

# Learning models of the negotiation partner in spatio-temporal collaboration

Yi Luo and Ladislau Bölöni

School of Electrical Engineering and Computer Science  
University of Central Florida, Orlando, Florida, USA.  
yiluo@mail.ucf.edu, lboloni@eecs.ucf.edu

**Abstract.** We describe an approach for learning the model of the opponent in spatio-temporal negotiation. We use the Children in the Rectangular Forest canonical problem as an example. The opponent model is represented by the physical characteristics of the agents: the current location and the destination. We assume that the agents do not disclose any of their information voluntarily; the learning needs to rely on the study of the offers exchanged during normal negotiation. Our approach is Bayesian learning, with the main contribution being four techniques through which the posterior probabilities are determined. The calculations rely on (a) feasibility of offers, (b) rationality of offers, (c) the assumption of decreasing utility, and (d) the assumption of accepting offer which is better than the next counter-offer.

## 1 Introduction

Spatio-temporal negotiation is a specific case of multi-issue negotiation where the issues under negotiation can be spatial or temporal values. In previous work [7, 8], we have shown that spatio-temporal negotiation has differentiating properties which require specific negotiation protocols and offer formation strategies.

In most practical negotiation problems, incomplete information is the default assumption. The self-interested negotiation partners disclose preferences only in the degree they believe that it allows them to reach a more favorable agreement. Naturally, a better knowledge of the opponent's preferences allows an agent to form better offers, and ultimately to reach a more favorable deal. Thus, in the recent years, a relatively lively research area deals with learning opponent preferences from the exchange of offers in the course of normal negotiation. In addition, argumentation techniques allow a more controlled way for agents to share a specific part of their preferences.

The preferences of the agent participating in spatio-temporal negotiation are defined in terms of physical properties such as current physical location, desired destination, current and maximum velocity, remaining fuel, desired trajectories and so on. This requires a different approach compared to worth oriented or task oriented domains.

In this paper, we outline a technique which allows an agent participating in a spatio-temporal negotiation to learn the preferences of the opponent agent. The

negotiation protocol we assume is a simple exchange of binding offers - that is, there are no arguments exchanged, the agent needs to infer the preferences of the opponent from its offers, or from the rejection of its own offers by the opponent. We are using the Children in the Rectangular Forest canonical problem as our working assumption, being a simplified environment, which however, represents all the properties of general spatio-temporal negotiations.

Our approach is based on Bayesian learning which was previously used for multi-agent negotiations by Zeng and Sycara [10], Li and Cao [6] and others. The agent updates its beliefs about the opponent’s preferences after each negotiation round.

The main contributions of this paper are the specific techniques which need to be used to calculate the posterior probabilities considering the spatial and temporal nature of the preferences, and the specific dependencies between the preferences. In addition, in contrast with most previous work in preference learning, we do not assume that the opponent uses a specific negotiation strategy.

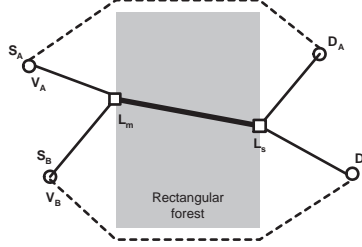
The only assumptions about the opponent are those dictated by common sense: (a) that it does not make binding offers which are not feasible for itself (b) that it does not make binding offers which are not rational for itself (they are worse than the conflict deal) (c) that from a pool of available offers it presents the ones with the higher utility for itself before the ones with the lower utility, and (d) that it doesn’t reject the offer which is better than the counter offer it plans to propose next round. Note that the third requirement does not necessarily imply a uniform concession. There is a very large space of possible strategies which verify these requirements. These four assumptions translate into three algorithms for the computation of the posterior probabilities in the Bayesian learning.

The remainder of this paper is organized as follows. We succinctly describe the CRF problem in Section 2. Then we introduce the theory of Bayesian learning in Section 3. We design three ways to determine the posterior probabilities of preference in learning agent. In Section 4, we design two strategies with different parameters for the opponent agent, and show the experimental study about the performance of learning. We talk about the related work in Section 5 and conclude in Section 6.

## 2 Justifying the CRF problem

Children in the Rectangular Forest (CRF) is a canonical problem designed to study spatio-temporal negotiations. It states that two children in the physical map go from their sources to destinations with their own speed. There is a rectangular forest in front of them. If the children join together, they can traverse the forest as a team with the speed of the slower child. Otherwise they have to go around the forest independently (see Figure 1). The key point for this problem is that the two children should negotiate and find a common path which potentially saves time compared to the case when they travel independently.

