Bilateral Multi-Issue Negotiation Model for a Kind of Complex Environment

Jun Hu[1], Li Zou[1,2], and Ru Xu[1,3]
[1]College of Computer Science and Electronic Engineering, Hunan University, China
[2]The Second Hospital, University of South China, China
[3]Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, China

Abstract: *There are many uncertain factors in bilateral multi-issue negotiation in complex environments, such as unknown opponents and time constraints. The key of negotiation in complex environment is the negotiation strategy of Agent. We use Gaussian process regression and dynamic risk strategies to predict the opponent concessions, and according to the utility of the opponent's offer and the risk function, predict the concessions of opponent, then set the concessions rate of our Agent upon the opponent's concession strategy. We run the Agent in Generic Environment for Negotiation with Intelligent multi-purpose Usage Simulation (GENIUS) platform and analyze the results of experiments. Experimental results show that the application of dynamic risk strategy in negotiation model is superior to other risk strategies.*

Keywords: *Multi-issue negotiation; jaussian process regression; dynamic risk strategy; concession strategy.*

Received October 16, 2014; accepted September 9, 2015

1. Introduction

Negotiation is a process of reaching agreement on issues among parties. Automated negotiation based Agent is widely used [4, 8], and is the main means to do the complex interactive process instead of man. The automated negotiation in the complex environment is one of the challenge issues in the field of automated negotiation, which is like that, negotiation has multi-issues, big negotiation searching space, and strict constraints on time in the negotiation process [10], but also the preference of negotiation opponents and negotiation strategy are unknowable [5], and multi-negotiation groups process together. This paper focuses on trying to establish the bilateral multi-issues negotiation that can adapt the automated negotiation in complex environment, without knowing the opponent preferences and negotiation strategies, and also satisfy strict time constraints. This model builds a negotiation protocol fitting the bilateral multi-issue negotiation, based on the alternating offers protocol. As incomplete knowledge of opponents , instead of using machine learning to learn opponent negotiation strategies, this model adopts Gaussian process regression to predicate the possible best negotiation offered by opponents in the future time, that is in order to provide a reference for his own negotiation proposal. Basic negotiation factor such as limiting time of negotiation, is ignored by the former researchers, or only used to compute negotiation counts to represent negotiation time. However, that can not reflect the true negotiation states. This model brings in a dynamic risk strategy changed by negotiation time, for providing its own value of objective negotiation, and determining its own negotiation proposal finally. In order to evaluate the negotiation model performance, this paper adopts automated negotiation Agent championships platform Generic Environment for Negotiation with Intelligent multi-purpose Usage Simulation (GENIUS) [12], and in this platform, our Agent negotiates with other Agents which have different negotiation strategies to evaluate the performance of the proposed model in this paper. This article can be organized as follows. Some background information regarding the bilateral multi-issue negotiation in the complex environment is showed in section 2. Section 3 describes the bilateral multi-issue negotiation in the complex environment proposed in this paper. Section 4, elaborate the proposed strategy to deal with the complex environment of the negotiation in four aspects , including Gaussian process regression, dynamic risk mechanism, concession strategy of negotiation Agent, and the value of negotiation offer. In section 5, our Agent negotiates with other Agents in the negotiation tournament experiments, and then we verify the effectiveness and efficiency of the proposed negotiation strategy. Finally we make summarize in section 6.

2. Related Work

For the non-associated multi-issue, unknown opponent's preferences and time constraints in complex environment negotiation, scholars in related fields has been proposed many negotiation strategies. Early researchers constructed entirely rational and balanced negotiation strategy based on game theory [14], but that method requires Agent must have unlimited resources and time, which is not consistent with the real situation. As we all know, unknown information about the opponents and strategy is the challenging issues to build negotiation strategy. Then machine learning is one of widely approaches to predict the opponent's utility function and preference. Yu *et al.* [18] use implicit history negotiation information, with a least squares support vector regression machine estimate the opponent's utility function, and combine their opponent's utility function, then make up a constrained optimization problem. Using genetic algorithm optimization to solve this problem, the optimal solution is the counter-offer. This strategy exhibits higher performance in the absence of prior knowledge; however, it does not consider the time constraints, and the limited application range. Adar *et al*. [1] proposed a negotiation strategy based on meta-learning. This strategy is mainly through the initial rounds of negotiations to determine the type of opponent, then determine its own concession strategy by judging whether the opponent is easy to cooperation or not. This strategy is mainly to learn and use their opponent's negotiation strategy to determine our own concession strategy. Obviously, this method requires a lot of resources to learn the opponent's preferences and characteristics. Kawaguchi *et al*. [9] considered a negotiation strategy based on rival historical information to determine their own concession strategies and proposed utilities. The negotiation Agent model named AgentK based on that strategy, ranked first in ANAC 2011. Şerban *et al*. [15] designed the AgentFSEGA is a time constraint reasoning model. The model has not been made a greater utility concession until close to the bottom time. Through a longer time to fully understand the characteristics of the opponent, The Agent makes appropriate proposals and agreement with others before the bottom line is arriving. An and Lesser [2] designed a Agent model called Yushu, based on heuristics. Characteristic of the model is choosing the smallest concession strategy in the negotiation. The negotiation strategy has better negotiation results in the case of opponent providing lower utility. Last [11] designed AgentSmith based on estimating the opponent negotiation model, which send the offer that most likely to be accepted by the parties, through learning opponents preferences in the negotiation. The model can quickly reach an agreement, but Agent does not get high utility. Kawaguchi *et al*. [9] designed Nozomi negotiation model, at the beginning of negotiation, Agent gives offer with maximum utility , and then based on the difference utility value and remaining time between two negotiate participates to adjust the negotiation strategy. This model is able to keep track of the opponent's concession, but cannot predict the opponent utility function.

