Decision Support Systems 73 (2015) 1–14

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

Automated negotiation for e-commerce decision making: A goal deliberated agent architecture for multi-strategy selection

Mukun Cao [a,*,1], Xudong Luo [b,*,2], Xin (Robert) Luo [c], Xiaopei Dai [a]

[a] School of Management, Xiamen University, Siming South Road 422, Xiamen 361005, China
[b] Institute of Logic and Cognition, Sun Yat-sen University, 135 Xingang West Road, Guangzhou 510275, China
[c] Anderson School of Management, The University of New Mexico, 1924 Las Lomas NE, MSC05 3090, Albuquerque, NM 87131, United States

ARTICLE INFO

Article history:
Received 12 August 2014
Received in revised form 7 January 2015
Accepted 16 February 2015
Available online 3 March 2015

Keywords:
Automated negotiation
Negotiating agent
Agent architecture
Negotiation strategy selection
Belief–desire–intention model
Goal deliberation

ABSTRACT

Automated negotiation plays an important role in dynamic trading in e-commerce. Its research largely focuses on negotiation protocol and strategy design. There is a paucity of further scientific investigation and a pressing need on the implementation of multi-strategy selection, which is crucially useful in human–computer negotiation to achieve better online negotiation outcomes. The lack of such studies has decelerated the process of applying automated negotiation to real world problems. To address the critical issue, this paper develops a multi-strategy negotiating agent system. More specifically, we formally define the agent's conceptual model, and design its abstract software architecture. Grounded on the integration of the time-dependent and behavior-dependent tactics, we also develop a multi-strategy selection theoretical model and algorithm. To demonstrate the effectiveness of this model algorithm, we implement a prototype and conduct numerous experiments. The experimental analysis not only confirms our model's effectiveness but also reveals some insights into future work about human–computer negotiation systems, which will be widely used in the future B2C e-commerce.

1. Introduction

Negotiation is a communication process among a group of parties with conflicting interests or preferences in order to reach an agreement or compromise [1,2]. The tremendous success of the online auction [3], which is a kind of one-to-many negotiation and has been employed as the main trading mechanism in the electronic market and smart market [4], suggests that the dynamic trading based on e-negotiation has gradually become the primary paradigm of decision making in e-commerce [5–8]. In addition, e-commerce oriented negotiation is increasingly assuming a pivotal role in many organizations, and a number of prominent negotiation models have been developed over the past decades [9].

There are three forms of e-commerce oriented negotiation [2]: human–human negotiation, computer–computer negotiation, and human–computer negotiation. With the rapid growth of global e-markets, there has been a significant interest in designing Automated Negotiation System (ANS) [10] that can serve as surrogates for human business decision-makers, where software agents are designed to autonomously act on behalf of the real-world parties [11,12]. As the automated negotiation is becoming crucially important and pervasive and agents promise exciting opportunities to turn conventional transactions into an automated, cost-efficient manner, the study of ANS has piqued increasing interest in the scholarly fields of e-commerce and artificial intelligence [13].

While the e-commerce and AI literatures mirror that the ANS can be used in computer–computer and human–computer negotiations, extant studies on ANS primarily focus on the former, leaving the latter comparatively unexplored [14]. In fact, human involvement in decision-making is still required in most of present online negotiations, and with the ever mushrooming growth of e-commerce and e-markets, there is an increasing potential for the use of software agents to more effectively and efficiently negotiate with human negotiators [11,15]. The human–computer negotiation plays a paramount role in the e-commerce oriented applications, especially in the B2C context where software agents act as business provider [16]. Compared with the traditional online sales mode where customers view the basic product or service information on the website and often need to negotiate with human salespeople through a "contact us" link, a human–computer ANS can help business organizations to reduce the labor cost for negotiation and greatly increase the transaction efficiency to the optimum extent.

Prior studies have been conducted to design various human–computer negotiating agent [14,15], which demonstrate that a software agent can proficiently negotiate with and even outperform people. Owing to the randomness of the human's behavior, the human–computer negotiation context is assumedly more complicated.

* Corresponding authors.
E-mail addresses: caomukun@xmu.edu.cn (M. Cao), luoxd3@mail.sysu.edu.cn (X. Luo), xinluo@unm.edu (X.(R.) Luo), daixiaopei0701@126.com (X. Dai).
[1] Tel.: +86 592 2181382.
[2] Tel.: +86 20 84113302.

http://dx.doi.org/10.1016/j.dss.2015.02.012
0167-9236/

The human–computer negotiation system accordingly needs much smarter software agents to negotiate with the human negotiators effectively. In automated negotiation, people entrust the software agent to negotiate automatically online, and normally expect that the agent can try different strategies to obtain a better negotiation outcome. In such cases, the ability to quickly and autonomously select an appropriate strategy among the candidates according to negotiation situation changes is a very important perspective for evaluating the designed agent's intelligence level.