Many real life applications can be abstracted into the CRF problem, such as cooperative control of unmanned air vehicles (scouting and convoy) [9], multi-agent routing [11], RoboCup soccer (when and where the robot receives the ball, and when and where it passes the ball to teammates), and so on.



**Fig. 1.** The CRF problem: two children  $A$  and  $B$  try to move from sources  $S_A$  and  $S_B$  to destinations  $D_A$  and  $D_B$  with their own speed  $V_A$  and  $V_B$ . The dashed line indicates the trajectory of their conflict deals, and the solid line indicates the trajectory of an agreement.

In previous work [7, 8], we found that the optimal trajectories of the conflict deal and the collaboration deal should be a sequence of straight lines, and the meeting and splitting locations should be at the edges of the forest. So the offer between two agents contains at least four issues: the meeting location  $L_m$ , the meeting time  $t_m$ , the splitting location  $L_s$  and the splitting time  $t_s$ .

When an agent receives an offer from its opponent, it should check if the offer is feasible and rational for itself. The offer is not feasible if the agent can not get to the designated locations on time. The offer is not rational if it is worse than the conflict deal. To evaluate the offer, the cost function is defined as the time to arrive destination. Mathematically, for an offer  $\mathbf{O} = (L_m, t_m, L_s, t_s)$ , agent  $A$  (going from  $S_A$  to  $D_A$  with speed of  $v_A$ ) can arrive the destination at

$$C^{(A)}(\mathbf{O}) = \begin{cases} +\infty, & \text{if } \frac{\text{dist}(S_A, L_m)}{v_A} \geq t_m \\ +\infty, & \text{if } \frac{\text{dist}(L_m, L_s)}{v_A} \geq t_s - t_m \\ t_s + \frac{\text{dist}(L_s, D_A)}{v_A}, & \text{otherwise} \end{cases} \quad (1)$$

where the  $\text{dist}(S_A, L_m)$  means the spatial distance between source location  $S_A$  and meeting location  $L_m$ . The first two conditions in Equation 1 indicate the agent couldn't reach the meeting location and splitting location on time. So the cost of the offer will be infinity.

The cost of the conflict deal ( $C_{\text{conflict}}^{(A)}$  for the agent  $A$ ) is the time to arrive the destination if the agent doesn't negotiate. Obviously, such value is a criterion to decide whether the opponent's offer is rational or not. In addition, each agent has a best offer  $O_{\text{best}}^{(A)}$  whose trajectory is a straight line between source and destination, and the corresponding cost  $C_{\text{best}}^{(A)}$  is the ideal time it can arrive the destination. However, in most cases, the best offer for an agent may be neither

feasible nor rational for the opponent. In this paper, the utility of an offer is defined by the time the agent can save, divided by the time saved by the best offer.

$$U^A(O) = \frac{C_{conflict}^A - C^A(O)}{C_{conflict}^A - C_{best}^A} \quad (2)$$

There are three states for the utility of an offer in Equation 2: (a) the utility is minus infinity, which means the offer isn't feasible for the agent; (b) the utility is a negative number, which means the offer isn't rational for the agent; and (c) the utility is a positive number between zero and one, which means that the offer is a potential deal for both negotiators. Thus, the objective of the negotiation is to find a deal between two agents, which is feasible and rational, and its utility is maximized in agent point of view.

For a negotiation with incomplete knowledge, it is hard for an agent to find an offer which is also feasible and rational for the opponent, unless it knows the opponent's preference. Such preference includes the opponent's source, destination and its speed. In the next section, we will start to discuss how the learning agent guesses these information from a sequence of offers proposed by the opponent.

### 3 Bayesian learning of preferences

The nature of the offer discloses some velocity information between agents. Specifically, for an offer by the opponent, the agent can easily calculate the common speed that the opponent wants to use to traverse the forest. This speed can not exceed the maximum speed of opponent, because it will not propose an offer which is not feasible for itself. In this way, to guess the speed of opponent, the learning agent just needs to calculate the maximum common speed from all the previous offers it received from the opponent. Moreover, it should add some time buffers in the splitting time field (if necessary) when proposing the next counter offer to the opponent.

To guess the source location and destination of the opponent, the map is divided into grid. The combination of a grid in the source area and another one in destination area is called a location model. The learning agent tries to guess the location model of the opponent, by updating the probabilities (belief) of all these combinations. Initially, each location model has equal probability, and the sum of these probabilities equals to one. From time to time, these probabilities are updated along the number of offers the learning agent receives from the opponent.