In order to effectively learn and estimate the opponent's decision function and preference profile, Hao and Leung [7] introduce the concept of non-exploitation point to adaptively adjust the degree that an agent exploits its negotiating opponent, which can be useful to make predictions on the preference profile of the negotiating partner. Fujita [6] proposed novel agent (AgentKF), which estimates the alternatives the opponent will offer based on the past negotiation sessions and could adjust the speed of compromising using the past negotiation sessions and find the pareto frontier. Williams [16, 17] proposed a negotiation strategy based on Gaussian regression analysis to predict the opponent's compromise, which mainly analyzes the proposal utility of opponents and decides our own concession strategy. This method considers the time constraint and the number of issues in complex environment, but does not consider the risk factors associated with the proposed utility of opponents, when setting rates concession. Negotiation model presented in this paper bases on the Colin's model, through adding dynamic risk, improves the efficiency of negotiation.

3. Negotiation Model in Complex Environment

3.1. Utility Function

The parties in negotiation give the offer *o*, which is expressed as $o= (v_1, v_2, \ldots\ldots, v_n)$. Wherein, v_i is the utility of issue i, w_i is the weight of issue i. The utility function of offer o is expressed as

$$U(o) = \sum_{i=1}^{n} w_i \cdot U_i(v_i) \tag{1}$$

Wherein, I is the set of issues. In the linear cumulative function, each issue is independent. In the negotiation, Agent knows its own utility function but don't know opponent's utility function. When receives the opponent's offer O_{opp}, the Agent can accept it or give a counter offer O_{own}. If the opponent accepts it, the negotiation is completed with agreement.

In the current study of negotiation in complex environment, we usually measured consumption in the negotiation process by negotiation rounds, which do not reflect the real time consuming. This paper proposes a model which use real time to represent the negotiation time. As negotiation rounds are uncertain within a time interval, this model reflects the true status of negotiations and the time characteristics of negotiation in complex environment. In model, we set the deadline as *tdl*. If the negotiation time exceeds the deadline, each Agent gets utility of zero. The utility of Agents reduces with the time elapsing. This model set a discount factor δ, and the discounting utility function is expressed as

$$D(u,t) = ue^{-\delta t/tdl} \tag{2}$$

In order to ensure the Agent in the negotiation can obtain a higher utility value, we introduce a dynamic risk function to change risk attitude of our Agent. The risk function is expressed as

$$R_{dynamic}(u) = u^{r(t)} \tag{3}$$

Wherein, $r(t)$ is the risk factor which changed by the time. This article will discuss how specific risk factor control Agent negotiation strategy in the next section.

3.2. Negotiation Algorithm

This paper uses alternating offer protocol, and the Algorithm1 gives the overview of our approach. The algorithm is divided into three steps to achieve. First, for each opponent's offer, our Agent records the time and the utility, and then predicts the concession rate by the function *gpr*(). Next, we can get the best offer time and the offer utility by *getbestUtility*() with predicted utility, variance, current time t_c and the risk factor $r(t)$. Finally, our Agent generates the offer by the function *generateoffer*(). If the opponent's value is higher than the utility value of our offer, it receives the offer and terminates negotiation; On the contrary, Agent gives the contra-offer. The details are described in the next section.

Algorithm 1: Bilateral negotiation algorithm

$Input: \delta, tdl$
$While\ t_c < tdl \quad do$
  $O_{opp} \Leftarrow receiveoffer()$
  $record(O_{opp}, tc)$
  $r \Leftarrow r(t_c)$
  $If\ \ regressionrequired(tc)\ \ then$
    $\mu, \sigma \Leftarrow gpr()$
    $tbest, Utbest \Leftarrow getbestUtility(\mu, \sigma, tc, \delta, r)$
  $end\ If$
  $Ut \Leftarrow getTARGET(tbest, Utbest, tc)$
  $If\ \ get\,Utility(O_{opp}, tc, \sigma) \geq Ut \quad then$
    $Accept(O_{opp})$
    $return$
  $end\ If$
  $O_{own} \Leftarrow generateoffer(Ut)$
  $proposeoffer(O_{own})$
$end\ While$

4. Negotiation Strategy

As described in algorithm 1, the negotiation strategy consists of three parts. First, by Gaussian process regression analysis to predict the opponent's concession; second, set our concession rate according to the predicted concession of opponent; third, our Agent gives the offer according to its own concession rate.

