineering
Ozyegin University
Istanbul, Turkey
yaren.bulut@ozu.edu.tr

I. Final Report

Abstract - This final report presents the strategic design and implementation of Group5, an enhanced version of the Simple Agent tailored for the Supply Chain Management League (SCML) 2024 competition. The report begins with an introduction to supply chain management and the SCML-OneShot world, outlining the objectives and structure of the competition. It then delves into the methodology behind the agent's development, emphasizing a focus on quantity-based negotiations over price concessions and adaptive negotiation tactics.

The literature review explores key components of automated negotiation systems, including acceptance, bidding, and opponent modeling strategies. Insights from studies by Baarslag et al., Keskin et al., and others inform the agent's design choices, such as incorporating the Nice Tit-for-Tat strategy and considering opponent behavior in negotiations.

Strategic objectives for Group5 include maximizing sales and supply fulfillment while prioritizing cooperative and compromising negotiation tactics. The bidding strategy emphasizes offering quantities aligned with sales and supply needs, while the acceptance strategy evaluates offers based on quantity rather than price.

The methodology section outlines the agent's decision-making process, including the consideration of opponent bids and the implementation of adaptive strategies. Strategies such as the Tit-for-Tat approach and Bayesian opponent modeling contribute to fair and optimal negotiation outcomes.

In summary, Group5 represents a strategic approach to SCML 2024, aiming to optimize negotiation outcomes through adaptive and cooperative tactics. The report provides valuable insights into the design and implementation of negotiation agents in dynamic and competitive environments.

A. *Introduction*

Supply Chain Management is the process of managing the flow of goods and services throughout a business. It includes everything from getting raw materials to production and finally delivering the finished product to the customer.

Supply Chain Management League (SCML) is an automated negotiation competition. In this competition, all the participants are expected to design and implement their own negotiation strategy for the supply chain management. SCM OneShot simulates a supply chain with multiple autonomous agent-managed factories. Factories negotiate to buy raw mate-

Fig. 1. Supply Chain Management Flow

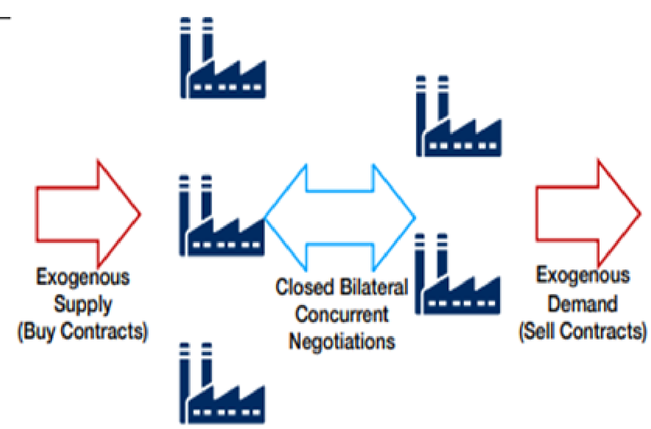

Fig. 2. SCML-OneShot world. Each factory is represented by an agent, whose goal to negotiate buy and sell contracts that maximize profits.

rials and sell final products, aiming to turn a profit. The agent with the highest total profit summed over all days, and then averaged across multiple simulations, wins.

In the competition for 2024, the components of the environment are products as raw materials, intermediate products, and final products. There is a production that consists of two manufacturing processes, converting materials to intermediate and final products. There are factories that are organized in two layers (L0 and L1), negotiating to buy and sell products. And of course, there are agents who function as factory managers, negotiating and managing contracts.

There is a negotiation protocol which is a variant of the bilateral alternating offers protocol. And the utility function represents agent profits, including revenues, costs, and penalties. Another component, trading price is a weighted average of past prices, affecting negotiation range and penalties. Lastly, the bulletin board provides static and dynamic information about the game environment and factories. The difference of one-shot SCML from the previous year 2023 is it includes a framework for building reinforcement learning agents. It is not a must but can be easier for the competitors.