Existing research has not yet shed light on such crucial issues as such strategic choices [9], and hence has stalled the much-needed development of the real-world applications of automated negotiation system [17]. Previous models mainly focused on specific protocols (e.g., the alternating offers protocol) and libraries of negotiation strategies (e.g., various concession strategies [18] and trade-off strategies [19]), and have investigated the behaviors of these strategies to determine the most effective strategy in various negotiation situations. Notwithstanding the achievements concerning protocols and strategies, there exists a gap where the strategy selection issue has not been addressed yet. As such, this study is one of the first efforts of advancing this line of research for automated negotiation in e-commerce decision-making. The main objective of this study is to construct and validate a generic, robust decision-making model in an effort to support multi-strategy selection during a course of automated negotiation in e-commerce.

The remainder of this paper is organized as follows. Section 2 discusses related work. Section 3 proposes a conceptual architecture in mathematical form for the negotiating agent. Section 4 presents our negotiating agent's software architecture based on the conceptual model, and describes the logical structure of its main inner components. Section 5 describes a goal deliberation process, which is the core function of the software architecture. Section 6 presents our multi-strategy selection theoretical model and the corresponding algorithm. Section 7 presents the experimental evaluation for our model and algorithm, and discusses the implication of the experimental results. Section 8 elaborates the contributions and limitation of the current work, and draws the picture of the future work. Finally, Section 9 summarizes the findings of this paper and suggests future research directions.

2. Literature review

In order to develop an automated negotiating agent system that has the ability of multi-strategy selection, it is of vital importance to elucidate two necessary issues: (1) how to design a decision making model to support the multi-strategy selection, and (2) how to design an agent architecture as the runtime platform for the decision making model. This section serves to revisit previous work in respect to these issues.

2.1. Negotiating agent architecture

In light of the theoretical foundation and the number of successfully applied systems, the most interesting and widespread agent architecture is the Belief–Desire–Intention (BDI) model [20], which consists of three mental attitudes: (1) beliefs, which capture informational attitudes, (2) desires, which describe motivational attitudes, and (3) intentions, which are deliberative attitudes of agents. However, such agent architecture cannot support various agent applications. In fact, most prior studies (e.g., [21]) did not shed light on the important pre-negotiation step of selecting proper strategies for a specific negotiation situation. Also, most prior studies assumed that the strategies do not change during a course of negotiation. This crucial void was further highlighted with few attempts made to develop models that can effectively choose strategies dynamically. Moreover, most extant models do not support the strategy selection as the negotiation unfolds [9] with an exception of the model proposed by [22,23].

Nevertheless, the work of [22,23] has the potential for further improvements. Firstly, the architecture of their negotiating agents is designed from the buyer's viewpoint and so provides limited guidelines for the architecture design of seller agents. Secondly, their model merely works in one-buyer multi-seller environments. In e-commerce practice, however, other one-to-many and one-to-one negotiation situations also exist. Thus, one of the main aims of this study is to go beyond their spectrum and build a more comprehensive and robust architecture model that can cope with a plethora of negotiation situations. Thirdly, the core of their strategy selection mechanism comprises two matrices: the percentage of success matrix and the payoff matrix, which are imposed artificial subjectivity. This contrasts with the primary underlying of the agent theory. To advance this line of research, the multi-strategy selection model we are going to design based on BDI model will provide the agent with more autonomous ability to cope with the ever changing negotiation situations, without any effect from the external environment, so that the agent can decide by itself to select an appropriate negotiation strategy and complete the decision making process.

2.2. Negotiation strategy

A negotiation strategy is a decision-making model used by the participants to achieve their purposes [24]. In negotiation, one party cannot control its opponent directly, so each should employ certain strategies to persuade the opponent towards the outcome they desired. The work of [25] proposed two typical strategies: (1) Behavior-dependent one is concerned with responsiveness to a partner's behavior, and imitates its behavior in a variety of ways. (2) Time-dependent one completely ignores the reaction of the opponent, i.e. it proposes offers only according to a predetermined time-dependent sequence [26]. Based on these strategies, a negotiating agent can make offers against its opponents complying with a fixed decision function during the course of negotiation. However, to be more successful, an agent needs to adapt to the behavior of its partners and changing environment. Accordingly, effective mechanism should allow a negotiating agent to learn from the previous offers of its negotiating partner in order to predict the partner's future behavior and adapt to it [27].

Much work has been done to equip the agents with the capability of predicting their opponents' negotiation behavior (e.g., price offer, reservation price, and negotiation deadline prediction) by learning from previous negotiations, so that they can achieve more profitable results and better resource utilization [27–30]. For example, in [28], a negotiation model is equipped with feed forward artificial neural network and thus can forecast the opponent's next price proposal according to its past three price proposals. This prediction is very effective and relatively accurate when the curve of the price proposal is regular and smooth. Yet when being near to the inflection point of the curve, the prediction would be increasingly hard and unreliable. In essence, as shown in Fig. 3, the area near the inflection point is the critical place of the negotiation. In addition, in a human–computer negotiation context, predicting human's behavior could be even more difficult because the human's offers do not comply with a fixed offer function.