4.1. Gaussian Process Regression

According to known information, our Agent predicts the opponent concessions through Gaussian process regression [13] and the risk attitude. For each offer, we want to record relevant information, including the time and the utility value. The utility value is calculated by our utility function. We get the prediction and its confidence by Gaussian process regression. The reason why we chose Gaussian process regression to predict is that its nature is completely determined by the mean and covariance functions. We use a linear mean function and Matérn covariance functions to complete the Gaussian process regression analysis [11, 15]. Gaussian process output is a Gaussian distribution

$$f(u; \mu_t, \sigma_t) = \frac{1}{\sigma_t \sqrt{2\pi}} e^{\frac{-(u-\mu_t)^2}{2\sigma_t^2}} \tag{4}$$

According to the characteristics of the Gaussian distribution, the mean μ_t indicates the most likely value of *u* in time *t*, and the standard deviation σ_t represents the prediction confidence of μ_t .The information we need to know is deadline and discount factor. For every opponent offer O_{opp}, we firstly recorded time and the utility value of the offer, and then determine risk factor *r* based on the opponent utility values. We obtained the mean *u* and the variance σ by Gaussian process regression, in order to predict the opponent concession strategy. According to the predicted value previously obtained and the risk factor, we can get the best time to give the offer and the corresponding utility value.

In the Gaussian process regression, we select the maximum utility of the proposed utility value in a time window rival as input parameters. Using the time window mechanism can reduce the amount of data on the Gaussian process inputs. If you use all the observed data, the data is too large and it will extend the regression calculation process, thereby slowing down the process of consultation. The selection of the maximum opponent utility are rather than the average as input parameters, which because when we observed that the maximum utility of the opponent's offer, we expect to reach an agreement on this value.

4.2. Dynamic Risk Mechanism

Generally believed, the attitude of Agent towards risk can be divided into three categories: averse, seeking and neutral [2]. When the risk function is a power function expressed as $R(u)=u^r$, if $0 < r < 1$, we call it risk-averse; if r=1 , it is risk-neutral; if r>1, it is risk-seeking. According to this characteristic of risk function, we apply it to a real-time negotiation and associate with the utility of opponents to determine our risk strategy so as to achieve better negotiation results. The value of risk factor r determines the attitude of risk function. In order to associate the utility function and the negotiation time with the risk function to achieve dynamic risk mechanism, the risk factor is expressed as:

$$r(t_c) = \alpha - \beta(\frac{t_c}{tdl})^2 \quad (5)$$

Where, t_c is the current time, t_c/tdl is the ratio of the current time and time deadline, the value of t_c/tdl is between 0 and 1. When t_c/tdl is small, the negotiation has consumed less time, and the value of r will approach to α , the initial value of α determines the Agent's initial risk attitude. With the time consuming, the value of r becomes small, and the degree of changes determined by the value of β . At the beginning of negotiation, the model is risk-seeking. When close to the time deadline, Agent gradually moving risk-averse. Through this approach, according to the utility of opponent's offer and the negotiation time, this model achieves the dynamic risk mechanism.

4.3. Dynamic Concession Strategy

The model set concession strategy through predicting opponent's concession strategies and corresponding risk factor. According to μ_t and σ_t , we can calculate the best time $t_{best} = \arg\max_{t\in[t_c,tdl\ ]} E(t)$ and the corresponding utility u_{tbest} . Where, t_c is the current time, Agent's expected utility is $E(t) = \int_0^1 D(f(u;\mu_t,\sigma_t),t)du$. Where, $f(\cdot)$ is the possible distribution of Gaussian process output. We need to deal with the value of $f(\cdot)$, making it between 0 to1, so that fits the characteristic of Agent utility. After getting t_{best} , we can compute its utility $u_{best} = \arg\max_{u\in[0,1]} E_{offer}(u,t^*)$. To the $u \le u_{tbest}$ whole offers, We only accept the offer, which its. We use a cumulative distribution $F(u;\mu_t,\sigma_t)$ to expression the meeting condition set about u. Therefore, the except value of the offer is

$$E_{offer}(u,t_{best}) = D\left(F\left(u^{'},\mu_{t_{best}},\sigma_{t_{best}}\right),t_{best}\right) \quad (6)$$

Through Gaussian process regression, we can predict opponent's concession strategies and set our Agent's concession strategy. We introduce risk mechanism to the process of predicting opponent's concession. This paper accepts dynamic risk mechanism to set concession rate.

After using dynamic risk function, to those offers' utility is u, the expect value of those offer expressed as:

$$E_{offer}(u,t_{best}) = D\left(R_{dynamic}(u)F\left(u^{'},\mu_{t_{best}},\sigma_{t_{best}}\right),t_{best}\right) \quad (7)$$

In this way, a higher utility offer can reduce risk; on the contrary, the lower utility offer must bear a higher risk. With time consuming, Agent can dynamically change their own risk strategy, and improve the efficiency and performance of negotiation.