In the implementation part we are going to do the following:

1) Initialize our agent.
2) Repeat the following steps
    a) Update product trading prices.
    b) Assign contracts and costs to agents.
    c) Execute agents' before-step functions.
    d) Run negotiations until completion.
    e) Calculate agents' profits and update balances.
    f) Execute agents' step functions.

Our agent, which we call Group5, is an enhanced version of the Simple Agent, which is better for SCML 2024 since the focus of the competition has shifted from price to quantity in terms of negotiation issues [1]. In our research, we have realized that many agents of the past year have been an improvement on the built-in agents of the SCML league by improving the quantity being offered or the adaptiveness of the agent. Thus, we are proposing an enhanced version of the built-in SimpleAgent which aims to secure its sales and supply needs while prioritizing quantity considerations over price, thereby improving negotiation outcomes and overall performance. Our agent will also take some of the design strategies from the CCAgent, which was a finalist in last year's competition, and which was also an improvement on the SimpleAgent, with an addition in cooperation and compromising. We also consider improving the behavior-dependent strategy on top of CCAgent, which we describe in more detail in the Methodology section.

B. *Literature Review*

Automated negotiation systems have become indispensable tools in today's complex supply chain management landscape. These systems rely on sophisticated strategies to facilitate transactions and optimize outcomes within dynamic and competitive environments. Three key components of these strategies are acceptance, bidding, and opponent modeling.

1-Acceptance Strategy

Acceptance strategies serve as guiding principles for automated negotiation agents, determining when they should agree to proposed deals. These strategies help agents evaluate offers based on various factors to maximize their utility and achieve favorable outcomes. Effective acceptance strategies are pivotal for the efficiency and success of negotiations, ensuring that agents make optimal decisions in accepting or rejecting offers.

Tim Baarslag et. al. emphasizes the crucial role of acceptance conditions in real-time automated negotiations. These conditions, which determine whether an agent accepts or rejects an offer, are influenced by factors like agent preferences, negotiation context, opponent behavior, and overall environment. The authors suggest different strategies for defining effective acceptance conditions. One approach involves utility-based criteria, where agents evaluate offers based on their preferences and expected opponent utility. Another strategy integrates fairness considerations to ensure perceived fairness in negotiated outcomes. Adaptability is also highlighted as crucial, allowing agents to dynamically adjust acceptance thresholds based on evolving negotiation dynamics. By carefully designing and adapting acceptance conditions, agents can enhance negotiation performance, achieve mutually beneficial agreements, and contribute to the efficiency of supply chain management operations. In this paper, they mention three acceptance condition: Based on utility, based on remaining time and based on a threshold. Acceptance based on utility approach prioritizes offers that maximize the agent's utility or benefit, considering factors such as cost, value, and preference. By assessing the utility of incoming offers, the agent can make informed decisions to accept or reject proposals during negotiation. Acceptance based on the remaining time approach enables the agent to make decisions that optimize outcomes within the given time frame, balancing efficiency and effectiveness in the negotiation process. In the last condition which is acceptance based on a threshold, the agent evaluates incoming offers against this threshold, accepting proposals that meet or exceed the predefined criteria. By establishing clear acceptance thresholds, the agent can focus on offers that align with its objectives and preferences. [8]

In another study by T. Baarslag et. al. proposed a strategy which aims to maximize utility by accepting offers that provide the greatest benefit relative to the agent's reservation value, taking into account the uncertainty associated with incomplete information. This strategy optimizes utility by strategically accepting offers based on incomplete information. It improves negotiation outcomes by adapting to uncertainty and variability in negotiation settings and provides a systematic approach for determining acceptance thresholds in automated negotiation scenarios. But on the other side its complexity may increase in scenarios with multiple issues or agents, requiring sophisticated algorithms for computation. [13]

S. Kawaguchi, et. al. proposed a compromising strategy involving estimating the maximum utility that can be achieved in a negotiation scenario. By determining this maximum utility, the negotiating agent aims to strike a balance between concession and assertiveness to reach mutually beneficial agreements. The strategy incorporates elements of compromise, adaptability, and strategic decision-making to optimize negotiation outcomes. It allows the negotiating agent to adapt to varying negotiation contexts and counterpart behaviors, enhancing its ability to achieve favorable outcomes in diverse scenarios. It promotes cooperation and collaboration by encouraging concessions while striving to maximize utility for both sides. But it may require sophisticated algorithms and computational resources to estimate maximum utility accurately and execute strategic decisions effectively. The effectiveness of the strategy may be impacted by uncertainties in the negotiation environment, such as incomplete information. [14]