#### 3.1 Bayesian learning

Bayesian learning is the classical method to update the belief based on evidences [6, 10]. Mathematically, the probability that the opponent is in the location model  $\{sx, sy, dx, dy\}$  (the coordinates of the grid cells), when receiving a new

evidence  $O_t$  (receiving an offer from opponent) can be calculated based on Bayes' theorem.

$$Pr(\{sx, sy, dx, dy\}|O_t) = \frac{Pr(O_t|\{sx, sy, dx, dy\})Pr(\{sx, sy, dx, dy\})}{\sum_{i,j,k,l=0}^{grid-1} Pr(O_t|\{i, j, k, l\})Pr(\{i, j, k, l\})} \quad (3)$$

where *grid* is the number of pieces the learning agent divides the map in each dimension, and  $t$  is the order of the offers it receives from the opponent. The formula shows that the posterior probability of a location model can be calculated by the prior probability times the probability to propose the offer given the opponent is indeed in the specific location model, and then normalized by all the updated probabilities. The learning algorithm is shown in algorithm 1.

---

**Algorithm 1** Algorithm for the learning agent

---

```

1: initialize all location models and assign them equal probability;
2: for  $t = 1$  to theEndOfNegotiation do
3:   get the opponent's offer  $O_t$ ;
4:   for all location models  $\{i, j, k, l\}$  do
5:     calculate  $Pr(O_t|\{i, j, k, l\})$ ;
6:     updated posterior probability  $Pr(\{i, j, k, l\}|O_t)$ ;
7:   end for
8:   normalize all the updated probabilities;
9:   propose next offer to opponent;
10: end for

```

---

### 3.2 Determining the posterior probabilities

In this subsection, we will discuss how the learning agent calculates  $Pr(O_t|\{i, j, k, l\})$  - the probability to propose the offer  $O_t$ , given that the opponent is in location model  $\{i, j, k, l\}$ . First, we establish four basic rules according to the assumptions of opponent agent. We let the learning agent eliminate non-rational location models which break these rules. Next, the learning agent will calculate the expected utility of opponent at a specific negotiation round, and increase the probabilities of the location models whose actual utilities of the offer are close to the expected one. At last, we introduce a half Gaussian approach to overcome the case where the learning agent doesn't know the expected utility for the opponent.

#### The four basic rules

We are going to make four basic assumptions about the behavior of the opponent agent in the negotiation. First, the opponent will not propose an offer which is not feasible for itself. Second, the opponent will not propose an offer which is not rational for itself, (otherwise, it will arrive the destination later than

its conflict deal). The third assumption is the opponent will propose a counter offer whose utility for itself is less or equal than the previous offers. This means that at each round of negotiation, the opponent should concede or at least insist on its last offer. The last assumption is that the opponent will accept the agent's offer if its utility is higher than the next counter offer. If the opponent in an assumed location model proposed an offer which breaks these rules, the learning agent will eliminate the possibility of that location model.

Practically, the value of  $Pr(O_t|\{i, j, k, l\}) = 0$  if the opponent was assumed at location model  $\{i, j, k, l\}$  but its last  $O_t$  breaks the four basic rules. All the other location models in the learning agent's belief share the same probability. Next, the learning agent will continue to discriminate these rational models and finds the one more likely.

### Updating belief based on expected utility

A self-interested agent will not only act rational, but also propose the most profitable offers at first, and concede to less profitable ones later. Using this idea, the learning agent can calculate the expected utility at a specific negotiation round, and assign more probabilities to those location models for which the utility of the offer is close to the expected one. In practice, the learning agent assumes that the opponent proposes offers with utilities starting from 1.0 at the first call and linearly decreasing during the negotiation.

$$EU(t) = 1 - \alpha \times t \quad (4)$$

where  $t$  is the order of the offers by the opponent and  $\alpha$  is the conceding speed. At each negotiation round, the location model whose utility of the offer  $O_t$  is close to  $EU(t)$ , will have its probabilities increased based on the Gaussian p.d.f.

$$Pr(O_t|\{i, j, k, l\}) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(U_t(O_t, \{i, j, k, l\}) - EU(t))^2}{2\sigma^2}} \quad (5)$$

