Intelligent Agents Coursework Report

Agent 26

Group 26 Hang Liu
University of Southampton
Southampton, Hampshire, UK

ABSTRACT

This report provides the design and analysis on negotiating agent 26 used in the coursework. After showing the result of the competition, it gives the evaluation and future development for this negotiating software program. In all, this report shows how the knowledge in the field about game theory and Negotiation used in competition.

ACM Reference Format:
Group 26 Hang Liu. 1997. Intelligent Agents Coursework Report: Agent 26. In *Proceedings of ACM Woodstock conference (WOODSTOCK'97).* ACM, New York, NY, USA, Article 4, 3 pages. https://doi.org/10.475/123_4

1 INTRODUCTION

Negotiation between two agents is a game with a specific target, which tries to help them to reach a common-beneficial agreement. This report presents Agent 26 and its strategy. It also shows the motivation of this Agent. Then, the evaluation of its performance in the competition will be given. The challenges and originality of its strategy will be included in this part. Finally, suggestions and recommendations for future development will be proposed.

2 ANALYSIS AND DESIGN

This assignment requires to build a negotiating software agent to participate in Automated Negotiating Agent Competition 2018 (ANAC). Based on the guidelines given by task requirement [1], the main challenge is that agent has preference uncertainty. Namely, the agent only knows the part of its own information space. Therefore, this programming should not only build its model to the estimated cardinal utility function, it should also build a dynamic model to predict the utility of the opponents. Evenly, it can compute the Pareto optimal or a Nash solution to one bid relied on these two models.

To achieve this objective, the design for this programming is shown in figure 1. It consists by three components: Initialization, Strategy, Opponent Model.

2.1 Initialization

In this module, two functions are realized as design. the first function generates a series of parameters to make a decision in processing produce. The second one can build the preference model for Agent 26.

WOODSTOCK'97, July 1997, El Paso, Texas USA

ACM ISBN 123-4567-24-567/08/06.
https://doi.org/10.475/123_4

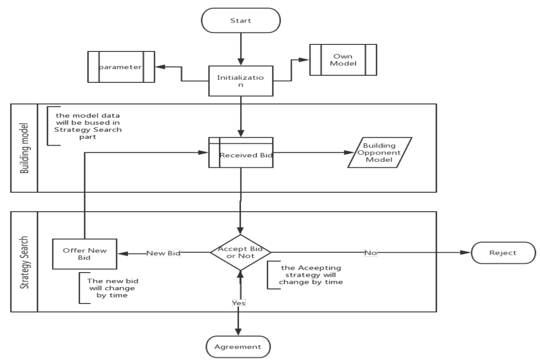

Figure 1: flowchart

2.1.1 *Parameter.*

The programming uses 4 parameters to control the concession: Maximum Bias, Last Received Offer (LRO), Last Offer (LO) and Time.

2.1.2 *Building Agent Model.*

As the preference of given agent deals is uncertainty, it should receive an ordered preference list partly. One method used in calculation of the utility is estimated cardinal utility function. The project named JohnnyBlack [3]suggests that frequency could be used to estimate the utility. However, the frequency cannot reflect the sequence in the reference list. furthermore, the bias instead of frequency to estimate utility. To calculate the bias, the first step is dividing the preference list into two parts and counting the frequency of one value appear in the issue list respectively. After recording the frequency, the second step is to calculate the original bias with the followed equation:

$$b_{vn} = f_{vnb} - f_{vnf}$$

Finally, these Bias will be regularized with followed equation:

$$Bias_{i1v1} = b_{i1v1} - Min(b_{i1vn}) + 1$$

The estimated utility function of a bid to the agent is:

$$U_1 = Bias_{i1v1} + Bias_{i2v1} + .. + Bias_{inv1}$$

In this programming, the sum of bias of value in the bid is regarded as the utility. One special situation in this design is that all bias in one issue is close to one another, which means that the change of value in this issue does not affect the utility.

2.2 Strategy

2.2.1 *Accepting Strategy.*

The situation agent accepts an offer when the sum of bias of received offer is greater than the acceptable bias. The acceptable bias of this agent is decided by the weighted parameter and the bias of last offer (LOB) or last received offer

Figure 2: minimum concession

Figure 3: The forth strategy

(LROB). The strategy module has different strategies to choose over time. Table 1 shows the equation of acceptable bias.

1. U_{max}
2. $U_{accept} = U_{max} * (l - t * 0.375)$
3. $U_{accept} = U_{max} * 0.7 - (t - 0.8) * (0.7 * U_{max} - U_o)/0.15$
4. $U_{accept} = Argmax(U_{hisortyofLOB})$
5. $U_{accept} > 0$

2.2.2 Bidding Strategy. The motivation for this strategy is try to search an acceptable bid for two agents in this computation. This acceptable bid should be close to Nash Solution, even it can have the largest social welfare.

Stage 1: The No-compromise Stage At the beginning of this negation, the bidding strategy is to offer the bid which has the highest bias in its own models. In other words, If the opponent accepts this bid, Agent 26 could reach the Pareto Frontier and has maximum utility in this negotiation competition. After 9 seconds, the strategy switches to the second stage.

Stage 2: The Minimum Concession Stage At this part,this negotiating party find the minimum concession of the LOB. Figure 2 shows the situation of phase2. The programming will calculate the difference between two value given from LOB and LROB in the same issue. After acquiring these differences, the issue has a a least difference will be picked. the new offer is the opponents value in this issue list. The new bid will be reconstructed by this new value and other old value from the LOB. Agent 26 will offer a new bid to opponents.