2-Bidding Strategy

Bidding strategies form the backbone of negotiation processes, dictating how agents make initial offers and respond to counter offers from their counterparts. These strategies cover different aspects of a range of considerations, including offer formulation, counter offer computation, and utility optimization. By carefully crafting bidding strategies, agents can enhance their negotiating position, improve their chances of securing favorable deals, and ultimately, maximize their profitability.

The provided excerpts offer insights into various bidding strategies employed in automated negotiation scenarios, showcasing the diversity of approaches aimed at optimizing negotiation outcomes. Here's a review of each strategy outlined:

M. O. Keskin et. al. propose a Hybrid Strategy which introduces novelty by combining behavioral and time-based tactics to adaptively respond to opponents' bidding patterns. By prioritizing recent bid changes within a window of opponent bids, the Solver Agent aims to estimate utility shifts effectively. This approach demonstrates a proactive stance toward opponent modeling, seeking to capitalize on evolving negotiation dynamics to maximize utility. [9]

R. Ros et.all introduce NegoEngine, emphasizes concessions and trade-offs to navigate negotiation spaces effectively. By considering multiple decision variables and incorporating fairness considerations, NegoEngine aims to propose offers that strike a balance between conflicting interests. The strategy's focus on understanding opponents' preferences reflects a nuanced approach to negotiation, aiming to craft proposals that are mutually acceptable and advantageous. [10]

T. Baarslag et.al. introduces a negotiation strategy known as the Nice Tit for Tat Agent, which was developed for the Second Automated Negotiating Agent Competition (ANAC2011). The Nice Tit for Tat Agent strategy is based on the Tit for Tat principle, which involves cooperating on the first move and then mirroring the opponent's actions in the preceding round. This strategy relies on reciprocity for cooperation. We have incorporated the strategy used in this paper.[11]

R. Aydogan et. al. investigate why individuals might be hesitant or unwilling to engage in negotiation interactions with robotic counterparts. They delve into psychological and sociological factors that influence human trust and acceptance of robotic negotiators. The main reasons are lack of empathy, uncertainty about the robot's decision-making process, and skepticism about its ability to understand complex human needs and emotions. They discuss how to improve the acceptance of robotic negotiators. They propose enhancing the robot's communication skills, implementing transparent decision-making mechanisms, and incorporating human-like qualities to foster trust and rapport [12].

3-Opponent Modeling

Opponent modeling plays a critical role in automated negotiation by enabling agents to predict and counter the strategies employed by their counterparts. By analyzing past interactions and observable behaviors, agents can develop models of their opponents' preferences, tendencies, and decision-making processes. This information empowers agents to adapt their negotiation strategies dynamically, anticipate their opponent's moves, and gain a competitive edge in the negotiation process.

In the literature, there are several techniques proposed for opponent models like time series predictive models, prediction-based strategy with dynamically changing reservation value, taking inventory changes into account, Bayesian learning, frequency-based techniques, which analyze past behaviors to predict future actions, etc. All of them have advantages and disadvantages of course. G. Yesevi et. all propose the time series predictive models which can capture temporal patterns in negotiation behavior, allowing negotiation agents to adapt their strategies dynamically. On the other side, obtaining sufficient and representative data may be challenging, particularly in complex negotiation scenarios with limited historical records [4]. Prediction-based strategy with dynamically changing reservation value by A. S. Gear et.al. enables proactive adaptation of strategies and increases efficiency by anticipating opponents' moves and adjusting negotiation tactics accordingly. However, the effectiveness may diminish in highly uncertain or volatile negotiation environments [5]. C. Yu et. al. propose a Bayesian learning which enables agents to learn from experience and improve negotiation performance over time. However, it requires a sufficient amount of negotiation data for accurate Bayesian updating, which may not always be available. Its complexity may increase with the number of negotiation variables and the size of the negotiation space [15]. In general, opponent modeling relies heavily on frequency-based techniques, which analyze past behaviors to predict future actions. However, O. Tunalı et. al. say this approach may not adequately capture the complexity of negotiation scenarios, where various factors beyond historical frequencies influence decision-making [16].