In theory, negotiating agents are designed to imitate human being's thinking to negotiate autonomously. However, human negotiators usually perform a behavioral game process [31], rather than surmising the opponent's next offer in real world negotiations. Normally it is required to observe the opponent's behavior, including offers, words, actions, and so on, to collect enough information before making the next decision. During this process, imitating the opponent's negotiation behavior is the most conventional method, just as [25] pointed out. In essence, we consider that an intelligent method for the agent to enhance its capability of learning is not to solely predict the opponent's behavior, but to quickly adjust its offer strategy according to the opponent's changing proposals. This lays the theoretical foundation for the multi-strategy selection, so that we can further associate the conceptual underpinning with the practical endeavors on how to translate and accomplish the idea in negotiating agent system.

Departing from the prior studies [27,32], we conceive a possible way of conducting multi-strategy selection: combining the behavior-dependent and time-dependent to take both the opponent's negotiation history and the time factor into consideration. With this novel method, our study makes significant contributions to the research community with the following advantages: (1) our negotiating agent can select one suitable strategy among the candidates to deal with the changing negotiation situation; (2) our agent can imitate the opponent's offer behavior, while following the time-dependent; and (3) our novel model can easily be implemented and only requires a few negotiation series, comparing to the agent's deadline expiration round, to collect data needed by our selection algorithm.

3. The negotiating agent's conceptual architecture

This section presents the conceptual architecture of our negotiating agent. More specifically, we construct the negotiating agent's conceptual architecture based on the Belief–Goal–Plan (BGP) model, which is extended from the classical BDI model. Once the negotiating agent's concept is formally constructed, its software architecture can also be derived.

Belief (B) is the negotiating agent's understanding and cognition of the negotiation environment, including the domain knowledge, the environment parameters, the opponents' believes, and so on. It is the foundation for the agent's negotiation decision-making. It can also be viewed as the negotiating agent's knowledge [13], which is updated dynamically during the agent's negotiation decision-making process. It also contains some static beliefs (e.g., negotiation strategy and the utility model) set by the users when a negotiating agent is instantiated. An agent's knowledge then leads to accomplish the initial negotiation goals or to react to certain offers from its negotiating partners. As such, we propose:

Definition 1. A negotiating agent's belief is defined as $B = <I, R, S, M>$, where

(i) I represents the beliefs triggered by interaction, which is received from the environment or other negotiating agent during the negotiation process.
(ii) R represents the run time beliefs, which are the records acquired when the agent executes its reasoning function. These beliefs may change over time.
(iii) $S = \{s_1, s_2, \cdots, s_n\}$ is a set of negotiation strategies performed by the negotiating agent. Where negotiation strategy s_i is a function $s : I \rightarrow O$, meaning that the agent receives some input proposals (in the set of I) from its opposing negotiation party, and implements the current negation strategy, and then makes some new offers proposals (in the set of O) against that of its negotiating partner.
(iv) $M = \{m_1, m_2, \cdots, m_n\}$ is the utility model, which is made up of several utility functions of the different proposals based on different strategies.

Definition 2. A negotiating agent is a tuple of $<B, G, P, C$, listen, deliberate, plan, react$>$, where:

(i) B is the negotiating agent's belief;
(ii) G is a set of the goals, which represent the states of the negotiation that the agents wish to achieve;
(iii) $P = \{p_0, p_1, \cdots, p_n\}$ is a set of the negotiation plans, which are the concrete actions an agent may carry out to reach its goals;
(iv) $C = \{c_1, c_2, \cdots, c_n\}$ is the negotiating agent's capability;
(v) *listen* : $\wp(B) \times GS \times LS \rightarrow \wp(B)$ is the listening function, which represents the new beliefs generated from the current beliefs, the global state (GS), and the local state (LS);
(vi) *deliberate* : $\wp(B) \times \wp(G) \rightarrow \wp(G)$ is the function describing the negotiating agent's deliberation process, which takes sets of beliefs and current goals to generate new *sets* of goals;
(vii) *plan* : $\wp(B) \times \wp(G) \rightarrow \wp(P)$ is the function that plans the negotiation acts; and
(viii) *react* : $GS \times LS \rightarrow \wp(P)$ is the reaction function.

In the above negotiating agent model, the *Goal G* is concrete, momentary negotiation desires of an agent. That is, for any goal it has, an agent will more or less directly engage into suitable actions until it considers the goal being reached, unreachable, or not desired any more. The *Capability C* is an encapsulated agent module composed of beliefs, goals, and plans [33]. Function *deliberate* selects the goals to be activated from the existing option goals based on the current negotiation situation and strategies (see Section 5 for a detailed discussion). The *plan* consists of a series of actions that the agent will take to achieve a certain goal according to an agent's current beliefs and goals. To promote the reactive ability of the negotiating agent, sometimes the negotiating agent can bypass the above deliberation process and respond directly to the changing environment through function *react*.