Stage 3: The Concession according to Opponent Preference

This stage is to reduce the lose from the LOB. Before the programming finds the max-value has the maximum bias in the LOB. The new bid is based on LROB but one value is instead of the max-value. As figure 4 shows, Agent 26 will produce this new bid which keeps the core utility of Agent 26.

Stage 4: The Consensus Stage

At this stage, the program provides the bid that has the maximum bias for Agent 26 from opponent history.

Stage 5: The Agreement Stage

The acceptable bias is adjusted to 0, which means this agent will receive nearly most bids from the opponent.

Figure 4: Agent 26 performance in three different size

2.3 Opponent Model

The opponent Model is built during all time after Agent 26 received the first offer. The way to build this model is using bias to decide the weighted of every value. Once bias are updated, the issues will be ordered. This method is similar with building Agent 26 model except it is dynamic.

3 RESULT

Figure5 shows the competition results of Agent 26 in three domains of different sizes, it said that this agent often lost when the domain is of a larger size. Moreover, its reliability in medium size is worse than that in the small size. Similarly, The utility of Agent 26 in the different domain also suggests that Agent 26 has worse performance in the larger size.

Figure6,7,8 shows the run time, the distance to Pareto Frontier and Nash equilibrium when Agent 26 against Agent 1, Agent 2, Agent 7 and Agent 13.

	Run time (s)	Dist. to Pareto	Dist. to Nash
0	140.5813	0.036767	0.47298166666666663
1	127.44193333333335	0	0.312368
2	160.31733333333335	0.025341666666666665	0.24625266666666668
3	164.2577	0.14627099999999998	0.37580299999999994

Figure 5: the run time, the distance to Pareto Frontier and Nash equilibrium in large size

	Run time (s)	Dist. to Pareto	Dist. to Nash
0	136.0331666666667	0.04324033333333333	0.28030733333333335
1	76.9141	0.0371566666666667	0.10058366666666667
2	109.86606666666667	0.056867	0.14998266666666668
3	145.35776666666666	0.10641633333333333	0.26541566666666666

Figure 6: the run time, the distance to Pareto Frontier and Nash equilibrium in medium size

	Run time (s)	Dist. to Pareto	Dist. to Nash
0	17.2406	0.0021133333333333334	0.046171333333333335
1	17.99846666666667	0.024632666666666667	0.04941
2	3.0101999999999998	0.034949	0.07531566666666667
3	17.8029	0.048682333333333334	0.06291533333333334

Figure 7: the run time, the distance to Pareto Frontier and Nash equilibrium in small size

4 EVALUATION

4.1 The Performance Compares Agent 26 Against Opponents in larger Preference

From the analysis part, Agent 26 has better utility at the beginning of negation competition. The run-time in the small domain is always close to 17s. In other words, the agreement is reached at phase 2. To the contrary, the rum time for the larger domain lies between 127.4s to 164s, which means the agreement is accepted in phase 4 to 5. it thought that Agent do not need enter phase 4 when there is 16s left.

The distance to Pareto varies from 0.106 to 0.03 in small domain. However, the value becomes to 0.14 in larger size domain. It this inferred that Agent 26 cannot distinguish which value is closer to the end of the bias list when the frequency of two value are closed. This influence of problem increase with the volume of domain size, which generate the reduction of the accuracy of eliminated function.

The distance to Nash is 0.5 to 0.3 in large domain. According to the research [2], the Nash equilibrium is related to the stability of this strategy. Namely, the result in the larger domain has a high probability to change in the next negotiation.

4.2 The Performance in Selfish Opponents

Agent 26 has a competitive disadvantage with the agent that just offers the max utility bid. If this agent always offers one specific bid, Agent 26 will accept this offer at the end of negotiation competition. As the utility of disagreement is 0 and the utility of receiving the offer is greater than 1. Therefore, Agent 26 accept this offer in the end. Although The reservation value in this assignment is changing with time, it cannot decrease the utility from an unchangeable agent.

4.3 Future Development

From the evaluation,Agent 26 suffers a greater loss at stage 5. It is essential to use an advanced approach to minimize the loss. It can slow down the concession in these two stages.

The second one is the way to build the model of opponents. The old design that adds a small coefficient to make one value appears in lower list have greater weights. However, the performance of this design is even worse than that of the current design. It referred that the test domain is so small that coefficient affects the reliability of the model. It is crucial to test out this coefficient in the larger domain as well.

The third one may arise when the opponent does not adhere to the built model, the model may not reflect reality well. According to Tim Bassrslag [4], it is important that test the robustness of the model. one possible solution to this problem is adding the feedback to test the difference between the predicted bid of opponent and the actual bid of opponent. the parameters of model should be adjusted based on this model.

5 CONCLUSIONS

This report provides an overview of the design and strategy of Agent 26. The result and evaluation could help to improve the performance of this programming. In this project, it is important to infer the preference of opponents and pull the agreement to Nash equilibrium and Pareto frontier. Actually, this programming is a practice to acquire a deeper understanding game theory and negotiation. It is thought that this competition is not a zero-sum game but a win-win game.

REFERENCES

[1] Enrico H. Gerding. 2018. COMP6203 : Intelligent Agents Coursework Specification. (2018).

[2] Peyman Faratin Alessio Lomuscio Simon Parsons Carles Sierra Michael Wooldridge Nicholas R. Jennings. 1998. Automated Negotiation : Prospects , Methods and Challenges. (1998).

[3] Jon Hoffman Sandip Sen Osman YucelEmail. 2017. Jonny Black : A Mediating Approach to Multilateral Negotiations. (2017), 16. https://doi.org/doi.org/10.1007/978-3-319-51563-2_18

[4] Mark Hendrikx Koen Hindriks Catholijn Jonker Tim Baarslag. 2012. AI 2012: Advances in Artificial Intelligence. 7691 (2012), 1–14.