C. Negotiation Strategies

The primary strategic objectives of our agent include maximizing the fulfillment of sales and supply needs by prioritizing quantity-based negotiations over price concessions. And implementing adaptive negotiation tactics to respond effectively to opponent offers, we plan to convey a strategy that is cooperative and compromising. When a negotiating agent does not take their opponent's moves into account, the negotiation may end up with an unfortunate agreement for itself. It is essential to consider the opponent's attitude during the negotiation and act accordingly [2].

We plan to incorporate the Nice Tit-for-Tat strategy, which is a Behaviour-dependent bidding strategy based on the principle of Tit for Tat that cooperates on the first move and then mirrors whatever the other player did in the preceding round[11]. To adapt to opponent behavior and negotiate effectively, our agent incorporates adaptive negotiation tactics based on the Nice Tit-for-Tat strategy. So, it will start with cooperative behavior, but retaliate against uncooperative opponents by mirroring their behavior. This promotes fairness, reciprocity, and mutually beneficial agreements. Agents participating in SCML do not have direct access to the opponent's utility function and they typically rely on their own utility functions to make decisions while attempting to infer aspects of the opponent's preferences or behaviors based on past interactions or observed patterns. As a result, agents in SCML often employ strategies that prioritize their own objectives and adapt based on observed outcomes and opponent behavior rather than relying on explicit knowledge of the opponent's utility function. This approach mirrors real-world negotiation dynamics where agents must make decisions based on incomplete or uncertain information about their counterparts. Thus, our agent will reciprocate according to the agent's own utility function. Since, the agent has full knowledge of its own utility function, when the opponent submits a bid advantageous to the agent, the agent reciprocates by offering a bid that yields lower utility for itself.

1-Bidding Strategy

Our agent's bidding strategy focuses on offering quantities that align with its sales and supply needs, rather than prioritizing price concessions. Our aim is to create offers that the opponent will be more likely to accept. Hence, our bidding strategy is cooperative. Our strategy will be adaptive and will bid according to the changes in the opponent dynamically. So, it will start with the best offer and make the offers based on the behaviour-dependent tactic as explained in detail in the next section.

2-Acceptance Strategy

Our agent's acceptance strategy is adaptive and responsive to opponent offers. We use ACnext acceptance criteria for our acceptance strategy in which: agent A will accept when the utility for the opponent's last offer at time t is greater than the value of the offer agent A is ready to send out at time t[8]. The acceptance condition above depends on the agent's upcoming offer. In the next section, we explain this implementation of the strategy in details.

D. Methodology

While building our agent, first we tried to answer the question: How do we humans negotiate? We came up with couple important points that we tried to incorporate into our agent design. First one is, market research. Secondly, we know that the first offer is the most important one for us, humans as we negotiate. Negotiators often exhibit flexibility by being willing to compromise on certain aspects of the deal, such as price or quantity, in exchange for concessions from the other party. This involves identifying and prioritizing preferences and being open to creative solutions. Also, establishing a positive relationship with the other party can facilitate smoother negotiations. We, humans usually do this by often engaging in small talk, active listening, and empathy to build rapport and establish trust, which can lead to more favorable outcomes. Another point while we negotiate is if someone offers us more quantity than we need, we reject. And pricewise, we would offer the minimum price if we are buying, and maximum price if we are selling. We questioned: when we are offered a quantity and a price, do we accept if the quantity is lower than or equal to our need if the price is low? Do we pay attention to price? As an answer, we found out that the price does not really matter that much in the settings for SCML 2024 OneShot because the price range is limited to only two consecutive values (e.g. (9, 10)).

We implemented our agent by following the implementation of BetterSyncAgent which is an extension of OneShotSyncAgent, as described in the documentation [17]. As in the ”firstproposals” method of BetterSyncAgent, our agent first randomly distribute our needs over its partners with best price for us. Finding best price is as suggested:

$$bestprice = \begin{cases} pmin, & \text{if buying} \\ pmax, & \text{if selling} \end{cases}$$

As explained in the SCML documentation: When countering offers, we should take into account the history of negotiation with each partner (in this round and previously) to make a more meaningful distribution of quantities over partners.