4. The negotiating agent's software architecture

This section presents the software architecture of our negotiating agent, which is designed for describing internal structure of negotiating agents based on the agent concept architecture model defined in Section 3.

Autonomy is the basic capability of a negotiating agent. Our negotiating agent can negotiate, without any intervention and guidance from human, according to its inner state and the input from the outer environment. We utilize the Theory of Practical Reasoning [20] to achieve the designing aim, for which the software developer can simply inform the agent of the things that need to be negotiated, without instructing them how to achieve the negotiation goal.

The software architecture of a negotiating agent and the negotiation reasoning process are depicted in Fig. 1, where the reasoning consists of two interleaved components: the deliberation process and the means–end reasoning process. The detailed process of negotiation can be described as the following steps:

Step 1 The negotiating agent receives bargaining information from other negotiating agents or humans.
Step 2 The incoming messages trigger the event-listening function to inform the agent of the changes of the negotiation status. If the incoming message is unnecessary or difficult to process (e.g., a wrong message), the agent will rapidly reply an exceptional message (this process performs the *react* function in the concept architecture model). If the incoming messages are processable, the informed events will be added to the beliefs base, which will be updated accordingly.
Step 3 The negotiating agent begins to handle the deliberation situations by applying varieties of the negotiation strategies, which have been stored in the belief base as a static belief, to generate negotiation goal options, which is so-called negotiation desires representing the possible negotiation goals that the agent tries to achieve. The negotiating agent then chooses among those possible goals based on certain constraints.
Step 4 All the negotiation desires will be submitted to the goal deliberation mechanism to determine which one will be selected as the final activated goal to be implemented (see Section 6 for detailed discussion). A selection model calculates the utility of all the current negotiation desires, and selects a goal with the largest utility from the options. Steps 3 and 4 together form what is called the deliberation process (see Section 5 for further discussion).

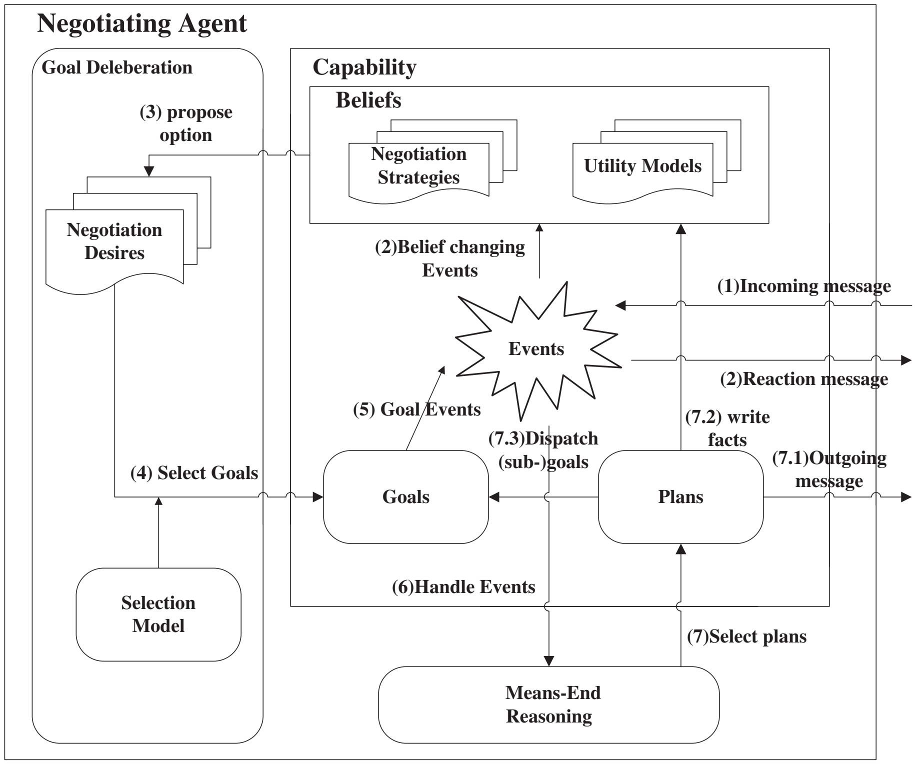


Fig. 1. The system architecture of negotiating agent.

Step 5 The selected goal triggers the listening events mechanism to inform the agent of the changes.
Step 6 The agent monitors the changes and drives the means end reasoning mechanism to handle it.
Step 7 The means–end reasoning mechanism searches the plan base to select an appropriate plan to realize what the current goal wants to do. This process may use the data stored in the belief base. In fact, the result of implementing plan has three directions as follows:

Step 7.1 The negotiating agent carries out speech act planning, and selects the appropriate speech act to express the content of the negotiation goal, and communicate with the other agent.
Step 7.2 The data of the negotiation process generated in the current round are stored into the agent's belief base. This step writes data back into the belief base.
Step 7.3 Besides the negotiation goals, the agent has some other goals to be realized, such as goals for driving communication process. These goals need to be activated in the plan body and dispatched as sub-goals.