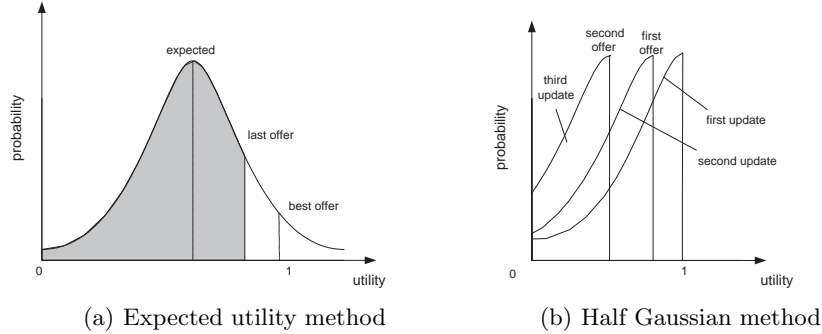
where  $U_t(O_t, \{i, j, k, l\})$  is the utility of the opponent's offer  $O_t$  when it is assumed in location model  $\{i, j, k, l\}$ , and  $\sigma$  is the coefficient of confidence. There are several approximations for this approach. The first one is we transfer a four-dimensional vector (offer  $\mathbf{O}_t$ ) into a value (utility  $U_t$ ) and assume they have the same posterior probabilities.

$$\begin{aligned} & Pr(O_t|\{i, j, k, l\}) \\ &= \frac{Pr(U_t|\{i, j, k, l\}) \times Pr(O_t|U_t, \{i, j, k, l\})}{Pr(U_t|O_t, \{i, j, k, l\})} \quad (\text{Bayes' theorem}) \\ &= Pr(U_t|\{i, j, k, l\}) \times Pr(O_t|U_t, \{i, j, k, l\}) \quad (\text{definition of utility}) \\ &= Pr(U_t|\{i, j, k, l\}) \quad (\text{assumption}) \end{aligned}$$

The equation assumes that  $Pr(O_t|U_t, \{i, j, k, l\}) = 1$ . In general, an agent may find many offers given a specific utility, and the assumption is not true for those strategies which want to try out every possible offer before conceding the utility. However, considering the negotiation time is crucial, we assume the opponent can only select one offer given a specific utility.

Another approximation for this approach comes from the four basic rules. The learning agent eliminates the probability of non-rational location models

whose utilities are negative or greater than the utility of the opponent’s last offer. Such elimination cuts off the Gaussian p.d.f (see Figure 2(a)), and the integral of the remaining part doesn’t equal one. The assumption here is we ignore these parts because all the probabilities will be normalized later, and we just need a discriminant value to judge the distance between the actual utility and the expected one. In the mean time, we can also change the variance of Gaussian p.d.f to reduce the impact of this approximation.



**Fig. 2.** Two methods to discriminate location models in the learning agent: 2(a): it updates belief based on Gaussian p.d.f which center at the expected utility and 2(b): it updates the probabilities based on half Gaussian p.d.f with the center at the utility of last offer.

The main deficiency of this approach is the difficulty to find a correct conceding speed to calculate the expected utility. If the opponent uses a different strategy which is not linear concession in utility, the learning agent may make a wrong guess. To overcome this problem, we need to model the opponent’s strategy and calculate the expected utility based on the probabilities of strategy models [5] (we leave it in the future work), or we can apply it in a save way which we will discuss next.

### Updating belief based on the half-Gaussian distribution

The idea of this approach is that an agent will concede step by step. At each step, it will give up a small amount of utility and see if the opponent accepts it. In this way, if the opponent which is assumed in a location model proposes two adjacent offers which have a big difference in utilities, the probability that the opponent is in that location model should be small.

Figure 2(b) depicts the way the learning agent calculates the conditional probability  $Pr(O_t|\{i, j, k, l\})$ . As we discussed above, the offer  $O_t$  is first transferred into utility  $U_t$ , given the assumption that the opponent is in location model  $\{i, j, k, l\}$ . Then, the learning agent calculates the probability of the offer based on utility and half Gaussian p.d.f, in which the mean of the Gaussian is at the utility of the last offer given the opponent is in the same location model.

## 4 Experimental study

### 4.1 Strategies used by the opponent

Before we study the performance of the learning, we introduce several simple strategies which the opponent might use in the CRF game. The first strategy is called Monotonic Concession in Space (MCS), which is parameterized by the conceding pace at each side of the forest ( $C_{meet}, C_{split}$ ). The MCS agent proposes its best offer at first, and then concedes in spatial fields to the opponent’s last offer. The meeting time field is tightly calculated based on the agent’s own speed, and the splitting time field is added some time buffer according to the maximum common speed in opponent’s previous offers. When the MCS agent doesn’t have space to concede its offer or the next concession breaks the rationality constraint, the agent quits the negotiation. On the other hand, if the next conceded offer is worse than the opponent’s last offer in utility, the MCS agent will agree the opponent’s offer (see Algorithm 2).