In the original BetterSyncAgent, this is just random. So, we are keeping a variable opponents_last_bid as a dictionary which keeps track of the history of offers from the partners. Hence, by iterating through the partners, it checks if a partner has made any offer yet. If so, the agent mirrors their bid (regards to our nice tit for tat strategy). We do this by checking via a threshold multiplied by the quantity. And we mirror the bid if it is above that threshold quantity. Otherwise, we make a new offer: A tuple (q, s, p) where q represents the quantity, s represents the step value, and p represents the price value. And if no offers made yet, our agent just generates a new one as described above.

In the following rounds, after the agent receives offers from the opponents, it makes the following bids. According to our Tit for Tat strategy to mimick the opponent's behaviour to some extent, we calculate the utility changes in its opponent's subsequent offers regarding its utility as seen (1)

$$\Delta U = U(O_h^t cur) - U(O_h^t prev)$$

$$TU = U(O_j^t prev) - \Delta U * \mu$$

where $U(O_h^t cur)$ and $U(O_h^t prev)$ denote the utility of the opponent's current and previous offers for our agent, respectively. A parameter, μ, estimates target utility TU as seen from the formula, where $U(O_j^t prev)$ denotes the utility of the agent's previous offer. The agent subtracts the scaled utility changes to mimic its opponent. The positive changes mean that the opponent concedes; hence, the agent should concede as well. It generates an offer whose utility is closest to the estimated target utility. A higher value of the utility parameter, μ, amplifies the effect of the difference in utility change ΔU, on the target utility. This means that even small changes in utility can lead to significant adjustments in the target utility. Also, a higher utility parameter makes the agent more responsive to changes in utility, allowing it to quickly adapt to evolving negotiation conditions. Conversely, reducing the utility parameter dampens the impact of changes in utility on the target utility. The agent becomes less sensitive to variations in utility and may maintain a more stable target utility despite fluctuations in the negotiation environment. In terms of adaptability, a lower utility parameter makes the agent less reactive to changes in utility, resulting in a more conservative and stable approach to setting the target utility.

Then the agents checks if the utility of the new offer meets the target utility. By checking the closest utility to the estimated target utility, agent says this is the offer for this partner and add that to its offers by opponents, otherwise it ignores that offer. We keep a utility for agent's previous offers as well to estimate the target utility, so we keep a variable for that here for the next round of offers.

When receiving offers in counter-all, we follow the BetterSyncAgent's implementation and treat suppliers and consumers independently as it is recommended even though it is not necessary for SCML-OneShot but it is a form of future-proofing that we get at a small cost. Then within the received offers in counter-all, we create a subsets of all offers from the partners, subset-offers. The agent checks whether the utility from these subsets is greater than the utility of the agent's next offer. If so, it accepts that subset of offers, o/w it makes a new offer. This is simple the ACnext strategy [8]. In ACnext strategy, we are using two parameters as described in the paper as follows:

$$AC_{next}(\alpha, \beta) = \alpha * U_A(x^t_{(B->A)}) + \beta >= U_A({x^{t\prime}}_{(A->B)})$$

Here we start with α, 1.2 and β 15. A higher value of α would give more weight to the utility from the subsets, the partners offers observed. This means the agent is more inclined to accept offers that are closer to the best offer it has received so far. Conversely, reducing α would reduce the influence of the utility from the opponents. This allows the agent to be more flexible in accepting offers that may not be as good as the best offer observed. On the other hand, a higher value of β would increase the threshold for accepting an offer. This means the agent requires a higher expected utility from the upcoming offer to accept it, while reducing β lowers the acceptance threshold, allowing the agent to accept offers with lower expected utility. Basically, by adjusting α and β, we can control the agent's behavior in terms of how it evaluates and accepts incoming offers.