In the following subsections, the realization of each of these main concepts in negotiating agent will be described in XML from the buyer's perspective.

4.1. Capability

The capability realizes the set of capabilities in the negotiating agent's concept architecture model. Capabilities allow for packaging a subset of beliefs, plans, and goals into an agent module and reuse this module wherever needed. We use two capabilities in our system *jadex.planlib.Protocols* and *jadex.planlib.DF*, which has been realized in the JADEX system as follows:

```
<capabilities>
        <capability name="procap" file="jadex.planlib.Protocols"/>
        <capability name="dfcap" file="jadex.planlib.DF"/>
</capabilities>
```

The protocol is in charge of the communication between the agents. The protocols capability enables an easy goal-driven usage of some often-used FIPA protocols. The DF is the directory facilitator, which is used to support the agents to find each other.

4.2. Belief base

The belief base is a container for the agent's current beliefs, which realizes the set of beliefs in the negotiating agent's abstract architecture model. The beliefs are designed as follows:

```
<beliefs>
    <beliefset name="orders" class="Order"></beliefset>
    <belief name="time" class="long" updaterate="1000"><fact>System.currentTimeMillis()</fact></belief>
    <belief name="initial_orders" class="Order[]" exported="true"/>
    <beliefset name="negotiation_reports" class="NegotiationReport"/>
    <belief name="strategy1" class="TimeDependentStrategy"><fact> new TimeDependentStrategy()</fact></belief>
    <belief name="strategy2" class="ResourceDependentStrategy "><fact> new ResourceDependentStrategy()</fact>
    </belief>
    <belief name="strategy3" class="BehaviourDependentStrategy "><fact> new BehaviourDependentStrategy()</fact>
    </belief>
    <belief name="utility_model" class="UtilityFunction"><fact> new UtilityFunction()</fact></belief>
</beliefs>
```

In the above codes, we define some initial beliefs, such as orders, initial orders, time, and negotiation report. The order contains some required attributes for purchasing commodity, including title, deadline, start price, and start time. The negotiation report contains user-relevant data (i.e., the order and details about negotiation and the time). Besides, we define some static beliefs, three negotiation strategies, and one utility model. The strategies are time dependent ones, resource dependent ones, and behavior dependent ones (realized by three Java objects: TimeDependentStrategy( ), ResourceDependentStrategy( ), and BehaviourDependentStrategy( ), respectively). A Java object UtilityFunction( ) realizes the utility model, which has various forms depending on the negotiation situation and the user's demand.

4.3. The goal structure

The goal structure stores the negotiation goal, and implements the set of goals in the negotiating agent's concept model. The structure is designed as follows:

```
<goals>
    <achievegoal name="purchase" recur="true" recurdelay="5000">
        <parameter name="order" class="Order"><bindingoptions>$beliefbase.initial_orders</bindingoptions></parameter>
        <creationcondition>$beliefbase.initial_orders!=null</creationcondition>
        <targetcondition>Order.DONE.equals($goal.order.getState())</targetcondition>
```

```
        <failurecondition>$beliefbase.time>$goal.order.getDeadline().getTime()</failurecondition>
    </achievegoal>
    <performgoal name="select" retry="true" exclude="never" >
        <contextcondition><!-- will be introduced in section 5--></contextcondition>
    </performgoal>
</goals>
```

In the above codes, we define three kinds of goals, the *achieve* goal, the *perform* goal, and the goals referenced from the capabilities. The *achieve* goal “purchase” is the buyer agent's final aim. All the negotiation behavior of the buyer agent is to achieve this goal. Therefore, there are some conditions (i.e., <creationcondition>, <targetcondition> and <failurecondition>) to judge when the goal should be created, and whether or not the goal is implemented successfully or not. The *perform* goal “select” is in charge of the goal deliberation process. Its main function is to select appropriate negotiation strategy to make the proposal for the current negotiation situation. It has some contextual conditions to control this process. These conditions will be discussed in detail in Sections 5 and 6.

4.4. The plan specification

The plan specification implements the set of plans in the concept model. At runtime, plans are instantiated to handle events and achieve goals. The plan is designed as follows:

```
<plans>
        <plan name="purchase_plan">
                <body class="PurchasePlan" /><trigger><goal ref="purchase"/></trigger>
        </plan>
        <plan name="select_strategy_plan">
                <body class="SelectStrategyPlan" /><trigger><goal ref="select"/></trigger>
        </plan>
</plans>
```

In the above codes, we define 3 plans: the purchasing plan, the strategy-selecting plan, and the proposal-evaluating plan. The purchase plan's body is realized in a Java object PurchasePlan( ), and triggered by the goal “*purchase*”. The strategy-selecting plan's body is realized in a Java object SelectStrategyPlan( ), and triggered by the goal “*select*”. The specific algorithm of the function SelectStrategyPlan( ) will be introduced in details in Section 6.