4.4. Selecting an Offer

Now we get the best time $t_{best} = \arg\max_{t\in[t_c,tdl\ ]} E(t)$ and the utility at this time u_{tbest} . We need to choose a utility value of offer at the current time t_c. We will not directly choose U_{tbest} as the offer utility, because it is too extreme by this way. We concede nonlinearly between (t_{lr} ,U_{lr}) and (t_{best} ,U_{tbest}), where is the time at which the regression was last performed and U_{lr} is the target utility at that time. The target utility U_t is given by

$$U_t = U_{lr} + (tc^2 - tlr^2)\frac{U_{tbest} - U_{lr}}{t_{best}{}^2 - t_{lr}{}^2} \quad (8)$$

The nonlinear selection lets this two points (t_{lr} ,U_{lr}) and (t_{best} ,U_{tbest}) in a quadratic function. Compared with the linear way, the advantage is that Agent can change the slope of varying U_t by t_c . When t_c is closer t_{best}, slope change sooner. When we get U_t, we need to ensure that the utility of U_{offer} as close as U_t . According to the result of test, we set $\Delta = 0$, if $|U_{offer} - Ut| \le \Delta$, represents U_{offer} within the desired range, otherwise we need to increase the value of Δ until the value U_{offer} satisfies the condition. Then our Agent generates an offer.

In summary, our Agent analyses the opponent concessions by Gaussian process regression in accordance with the opponent's utility, and then dynamically calculates concession rate by risk strategy of our Agent, finally calculates the utility value of our offer. Then the negotiation is finished

5. Experiments Design and Analysis

5.1. Experimental Settings

Experiment adopts GENIUS negotiation platform simulating complex negotiation environment. This platform not only can simulate multi-issues bilateral negotiation, but also can organize multiple Agents in the different negotiation domain to negotiate by alternate protocol, which called negotiation tournaments. GENIUS platform is able to obtain all the agents' negotiation utility. The negotiation results reflect to good or bad of the Agent, and can detect Agent negotiation performance.

GENIUS mainly includes two aspects, negotiation setting and negotiation simulator. We setup the negotiation setting, and make the Agents negotiate in the negotiation simulator. After that, we analyze the negotiation results and compare them. In this negotiation experiment, in consideration of the variation of t_c/tdl, the value of α is set as 4.5, and β as 4. At the beginning of negotiation, t_c is much smaller than tdl, and t_c/tdl also small, so the risk factor $r(t_c)$ approaches α. The value of risk factor is defined as 4.5, belonging type of strong risk preference. With the t_c/tdl increasing and closing to 1, the risk factor approach $\alpha-\beta$, this time Agent's risk factor approximate 0.5, so Agent belonging type of risking aversion.

5.2. Experimental Results and Analysis

5.2.1. RiskAgent Negotiate with IAMhaggler2011

We design RiskAgent which adopting dynamic risk mechanism to set concession rate. In the complex environment, compared with IAMhaggler 2011[17], which also using dynamic risk mechanism, RiskAgent both in the negotiation efficiency and negotiation utility, has the better performance. Because this Agent according to opponent's offer utility and negotiation time, it can change its own risk strategy. In order to verify RiskAgent performance, we need to get all agents' offer utility at every time, and the Utility value of the final agreement from the GENIUS. Before simulated negotiation, we set four GENIUS parameters, including negotiation protocol, negotiation participants, negotiation domain and terminal time. Each domain contains two negotiation participations. To compare the dynamic risk strategy, we select RiskAgent and IAMhaggler2011 as negotiation participants. Negotiation domain is England vs Zimbabwe，ITex vs Cypress and Camera_seller vs Camera_buryer. Negotiation protocol uses alternate proposed approach and the deadline is 180s. RiskAgent and IAMhaggler2011 will be as p_1 and p_2 in the three negotiation domains

The single negotiation session results in GENUUS negotiation environment are shown in Figure 1.

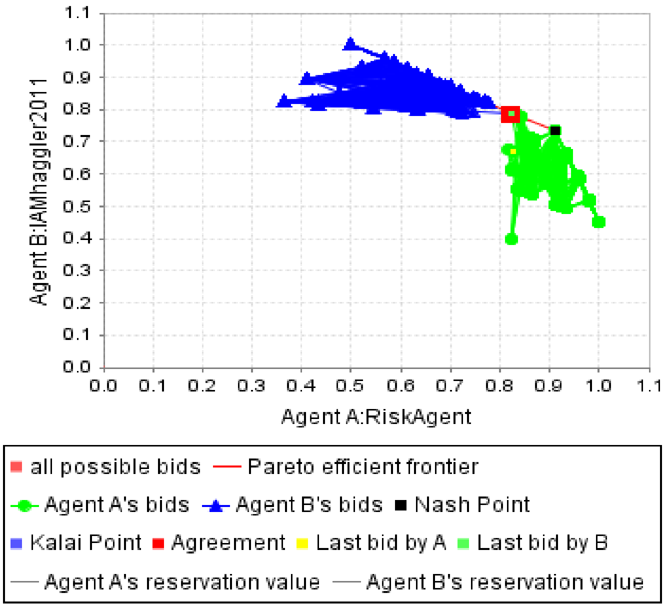

Figure 1. England vs zimbabwe domain negotiation results.