However, the results received from ACnext strategy gave us the incentive to look for another approach. As Baarslag et. al. suggests, in The Nice Tit for Tat agent, the acceptance strategy called ACcombi, which the paper shows to work better than the majority of more simple generic conditions, can be incorporated[11]. Baarslag et al describes ACCombi as follows: in case the bidding strategy plans to propose a deal that is worse than the opponent's offer, we have reached a consensus with our opponent and we accept the offer. However, if there still exists a gap between our offer and time is short, the acceptance condition should wait for an offer that is not expected to improve in the remaining time. Thus ACcombi is designed to be a proper extension of ACnext, with adaptive behavior based on recent bidding behavior near the deadline.[8]

Also in counter-all method, we keep the quantity offered by the partners, as "opponents last bid" to be used in our cooperative moves to mimic the opponent's behaviour. We also keep the utilities for the offers accepted in this step in "utility opponent previous offers", so we can use it for the next calculation of the target utility for our tit-for tat strategy as mentioned above.

E. Evaluation

We compared our agent, Group5, to other agents. The table shows our results compared to other agents.

As it is seen in Table I, our agent performs better than the GentleS in terms of the metrics that we checked like mean, standard deviation, and first and third quartiles. There is a slight difference with the CCAgent.

If we check the score results, again we see that our agent is slightly different than SyncRandomOneShotAgent but performs better than the RandomOneShotAgent.

Even in some trials, our agent performed the best among these agents. It shows promising potential but might require further development to reach the top level.

When evaluating our agent's performance against different opponents, we observed varying acceptance rates as seen in Table 2. When running against the CCAgent and GentleS, finalists of the 2023 SCML competition, our agent achieved an acceptance rate of 0.4. However, when running against the SyncRandomOneShotAgent and RandomOneShotAgent, our agent's acceptance rate increased to 0.6, as shown in Table II. This suggests that our agent's negotiation strategy may be more effective or better aligned with the behavior of these opponents compared to the CCAgent. Further analysis and comparison of negotiation outcomes against different opponents can provide valuable insights into the strengths and weaknesses of my agent's negotiation approach.

F. Conclusion

For this year's SCML competition, first of all we analyzed the previous years' agents' with the best performance. Taking these agents into account we determined our strategy. Mainly, our strategy is based on price. If we are a buyer we would

Agents	Mean	Std	25%	75%
CCAgent	1.029177	0.092824	0.975626	1.098695
GentleS	0.815994	0.228949	0.604219	1.042455
Group5	0.992724	0.150833	0.920851	1.096905

Agents	Score
SyncRandomOneShotAgent	1.11947
Group5	1.05328
RandomOneShotAgent	0.93761

TABLE I
TEST RESULTS OF GROUP5

Agents	Acceptance Rate
SyncRandomOneShotAgent	0.4
RandomOneShotAgent	0.6

TABLE II
ACCEPTANCE RATE FOR GROUP5

like a lower price but if we are seller it's visa versa. After that our agent checks the utility to accept or reject the offer which is the ACnext strategy. We gave importance to quantity as well and keep it to be able to make logical moves against to opponent's behaviour. In the experiment part, we compared our Group5 with the other winner agents from 2021, 2022, and 2023 and showed that our Group5 performed better than QuantityOriented, PatientAgent.

In this project, we especially focused on the acceptance and bidding strategy but we did not change the opponent modeling which uses Bayesian model. In the future some other techniques like time series models or taking inventory level account can be used as an opponent model and see how it performs in this way.

REFERENCES

[1] https://scml.readthedocs.io/en/latest/tutorials/02.develop-agent-scml2024-oneshot.html

[2] R. Aydogan, O. Keskin and U. Cakan, "Would You Imagine Yourself Negotiating With a Robot, Jennifer? Why Not?," in IEEE Transactions on Human-Machine Systems, vol. 52, no. 1, pp. 41-51, Feb. 2022

[3] O. Tunalı, R. Aydogan, and V. Sanchez-Anguix, "Rethinking frequency opponent modeling in automated negotiation," in PRIMA 2017: Principles and Practice of Multi-Agent Systems. Springer International Publishing, 2017, pp. 263–279.

[4] G. Yesevi, M. O. Keskin, A. Dogru, and R. Aydogan, "Time series predictive models for opponent behavior modeling in bilateral negotiations," in PRIMA 2022: Principles and Practice of Multi-Agent Systems, 11 2022, pp. 381–398.