---

#### Algorithm 2 The MCS agent

---

```

1: the agent receives an offer  $O_t$  from the opponent;
2: calculates conceded offer  $O_{next}$  according to  $(C_{meet}, C_{split})$ ;
3: if not exist  $O_{next}$  then
4:   if  $O_t$  is rational and feasible then
5:     agree the opponent’s offer  $O_t$ ;
6:   else
7:     quit the negotiation;
8:   end if
9: else
10:  if  $U(O_t) \geq U(O_{next})$  then
11:    agree the opponent’s offer  $O_t$ ;
12:  else
13:    propose counter offer  $O_{next}$ ;
14:  end if
15: end if

```

---

The second strategy is called Uniform Concession (UC), which is parameterized by the conceding speed  $\lambda$ . The idea is the MCS strategy doesn’t test all the combinations of meeting and splitting locations across the forest, nor add any time buffer in the meeting time field. In this way, it may omit some potential deals. The UC agent, however, searches the offers in the whole spatio-temporal domain, and uniformly concedes in utilities of those offers. Specifically, the agent proposes an offer based on a range of utility. The length of the range is the conceding speed  $\lambda$ . The higher boundary of the range is initialized as one (the utility of the best offer), and it decreases with the amount of  $\lambda$  at the next time. To calculate the next counter offer, the UC agent searches all possible combinations whose utilities are in the current utility range, and selects the one which is most similar to the opponent’s last offer. The similarity between two offers is



defined by the sum of squared difference for each issues  $\|\mathbf{O}_{\text{next}} - \mathbf{O}_{\text{opponent}}\|^2$ . When the lower boundary of the utility range is less than zero, the agent quits the negotiation without agreement. When the utility of the opponent's offer is greater than the lower boundary of the utility range, the UC agent agrees the opponent's offer (see Algorithm 3).

---

**Algorithm 3** The UC agent
 

---

```

1: the agent receives an offer  $O_t$  from the opponent;
2: Create  $Set\langle offer \rangle$  to hold all possible offers;
3: while  $Set\langle offer \rangle$  is empty do
4:    $lower = lower - \lambda$ ;
5:   if  $lower \leq 0$  then
6:     quit the negotiation;
7:   end if
8:   find all  $Offer$  that  $U(Offer) \in (lower, lower + \lambda)$ ;
9:   add all  $Offer$  in  $Set\langle offer \rangle$ 
10: end while
11: find  $O_{next} \leftarrow \arg \min_{Set\langle offer \rangle} Similar(offer, O_t)$ ;
12: if  $U(O_t) \geq lower$  then
13:   agree the opponent's offer  $O_t$ ;
14: else
15:   propose  $O_{next}$ 
16: end if

```

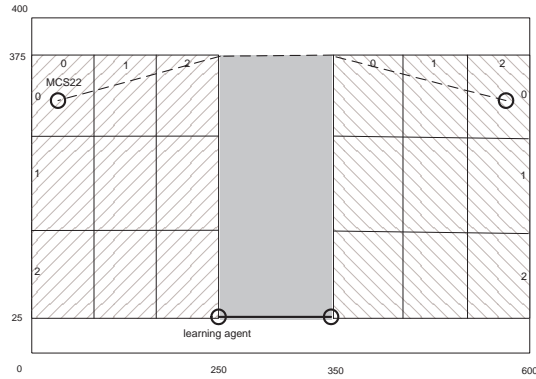
---

## 4.2 Performance of learning

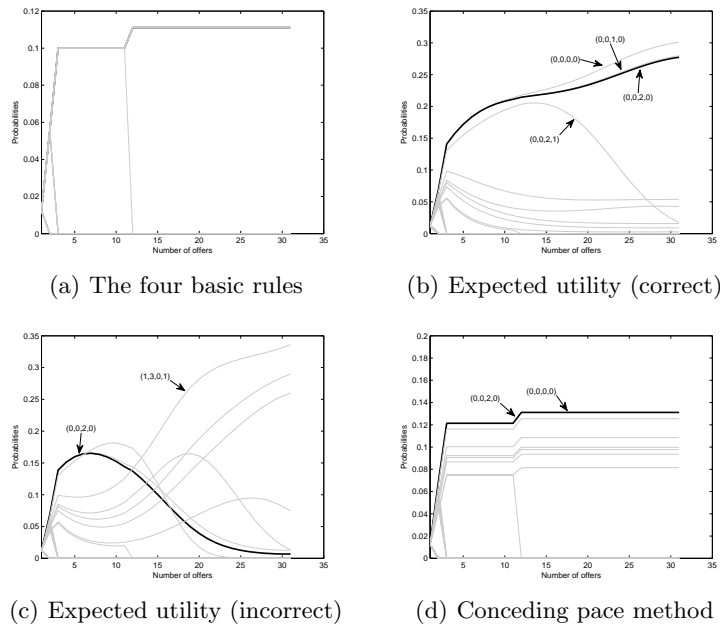
In this subsection, we focus on the accuracy of learning by comparing the opponent's actual location model with the probabilities of location models in the learning agent's belief. At first, we generate a typical scenario and see how the probabilities are updated during the negotiation. Then, we study the statistical performance in random generated scenarios.