RiskAgent as p1 negotiates with IAMhaggler2011 as p2 in the England vs Zimbabwe domain, and negotiation result shows in Figure 1. In this figure, area covered by green circles and connect lines represent RiskAgent offer records, area covered by blue triangles and connect lines represent offer records of IAMhaggler2011, the red box represents the agreement point, red border is on behalf of the Pareto boundary [8] .When finished negotiation, IAMhaggler utility value is 0.77701379, RiskAgent utility value is 0.84355046, and negotiation time is 0.848245646, which is the ratio of the end of negotiation time and time deadline（t/tdl）.

Similarly, RiskAgent and IAMhaggler2011 need to swap roles and negotiate with each other, and then negotiation of the two Agents in the domain is completed. RiskAgent and IAMhaggler2011 negotiate by this way in three domains. Experiment results are shown in Table 1.

Table 1. Negotiation results of risk agent vs IAmhaggler2011.

RskAgent	IAMhaggler2011	U(RiskAgent)	U(IAMhaggler)	Time(t/tdl)
England	Zimbabwe	0.843550457	0.777013792	0.84824564
Zimbabwe	England	0.830365308	0.754920567	0.88483381
Cypress	Itex	0.820789866	0.480437882	0.97512645
Itex	Cypress	0.808193359	0.361915324	0.98030355
Camera_seller	Camera_buyer	0.880800000	0.815413165	0.59570302
Camera_buryer	Camera_seller	0.803413165	0.825085714	0.31390785

According to results, compare the utility obtained from the consultation on RiskAgent and IAMhaggler2011 in different negotiation domains, results are showed in Figure 2.

From Table 1 and Figure 2, we can obviously see that, in the two negotiation domains, England vs Zimbabwe and ITex vs Cypress, RiskAgent is either as p_1 or as p_2, received higher utility than IAMhaggler2011. After swapping their role, although the utility of RiskAgent is lower than IAMhaggler2011, when they both as p_1, RiskAgent gets higher utility than IAMhaggler2011 in Figure 2. From comparison experiments, we can see that the RiskAgent using dynamic risk strategy has obvious advantages, and is better than the IAMhaggler2011, which not uses dynamic risk strategy.

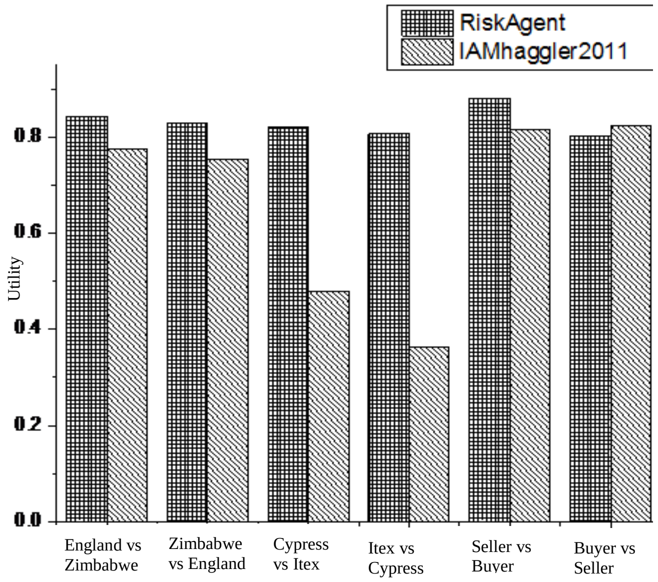

Figure 2. Utility of negotiation opponents.

5.2.2. Negotiation Tournament

In GENIUS platform, you can simulate the bilateral multi-issue negotiation as well as organize Agents to take turn to bilateral multi-issue negotiation. GENIUS platform is able to draw each utility values from negotiation, and the results of the negotiation can be integrated to reflect the merits of the Agent. We chose three negotiation domains England (Agent A) VS Zimbabwe (Agent B) , ITex (Agent A) VS Cypress (Agent B) and Camera(Seller vs Buyer) to make experiment analysis. We chose seven Agents negotiate in the tournament, including AgentFSEGA [15] AgentK [9], Smith [11], Nozomi Yushu [2], IAMhaggler [16] and RiskAgent.

In the tournament, each domain $D \in \Phi$ contains two parties P_1^D , P_2^D . Each Agent $A \in \Lambda$ need to be as P_1^D and P_2^D to negotiate. Each Agent's tournament final score [3] is calculated by the following formula:

$$s(A) = \frac{1}{|\Phi|}\sum_{D\in\Phi}\left(\frac{1}{2}\tilde{u}_D(A,P_1)+\frac{1}{2}\tilde{u}_D(A,P_2)\right) \tag{9}$$

Wherein, $\tilde{u}_D(A,P)$ is expressed as,

$$\tilde{u}_D(A,P) = \frac{\bar{u}_D(A,P) - \min_{B\in\Lambda\setminus\{A\}} \bar{u}_D(B,P)}{\max_{B\in\Lambda\setminus\{A\}} \bar{u}_D(B,P) - \min_{B\in\Lambda\setminus\{A\}} \bar{u}_D(B,P)} \tag{10}$$

Wherein, $\bar{u}_D(A,P)$ contains two parts, $\bar{u}_D(A,P_1)$, $\bar{u}_D(A,P_2)$, which is the average utility of the Agent to be as P_1^D P_2^D. Expressed as

$$\bar{u}_D(A,P_1) = \frac{1}{n-1}\sum_{B\in\Lambda\setminus\{A\}} U_{A\to B}^D \tag{11}$$

$$\bar{u}_D(A,P_2) = \frac{1}{n-1}\sum_{B\in\Lambda\setminus\{A\}} U_{B\to A}^D \tag{12}$$

Where, $U_{A\to B}^D$ is the utility in Domain D when Agent A is P_1^D.