5. The goal deliberation process

This section details the negotiating agent's goal deliberation process, which plays an important role in the multi-strategy selection. In order to explain it clearly, we illustrate the relationship in Fig. 2, which shows one snippet of the whole negotiation circle. It describes how the negotiating agent carries out goal deliberation. To distinguish between just adopted (i.e., desired) goals and actively pursued goals, a goal lifecycle is introduced consisting of three goal states: option, active, and suspended.

The whole process of the goal deliberation runs as the following steps:

Step 1 The negotiating agent creates new negotiation goals (desires) according to a creation condition, which is based on the belief from the environment and existing predefined negotiation strategies.

Step 2 When the goals are adopted, they become options and then are added to the agent's goal base, which is a queue data structure (i.e., complying with the first-in-first-out rule).

Step 3 The adopted goals come into state transition stage (i.e., circulating among the option, active and suspend states) according to a selection condition, which is based on a utility model to determine how to select an active goal. The selected negotiation goal can best satisfy the current needs of the negotiating agent, e.g., the present biggest utility value. The unselected optional goals will turn into suspended goal set, which is an intermediate state between the options and active. Besides, the active goal can also be suspended when the implementing condition does not exist (e.g., the network provisionally breaks down).

Step 4 All the unselected goals and the suspended goals will finally be dropped to finish the goal deliberation process according to the drop condition.

As shown in Fig. 2, there are three conditions for conducting the transferring of the goal states, including the creation condition, the selection condition, and the drop condition. The drop condition takes effect when the current round of negotiation is going to an end. The creation condition represents how the negotiating agent generates the negotiation goals. In our system, a classic negotiation strategy model in Section 6.1 implements the creation condition. The selection condition represents how the negotiating agent selects the active goal among some optional goals. In our system, the selection condition is implemented via a multi-strategy selection model that will be proposed in Subsection 6.2.

6. The multi-strategy selection model

This section presents our method for strategy selection. The simplest negotiation model is a bilateral negotiation with a single attribute. In most cases, however, the negotiators have to process several attributes of the product at the same time [1,18]. Before making concession, the negotiator should try to trade off among the different attributes, when they cannot trade off a satisfied result, they might concede according to the predefined concession strategies, evolving to a similar process with the single attribute negotiation. As a result, we just consider the price in our model.

6.1. The time dependent negotiation strategy

Our strategy selection model is based on Faratin's time-dependent concession model, which indicates that an agent is likely to concede more rapidly if it needs to reach an agreement by a deadline [25]. As depicted in Fig. 3, there is actually a family of concession curves, which can be defined simply by varying the value of parameter β determining the convexity degree of the curve. The shape of the each concession curve represents a human's negotiation behavior. As there are infinite proposal curves (corresponding to infinite values of β, one for each curve) included in the solution space, theoretically speaking,

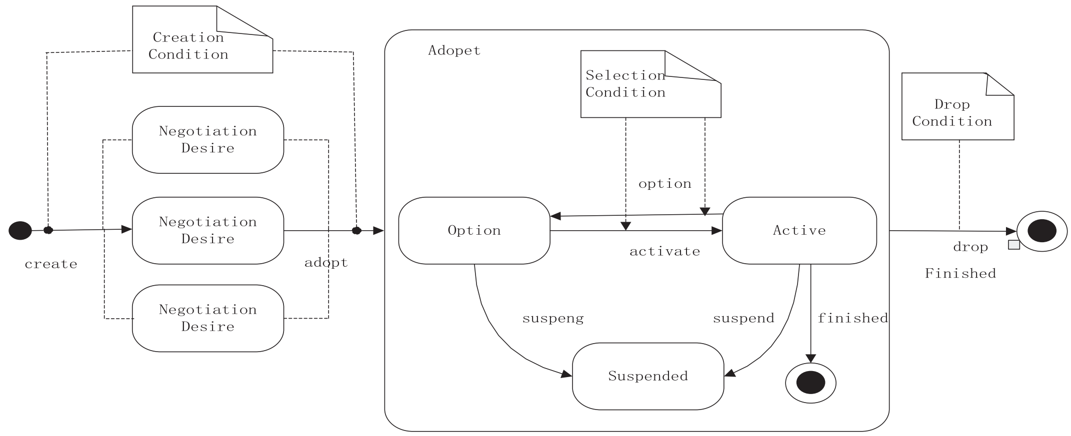


Fig. 2. The negotiating agent's goal deliberation.

the model covers the entire possible proposal curves the human being might choose to make concessions during a course of the negotiation. The task of our multi-strategy selection model is to select among all of these proposal curves dynamically to deal with the ever changing opponent's negotiation behavior, rather than fixing on one proposal curve from the beginning to the end of the negotiation as the prior studies did.