### A typical scenario

Figure 3 shows a typical scenario, where the opponent is located at the centers of grids in a specific location model and the learning agent is located at the lower corners of the forest. We let the learning agent use different methods to update the posterior probabilities. The opponent uses MCS strategy with parameter of (2,2). Figure 4 shows the updating progress in the learning agent's belief. For all these three methods, 81 location models are initialized as equal probabilities at the beginning. When the learning agent uses four basic rules to update the posterior probabilities, some of location models are eliminated when the learning agent believes them non-rational. At the end of learning process, there are still 9 models it couldn't decide. So they share equal probabilities in learning agent's belief (see Figure 4(a)). Then, we assign a conceding speed for the learning



**Fig. 3.** A typical scenario: both the source and the destination area is divided into  $3 \times 3$  grids, which corresponds to 81 location models. The opponent agent is located at the center of grid (0,2) and wants to move to the center of grid (0,2) with the speed of 1.0. The learning agent is located at the lower-left corner of the forest, insists its best offer until the end of negotiation.



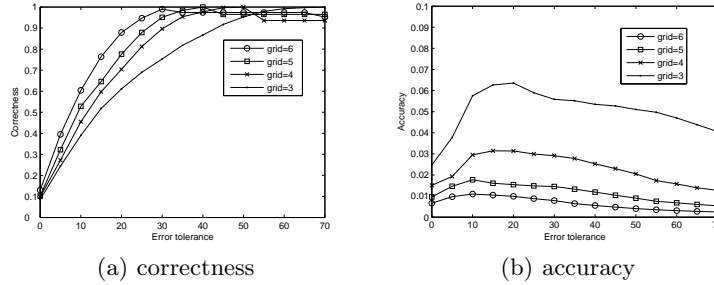
**Fig. 4.** The probabilities are updated along the number of offers from the opponent, the bold line is the opponent’s actual location model in the learning agent’s belief. The learning agent use: 4(a): four basic rules, 4(b): expected utility with correct conceding speed, 4(c): expected utility with incorrect conceding speed, 4(d): half Gaussian method, to determine the posterior probabilities.

agent to calculate expected utility by opponent, and let it negotiate with the same opponent. The 9 remaining location models are then further discriminated (see Figure 4(b)). However, the deficiency of using expected utility is disclosed when we assign an incorrect conceding speed in the learning agent’s assumption (see Figure 4(c)). At last, half Gaussian method gives a relatively compromised outcome, with the probability of the correct location model between the correct and incorrect Expected methods (see Figure 4(d)).

**A statistical study**

We enumerate all the other combinations where the opponent’s source and destination are initialized at the centers of grids. We let the opponent use simple strategies we discussed above and the learning agent use the four basic rules to update the belief. We calculate the averaged number of opponent models remained in the belief over all the combinations of grids and we increase the grid resolution in the map (see Table below):

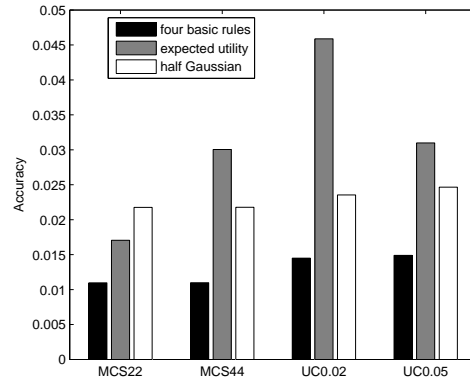
	MCS $C_{meet}=2, C_{split}=2$	UC $\alpha=0.02$
grid=3, 81(models)	5.271	4.099
grid=4, 256(models)	9.731	7.016
grid=5, 625(models)	17.733	11.9936



**Fig. 5.** The values of correctness and accuracy in the function of error tolerance in 1000 random generated scenarios.