In addition, we also need to calculate the average negotiation efficiency of each Agent, $Eff_D(a)$ is expressed as,

$$Eff_D(\mathrm{a}) = U_D(a)/T_D(\mathrm{a}) \tag{13}$$

Wherein, $T_D(a)$ is the average negotiation time, expressed as

$$T_D(\mathrm{a}) = \frac{\sum_{a'\in A\cdot a'\neq a} t_D(a,a')}{|A|-1} \tag{14}$$

Where, $t_D(a,a')$ is the negotiation time in Domain D when Agent a negotiate with Agent a', which is expressed as

$$t_D(a,a') = \frac{t_{agreement}(a,a')}{tdl} \tag{15}$$

Where, $t_{agreement}(a,a')$ is the agreement time, tdl is time deadline.

To evaluate the performance of our negotiation strategy, we recorded the average utility, the average negotiation time, the average negotiation efficiency and the final score of the tournament. Our experiments are repeated 100 times and the results are averaged and the 95% confidence intervals are provided to indicate the statistical significance of the results. (Sample statistic ± margin of error).

In Domain England vs Zimbabwe, the negotiation results of seven Agents are shown in Table 2.

Table 2. Negotiation results of domain england vs zimbabwe.

Agent	Utility	Time	Efficiency
AgentFSEGA	0.7495±0.0053	0.4043±0.0075	1.8534±0.0015
AgentK	0.8224±0.0037	0.7621±0.0054	1.0802±0.0075
AgentSmith	0.6183±0.0028	0.3461±0.0098	1.7875±0.0086
Nozomi	0.8076±0.0042	0.6275±0.0014	1.2877±0.0088
Yushu	0.7573±0.0035	0.7194±0.0017	1.0524±0.0088
IAMhaggler2011	0.7862±0.0081	0.5602±0.0032	1.4047±0.0046
RiskAgent	0.8246±0.009	0.6301±0.0066	1.3088±0.0095

As can be seen from the table, in this domain, the utility of RiskAgent is 0.8246±0.009, ranked first, the utility of AgentK is 0.8224±0.0037, ranked second, and the utility Nozomi is 0.8076±0.0042, ranked third. In the negotiation domain of England vs Zimbabwe, the average efficiency of negotiation of AgentFSEGA is 1.8534±0.0015 and it ranks the top. But its average utility is 0.7495±0.0053. Moreover, RiskAgent has the highest negotiation efficiency among the methods whose average utility is in the top three. Therefore, RiskAgent can achieve higher average utility and negotiation efficiency in the negotiation domain of England vs Zimbabwe.

In ITex VS Cypress domain, the negotiation results of sevenAgents are shown in Table 3.

Table 3. Negotiation results of ITex VS cypress domain.

Agent	Utility	Time	Efficiency
AgentFSEGA	0.6193± 0.0063	0.7820± 0.0015	0.7915± 0.0046
AgentK	0.6751± 0.0059	0.8206± 0.0034	0.8228± 0.0015
AgentSmith	0.3611± 0.0013	0.6118± 0.0025	0.5902± 0.0088
Nozomi	0.7198± 0.0042	0.8165± 0.0075	0.8812± 0.0012
Yushu	0.7533± 0.0065	0.9083± 0.0054	0.8289± 0.0034
IAMhaggler2011	0.7157± 0.0088	0.8212± 0.0071	0.8709± 0.0076
RiskAgent	0.7642± 0.0015	0.9166± 0.0004	0.8347± 0.0031

From this table, we can see that in this domain, the utility of RiskAgent is 0.7642±0.0015, ranked first, the utility of Yushu is 0.7533±0.0065, ranked second, and the utility Nozomi is 0.7198±0.0042, ranked third. In the negotiation domain of Itex vs Cypress, the average efficiency of negotiation of RiskAgent is 0.8347±0.0031 and it ranks the third. Nozomi has the highest negotiation efficiency (0.8812±0.0012), but its average utility is lower than that of RiskAgent. Hence, RiskAgent can also achieve higher average utility and negotiation efficiency in the negotiation domain of Itex vs Cypress.

In the Camera domain, the negotiation results of seven Agents are shown in Table 4.

Table 4. Negotiation results of the camera domain.