[5] A. S. Gear, K. Prakash, N. Singh, and P. Paruchuri, "PredictRV: A Prediction Based Strategy for Negotiations with Dynamically Changing Reservation Value," Group Decision and Negotiation: A Multidisciplinary Perspective, vol. 388, pp. 135–148, 2020. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7215186/

[6] C. O. B. Yavuz, C. Suslu, and R. Aydogan, "Taking Inventory Changes into ˘ Account While Negotiating in Supply Chain Management," in Proceedings of the 12th International Conference on Agents and Artificial Intelligence, vol. 1. SciTePress - Science and Technology Publications, 2020, pp. 94–103. [Online]. Available: http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0008976900940103

[7] C. Yu, F. Ren, and M. Zhang, "An adaptive bilateral negotiation model based on bayesian learning," in Complex Automated Negotiations: Theories, Models, and Software Competitions, ser. Studies in Computational Intelligence, T. Ito, M. Zhang, V. Robu, and T. Matsuo, Eds. Springer Berlin Heidelberg, 2013, vol. 435, pp. 75–93. http://dx.doi.org/10.1007/978-3-642-30737-9 5

[8] T. Baarslag, K. V. Hindriks, and C. M. Jonker, "Effective acceptance conditions in real-time automated negotiation," Decision Support Systems, vol. 60, pp. 68–77, Apr 2014. [Online]. Available: http: //dx.doi.org/10.1016/j.dss.2013.05.021

[9] M. O. Keskin, U. C. Akan, and R. Aydogan, "Solver agent: towards emotional and opponent-aware agent for human-robot negotiation," in Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems, 2021, pp. 1557–1559.

[10] R. Ros and C. Sierra, "A negotiation meta-strategy combining trade-off and concession moves," Autonomous Agents and Multi-Agent Systems, vol. 12, pp. 163–181, 2006. [Online]. Available: http://dx.doi.org/10.1007/s10458-006-5837-z

[11] Baarslag Tim , Hindriks Koen, Jonker Catholijn. (2013). A Tit for Tat Negotiation Strategy for Real-Time Bilateral Negotiations. 10.1007/978-3-642-30737-9-18.

[12] R. Aydogan, M. O. Keskin, and U. C. Akan, "Would you imagine yourself negotiating with a robot, jennifer? why not?" IEEE Trans. Hum. Mach. Syst., vol. 52, no. 1, pp. 41–51, 2022.

[13] T. Baarslag and K. V. Hindriks, "Accepting optimally in automated negotiation with incomplete information," in Proceedings of the 2013 International Conference on Autonomous Agents and Multi-agent Systems, ser. AAMAS '13. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2013, pp. 715–722. [Online]. Available: http://dl.acm.org/citation.cfm?id=2484920.2485033

[14] S. Kawaguchi, K. Fujita, and T. Ito, "Compromising strategy based on estimated maximum utility for automated negotiating agents," in New Trends in Agent-based Complex Automated Negotiations, Series of Studies in Computational Intelligence, T. Ito, M. Zhang, V. Robu, S. Fatima, and T. Matsuo, Eds. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 137–144. [Online]. Available: http://link.springer.com/content/pdf/10.10072F978-3-642-24696-8-8

[15] C. Yu, F. Ren, and M. Zhang, "An adaptive bilateral negotiation model based on bayesian learning," in Complex Automated Negotiations: Theories, Models, and Software Competitions, ser. Studies in Computational Intelligence, T. Ito, M. Zhang, V. Robu, and T. Matsuo, Eds. Springer Berlin

Heidelberg, 2013, vol. 435, pp. 75–93. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-30737-9-5

[16] O. Tunalı, R. Aydogan, and V. Sanchez-Anguix, "Rethinking frequency opponent modeling in automated negotiation," in PRIMA 2017: Principles and Practice of Multi-Agent Systems. Springer International Publishing, 2017, pp. 263–279.

[17] https://scml.readthedocs.io/en/latest/tutorials /02.develop_agent_scml2024_oneshot.htmla-better-syncagent