The next question is how to decide whether the error tolerance due to the resolution of the grids is not high enough, and the opponent is not at the centers of the grids. Intuitively, the learning agent may eliminate the correct opponent model based on the four basic rules if the opponent is not at the center of the grids. This is a trade-off between the correctness and the accuracy of learning. The correctness of learning is defined by the number of experiments that the correct location model is still remained in the learning agent’s belief, over the total number of experiments. The accuracy of learning is the averaged probabilities of the correct model in all the experiments. If the error tolerance is too small, the correct location model may be eliminated, so its probability will be zero. On the

other hand, if it is too large, the number of location models remained in the belief is also large, so the probability of the correct location model will be very low too. In the experiment, we generate 1000 random scenarios, we let the learning agent use the four basic rules negotiating with a MCS agent. We calculate the correctness of these 1000 learning, and the accuracy of the learning. In addition, we change the value of error tolerance as well as the grid number to see the tendency of correctness and accuracy (see Figure 5). From the experiment, by increasing the error tolerance until the amount of time for the opponent travels a half length of grid diagonal, both correctness and accuracy are balancing. That's because if the opponent's source and destination are at the edge of grids, it is still believed as rational as it is located at the centers.



**Fig. 6.** The statistical study about the three learning approaches, when the opponent use the MCS22, MCS44, UC0.02, and UC0.05 strategies respectively. The results come from 1000 random generated scenario with learning agent's grid number of 6, error tolerance of 10. The learning agent which uses expected utility method assume the opponent's conceding speed is 0.03.

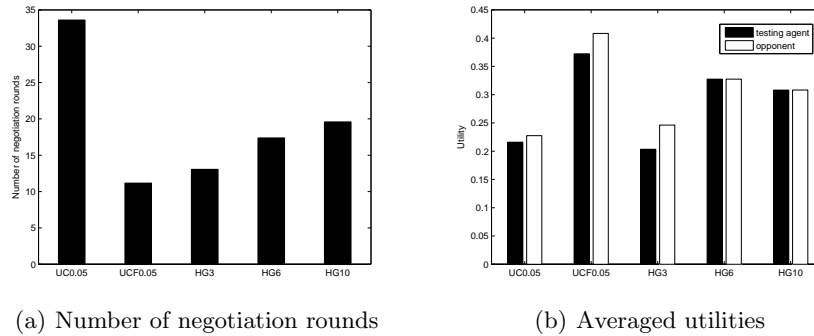
With a balanced error tolerance, we continue to study the performance of learning in 1000 random generated scenario when the learning agent use each of three methods to update the belief, and with opponent use simple strategies with different parameters (see Figure 6).

### 4.3 Performance of negotiation with or without learning

This subsection investigates how to apply the output of learning to accelerate the speed of negotiation. We design a strategy which is similar with the UC, but has full-knowledge about the opponent's preference (UCF). Contrasting to search offers within the small utility range, the UCF agent just has a conceding level. When proposing the next offer, it will choose the offer whose utility is higher than the current level and providing the opponent best utility. If there no

such offer exists, the level is decreased. If the opponent doesn't accept the offer, it will also decrease the level in the next round. When the level is less than zero, the agent quits the negotiation.

In the experiment, we generate 1000 random scenarios, letting the learning agent select its next counter offer in the same way as the UCF agent. The difference is it guesses the opponent as in the most probable location model in its current belief. If it couldn't find any offer satisfy the requirements, it doesn't decrease the level but continue to search by changing the assumption that the opponent in the second most probable location model. After a certain round of search (a certain amount of probabilities in belief have been searched), it decreases the utility level until find the next offer. The story behind this approach is that the learning agent initially doubts the correctness of its own belief before it concedes the utility level, which in other words, it updates the subjective things it can control before concede the objective things it can not control (such as the opponent's strategy, and the nature of scenario).



**Fig. 7.** The benefit of learning: UC0.05, UCF0.05, HG3, HG6 and HG9 negotiate with another UC0.05 in 1000 random scenarios

We compare the averaged number of negotiation rounds, and the average utility of the testing agent as well as the opponent gains, among the UCF agent (an agent with full knowledge), the UC agent (an agent without knowledge and without learning), and the learning agent (an agent which uses half Gaussian method, without knowledge but with learning), when each of them negotiates with another UC agent as a fixed opponent (see Figure 7). From the experiment, we can see: (a) the number of negotiation rounds is dramatically decreased if the agent is learning; (b) the utility of the deal is also improved if the agent learns; and (c) the effect of the first two observations are more obvious when the learning agent increase the grid resolution of the map.