Agent	Utility	Time	Efficiency
AgentFSEGA	0.7642± 0.0065	0.3756± 0.0034	2.0341± 0.0087
AgentK	0.8363± 0.0024	0.6918± 0.0049	1.2353± 0.0064
AgentSmith	0.6629± 0.0083	0.4172± 0.0006	1.5897± 0.0042
Nozomi	0.8139± 0.0029	0.5127± 0.0075	1.5883± 0.0063
Yushu	0.8817± 0.0094	0.7954± 0.0096	1.1084± 0.0045
IAMhaggler2011	0.8061± 0.0078	0.4214± 0.0023	1.9165± 0.0089
RiskAgent	0.8665± 0.0008	0.5125± 0.0072	1.6918± 0.0079

From Table 4, in the Camera domain, the average utility of RiskAgent is 0.8665± 0.0008, ranked second. The average utility of Yushu is0.8817± 0.0094 ranked first. From the view of average negotiation efficiency RiskAgent is 1.6918± 0.0079, still ranked third, AgentFSEGA is 2.0341± 0.0087, ranked first. IAMhaggler2011 is 1.9165± 0.0089, ranked second. The average efficiency of Yushu is 1.1084± 0.0045. Fully visible in the Camera domain, RiskAgent can get high average utility. In the first three Agents on average utility, the average negotiation efficiency of RiskAgent ranked the first.

Based on the above results of the tournament, the comparison of the utility in three domains is shown in Figure 3.

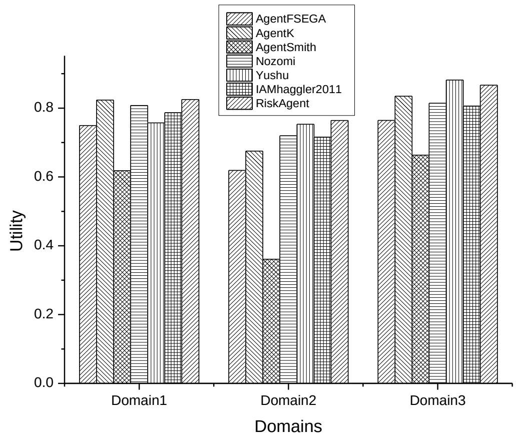

Figure 3. Comparison of the utility in three domains.

As can be seen from Figure 3, in three domains, RiskAgent ranked first in two domains and ranked second in the third domain, which fully proved dynamic risk mechanism has better performance.

Finally, according to the calculation rules in GENIUS tournament, all Agents ranked by formula to calculate the score, as shown in Table 5.

Table 5. Tournament results of seven agents.

Agent	AgentFSEGA	AgentK	AgentSmith	Nozomi	Yushu	IAM haggler	Risk Agent
Score	0.557± 0.09	0.823± 0.08	-0.942± 0.07	0.763± 0.02	0.911 ± 0.05	0.713± 0.06	0.928 ± 0.09
Rank	6	3	7	4	2	5	1

By calculating tournament score, RiskAgent 0.928± 0.09 ranked first, Yushu0.911± 0.05 ranked second, AgentK 0.823± 0.08 ranked third. We can see, the performance of RiskAgent is over all participating Agents.

Comparing the tournament results of seven different agents in three negotiation domains, the scores of RiskAgent are always higher than IAMhaggler2011. It proves that the Agent using Dynamic risk strategy has a higher average utility than the Agent using fixed risk value. In general, the average utility of RiskAgent ranks the top in domain of Itex vs Cypress among seven Agents and Itex vs Cypress. Although it ranks the second in domain of Camera, it achieves high negotiation utility. From the view of negotiation efficiency，RishAgent is the third in the three negotiation domains. In summary, RishAgent get good utility without reducing the negotiation efficiency. Meanwhile, the performance of RishAgent integrated ranked first based on the count rules of GENIUS negotiation tournament. Therefore, RishAgent has the best negotiation performance among the seven Agents.

Therefore, the dynamic risk strategy can associate the utility value with the negotiation time to dynamically change the risk factors, which have obvious advantages to negotiate in complex environments in real time. First, RiskAgent inherited the Agent designed by Colin etc., which predict the opponent concessions by Gaussian process regression. In addition, RiskAgent introduce the dynamic risk strategy, which not only consider the opponent's utility of offer and the impact of the concession rate on our Agent's concession rate, but also takes into account the issue of time elapsing, which is crucial in the negotiation in real-time complex negotiation environment. As negotiated in complex environment without considering the number of negotiation interaction rounds, the negotiation efficiency depends directly on the time elapsing. Agent generate offer according to the dynamic risk strategy based on elapsing time and the predicting of opponent concessions. The performance of RiskAgent has greatly improving compared to IAMhaggler2011, and significantly better than the AgentFSEGA based time constraints. Comparing with Yushu which choose the smallest concession strategy, the performance of RiskAgent is also better. What's more, the results show that, RiskAgent using dynamic risk strategies get the best performance in all the Agents.

6. Conclusions

In this paper, we proposed a negotiation strategy which introduces the negotiation dynamic risk mechanism based elapsing time in the negotiation model proposed by Colin et al which is based on Gaussian regression to predict opponent's concessions. Because the risk factors is associated with the utility of the opponents and the negotiation time, in the whole negotiation process, Agent needs to determine the concession rates and concessions strategy according to the opponent, while analyzing of the opponent concession strategy. Finally, we analysis the comparative empirical results, and verify the efficiency of the negotiation strategy is higher and more stable.