There are two different patterns of behavior: (1) the Boulware, discriminated by $\beta < 1$, maintains the offered value until the time is almost exhausted, whereupon they concede up to the reservation value; and (2) the Conceder, discriminated by $\beta > 1$, leads the agent to go quickly towards its reservation value. The curve with $\beta = 1$ represents the intermediate state between Boulware and Conceder.

The family of the proposal curves can be defined by function $\alpha(t)$ as follows:

$$\alpha_j^a(t) = \exp^{\left(1-\frac{\min(t, t_{\max}^a)}{t_{\max}^a}\right)^{\beta} \ln K_j^a} \tag{1}$$

where a is the agent's name, j denotes the negotiation issue, t is a time factor used to decide which value to offer in the next round of negotiation, t^a_{max} is the time by which agent a must have completed the negotiation, and K_j^a is a constant that when multiplied by the size of the interval, determines the value of issue j to be offered in the first proposal by agent a. So, we have $\alpha_j^a(0) = K_j^a$ and $\alpha_j^a(t^a_{max}) = 1$.

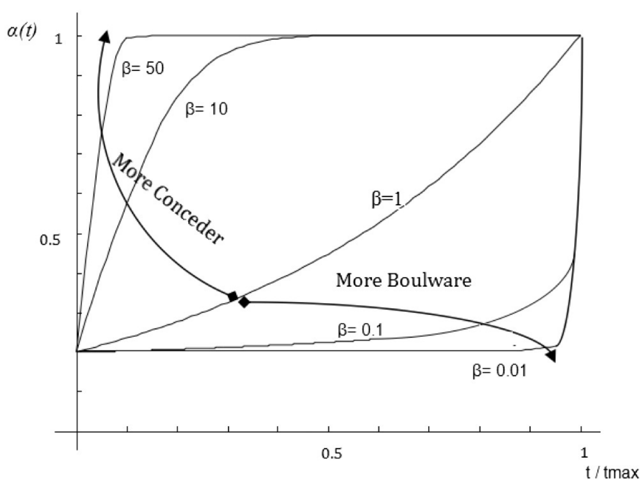


Fig. 3. The exponential functions for the computation of $\boldsymbol{\alpha}(\boldsymbol{t})$. Time is presented as relative to $\boldsymbol{t^a_{max}}$ [25].

6.2. The selection model

The subsection proposes our multi-strategy selection method mainly for the concession tactics. In the real life negotiation, the negotiator often keeps learning its partner's negotiation behaviors and then adjusting its current strategy to a proper one at a proper time to respond to the opponent's possible price changes. With the opaque of both negotiators' strategies, we can only conjecture, imitate and adjust through the offer prices that we can see. As to the imitation, we do not simply make the agent to imitate the opponent's concession, but imitate the opponent's concession rate, which is the ratio between the two neighboring concessions. That is the main contribution of our strategy selection.

Before presenting the formal model, we introduce two basic concepts first.

Definition 3. A concession is the difference between the agent's two neighboring offer prices, which can be expressed formally from the seller's perspective as:

$$x^t_{s\to b} - x^{t-1}_{s\to b} \tag{2}$$

Definition 4. The concession rate, denoted as θ, is the ratio between the two neighboring concessions, which can be expressed formally from the seller's perspective as follows:

$$\theta = \frac{x^t_{s\to b} - x^{t-1}_{s\to b}}{x^{t-1}_{s\to b} - x^{t-2}_{s\to b}} \tag{3}$$

where $x^t_{s\to b}$ is the price offered by seller s to buyer b at time t.

Although both the buyer and seller can change their strategies during the negotiation, in order to clearly describe the agent's changing process of the strategy, we assume that the seller keeps its negotiation strategy unchanged from the beginning to the end. The buyer can change its negotiation strategy according to the seller's negotiation behavior in the process of the negotiation.

In our model, the seller initiates the negotiation process and makes offers first. In the first three round of the bargaining, seller agent s and buyer agent b offer their prices according to their initial negotiation strategies. When the seller offers its third price, the buyer can calculate the seller's concession rate θ of the first three round by formula (3). Based on the value of θ, the buyer makes the decision whether to change the strategy or not. If the buyer decides to change, it will offer its third price complying with the new strategy. In accordance with the same pattern, the buyer keeps on collecting the seller's last three bid prices in subsequent each rounds of negotiation to get the seller's θ for strategy selection decision. There are three situations as follows:

- When $\theta = 1$, the seller keeps a steady concession rate, or make the same concessions between the last two neighboring offers. So, the buyer will simply keep the current strategy unchanged.
- When $\theta > 1$, the seller accelerates concession to approach its deadline. Thus, to reach an agreement surely, the buyer has to adjust its strategy to cater to the seller. Namely, the buyer imitates the seller's concession rate θ, where the agent can deduce its new strategy function.
- When $\theta < 1$, the seller agent decelerates concession. According to the time dependent tactic model, this kind of situation takes place when the agent makes big concession at the beginning of the negotiation. Then the agent gradually decreases concession to approach its reservation price, and finally terminates at the deadline. In this circumstance, our algorithm lets the buyer agent take $1/\theta$ as its concession rate, from which the buyer agent deduces a different β for a new strategy. The reason why the agent takes $1/\theta$ instead of θ is because $1/\theta > 1 > \theta$, by which the agent can deduce a bigger β, then accordingly develops a new Conceder strategy to cater to the seller's fast concession and reach an agreement quickly, which will be proved by the experiments in Section 7.