## 5 Related work

Fatima et al. [4] used the shrinking pies model to investigate the multi-issue negotiation problem with deadlines. They assume a common preferences pool that both agents know in advance. During the negotiation, the agent assumes that the opponent uses the same strategy, and it guesses which type of preference in the common pool that the opponent is. In this paper, we also divide the preference into discrete representations of grids, and we remove the non-rational opponent model in the same way. However, we don't assume the learning agent knows the opponent's offering strategy, but we try to abstract the offer into the utility point of view to update the probabilities.

Hindriks et al. [2] used the Bayesian learning to study the opponent's type for each issue. They apply the probabilistic guess over a set of hypothesis and update the probabilities based on the distance between the opponent's expected bid and its actual bid. In this paper, we also try to guess the expected utility of opponent's offer, but we found that it is difficult to arbitrarily decide the conceding speed in the spatio-temporal negotiations. Moreover, issues of the offer in our problem are inter-dependent and the utility function is non-linear. Thus, we update the probability based on the expected utility and we design the half Gaussian method to compromise the risk of incorrect guess.

In addition, this paper also introduces some similar strategies which apply the learning result to improve the negotiation outcome, like the meta strategy introduced by Faratin et al. [3], the Bazaar model introduced by Zeng et al.[10], the learning strategy introduced by Bui et al. [1] and others.

## 6 Conclusion

In this paper, we applied the Bayesian learning in the spatio-temporal negotiation problem. The learning agent guesses the opponent's preference from the sequence of offers it received. We designed three approaches to update the probabilities of opponent's location models in learning agent's belief. First of all, for those non-rational models in which a rational opponent will not propose the offer, we eliminate their possibilities immediately. Then we continue to distinguish location models based on the expected utility for a specific negotiation time. At last, half Gaussian method is introduced to punish those models whose utilities of two adjacent offers have large difference. At the end of this paper, we evaluate these approaches and show the accuracy of learning by statistical analysis, then we show the benefit of learning when it negotiates with a fixed opponent in random scenarios.

Our future work is to continue the learning for the opponent's strategy models, or the belief about belief if the opponent is also learning. Combining the preference model with the strategy model will lead the negotiation to a decision-making problem, which gives the learning agent much more advantageous than its opponent. We will also apply the output of strategy models to help the agent

calculate the next expected offer (instead of calculating the expected utility in this paper). In this way, the preference models can be further updated.

## Acknowledgments

This work was partially funded by NSF Information and Intelligent Systems division under award 0712869.

This research was sponsored in part by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-06-2-0041. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

## References

1. H. H. Bui, D. Kieronska, and S. Venkatesh. Learning other agents preferences in multiagent negotiation. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-96)*, pages 114–119. AAAI Press, 1996.
2. K. H. Dmytro Tykhonov. Opponent modelling in automated multi-issue negotiation using bayesian learning. In *Seventh International Conference on Autonomous Agents and Multiagent Systems (AAMAS 08)*, pages 331–338, 2008.
3. P. Faratin, C. Sierra, and N. R. Jennings. Using similarity criteria to make issue trade-offs in automated negotiations. *Artificial Intelligence*, 142:205–237, 2002.
4. S. S. Fatima, M. Wooldridge, and N. R. Jennings. Multi-issue negotiation with deadlines. *Journal of Artificial Intelligence Research*, 27:381–417, 2006.
5. S. Ficici and A. Pfeffer. Simultaneously modeling humans’ preferences and their beliefs about others’ preferences. In *Seventh International Conference on Autonomous Agents and Multiagent Systems (AAMAS 08)*, pages 323–330, 2008.
6. J. Li and Y.-D. Cao. Bayesian learning in bilateral multi-issue negotiation and its application in mas-based electronic commerce. *iat*, 00:437–440, 2004.
7. Y. Luo and L. Bölöni. Children in the forest: towards a canonical problem of spatio-temporal collaboration. In *The Sixth Intl. Joint Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS 07)*, pages 986–993, 2007.
8. Y. Luo and L. Bölöni. Collaborative and competitive scenarios in spatio-temporal negotiation with agents of bounded rationality. In *Proceedings of the 1st International Workshop on Agent-based Complex Automated Negotiations , in conjunction with the The Seventh Intl. Joint Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS 08)*, pages 40–47, 2008.
9. R. W. B. Tim McLain. Unmanned air vehicle testbed for cooperative control experiments. In *American Control Conference, Boston, MA*, pages 5327–5331, 2004.
10. D. Zeng and K. Sycara. Bayesian learning in negotiation. *Int. J. Hum.-Comput. Stud.*, 48(1):125–141, 1998.

11. X. Zheng and S. Koenig. Reaction functions for task allocation to cooperative agents. In *Seventh International Conference on Autonomous Agents and Multiagent Systems (AAMAS 08)*, pages 559–566, 2008.