Acknowledgement

This work was supported by Guangxi Key Laboratory of Trusted Software (No. kx201537).

References

[1] Adar M., Sofy N., and Elimelech A., "Gahboninho: Strategy for Balancing Pressure and Compromise in Automated Negotiation," *Complex Automated Negotiations: Theories, Models, and Software Competitions Studies in Computational Intelligence*, pp. 205-208, 2013.

[2] An B. and Lesser V., "Yushu: A Heuristic-Based Agent for Automated Negotiating Competition," *New Trends in Agent-Based Complex Automated Negotiations Studies in Computational Intelligence*, pp. 145-149, 2012.

[3] Baarslag T., Hindriks K., Jonker C., Kraus S., and Lin R., The First Automated Negotiating Agents Competition, *New Trends in Agent-Based Complex Automated Negotiations*, pp.113-135, 2012.

[4] El-Ashmawi W., Jun H., and Renfa L., "Stability Coalition Formation with Cost Sharing in Multi-Agent Systems Based on Volume Discount," *The International Arab Journal of Information Technology*, vol. 12, no. 3, pp. 296-303, 2015.

[5] Fatima S., Wooldridge M., and Jennings N., "Optimal Negotiation Strategies for Agents with Incomplete Information," *in Proceedings of International Conference on Autonomous Agents and Multiagent Systems*, Bologna, pp. 377-392, 2002.

[6] Fujita K., "AgentKF: Automated Negotiating Agent with Strategy Adaptation," *Next Frontier in Agent-Based Complex Automated Negotiation*, pp. 137-142, 2015.

[7] Hao J. and Leung H., "CUHKAgent: An Adaptive Negotiation Strategy for Bilateral Negotiations over Multiple Items," *Novel Insights in Agent-based Complex Automated Negotiation, Studies in Computational Intelligence*, pp. 171-178, 2014.

[8] Jennings N., Faratin P., Lomuscio A., Wooldridge M., Parson S., and Sierra C., "Automated Negotiation: Prospects, Methods and Challenges," *International Journal of Group Decision and Negotiation*, vol. 10, no. 2, pp. 199-215, 2001.

[9] Kawaguchi S., Fujita K., and Ito T., "AgentK: Compromising Strategy Based on Estimated Maximum Utility for Automated Negotiating Agents," *New Trends in Agent-Based Complex Automated Negotiations Studies in Computational Intelligence*, pp. 137-144, 2012.

[10] Kraus S., Wilkenfeld J., and Zlotkin G., "Multiagent Negotiation under Time Constraints," *Artificial Intelligence*, vol. 75, no. 2, pp. 297-345, 1995.

[11] Last N., "Agent Smith: Opponent Model Estimation in Bilateral Multi-Issue Negotiation," *New Trends in Agent-Based Complex Automated Negotiations Studies in Computational Intelligence*, pp. 167-174, 2012.

[12] Lin R., Kraus S., Baarslag T., Tykhonov D., Hindriks K ., and Jonker C., "Genius: An Integrated Environment for Supporting the Design of Generic Automated Negotiators," *Computational Intelligence*, vol. 30, no. 1, pp. 48-70, 2014.

[13] Rasmussen C. and Williams C., *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[14] Rubinstein A., "Perfect Equilibrium in a Bargaining Model," *Econometrica*, vol. 50, no. 1, pp. 97-109, 1982.

[15] Şerban L., Silaghi G., and Litan, C.,

"AgentFSEGA: Time Constrained Reasoning Model for Bilateral Multi-Issue Negotiations," *New Trends in Agent-Based Complex Automated Negotiations Studies in Computational Intelligence*, pp. 159-165, 2012.

[16] Williams C., "Practical Strategies for Agent-Based Negotiation in Complex Environments," Thesis, University of Southampton, 2012.

[17] Williams C., Robu V., Gerding E., and Jennings N., "Using Gaussian Processes to Optimise Concession in Complex Negotiations Against Unknown Opponents," *in Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, Barcelona, pp. 432-438, 2011.

[18] Yu C., Ji G., Hua-Mao G., and Zhao-Yang F., "Automated Negotiation Decision Model Based on Machine Learning," *Journal of Software*, vol. 20, no. 8, pp. 2160-2169, 2009.

Jun Hu born in 1971 and received M.Sc. in Computer Application from Kunming University of Science and Technology, Kunming, China, and Ph.D. in Computer Science and Technology from Zhejiang University, Hangzhou, China. In 2010, he was an academic visitor at University of Southampton working on multi-agent system. Currently, he is an associate professor of Hunan University, Changsha, China. Senior member of China Computer Federation (CCF). His research interests include multi-agent system, distributed artificial intelligence and software engineering.

Li Zou, born in 1988. He gets the M.A's degrees in computer science and technology from Hunan University, Changsha, China, 2014. His main research interests include automated negotiation and multi agent system.

Ru Xu, born in 1990. She gets the M.A's degrees in computer technology from Hunan University, Changsha, China, 2015. Her main research interests include artificial intelligence and multi-agent system.