As designed in the time dependent tactic model, parameter β solely determines the curve's shape of the negotiation strategy. As different negotiation strategies correspond to different values of β, the multi-strategy selection is actually to select an appropriate value for β. Then we discuss how to get a value for β from known concession rate θ. Firstly, the buyer agent calculates the seller's concession rate θ by formula (3). Secondly, to make an offer for the current round, the buyer imitates θ. The detailed derivation process can be found in Appendix 1.

6.3. The algorithm

Based on the theoretical model proposed in the previous subsections, Fig. 4 shows the formal description of the multi-strategy selection algorithm, which consists of the following seven steps:

Step 1 At the beginning, the first offer is made by the seller.

Step 2 Since the buyer needs 3 sequential offers of the seller to get seller's concession rate, if it is the first two round of negotiation, the process will proceed to step 3; otherwise, the process goes to step 5.

Step 3 If the buyer's current offer is larger than the seller's current offer, the buyer does not need to propose its offer. Instead, the buyer accepts the seller's current offer as a better choice, and goes to the ending point; otherwise, the buyer will propose its current offer to the seller.

Step 4 After the buyer's offer, the seller will make its new offer according to its strategy. If the next offer is less than the buyer's current offer, the seller will accept the buyer's current offer and go to the end; otherwise, sets the new offer of the next round, and go back to step 2.

Step 5 If the current negotiation is between the third round and the negotiation deadline (i.e., the shorter one between the buyer's and seller's deadlines), the flow goes to step 6, otherwise, terminates.

Step 6 The system calculates the seller's concession rate based on its three sequential offers. Then:

Step 6.1 If the seller's concession rate is greater than 1, the buyer will imitate the seller's concession rate by formula A.1 (see it in the appendix) and change its negotiation strategy based on the selection model to make its new offer against the seller's offer.

Step 6.2 If the seller's concession rate equals 1 (i.e., the seller concedes steadily during the last three offers), the buyer will keep using its current negotiation strategy to respond to the seller until the seller has some obvious change in its subsequent concessions.

Step 6.3 If the seller's concession rate is less than 1 (i.e., the seller is taking a Conceder strategy), the buyer will imitate the reciprocal of the seller's concession rate by formula A.2, and accordingly adjust its strategy to meet the seller's keen intention to reach an agreement quickly. Since the buyer's initial strategy is a Boulware one ($0 < \beta < 1$), for the first time the buyer detects the seller's conceder intention (i.e., $t = 2$ in Fig. 4), the buyer instantly changes to the strategy which β equals 1 to make its third offer in the negotiation. After that, the buyer will select a Conceder strategy ($1 < \beta < 50$). By so doing, its strategy is changed to a Conceder one from the original Boulware one.

Step 7 After the buyer decides the strategy for the next round, it will make the new offer, and go to step 3.

Note that steps 2 and 3 embody the agent's accepting strategy, which defines the condition that the agent will accept the opponent's offer.

7. Experimental evaluation of the strategy selection model

This section will conduct lots of experiments to evaluate the effectiveness of our negotiation strategy selection model, which will practically benefit real automated negotiation system development.

7.1. Environments and tactics setting

An environment is characterized by the number of agents in negotiation, the issues to be discussed, the deadline for reaching an agreement, and the expectations of the agents. Since it is impossible to discuss infinitely possible environments, we only consider a representative environment in which we can assess an agent's negotiation performance to test our strategy selection algorithm. In the same environment, if an agent with the strategy selection algorithm can obtain a higher success rate and better negotiation results than those without the algorithm, then the effectiveness and correctness of the algorithm can be proven.

To this end, the experiments are simply limited to a bilateral negotiation between a buyer and a seller over the single issue of price. The buyer agent has the strategy selection ability, its initial offer price is 5, and its reservation prices are 80. The seller agent negotiates according to a pre-set strategy with the initial price offer 115, and the reservation price 40. Thus, the experimental environment is uniquely defined by: $[\beta^b, \beta^s, t^b_{max}, t^s_{max}, k^b, k^s, min^b_{price}, max^b_{price}, min^s_{price}, max^s_{price}]$, i.e., time available to make an agreement (t^b_{max}, t^s_{max}), the initial offer (k^b, k^s), and price the intervals of the buyer and seller. In the following experiments, we set $[k^b, k^s, min^b_{price}, max^b_{price}, min^s_{price}, max^s_{price}] = [0.1, 0.1, 5, 80, 40, 115]$, in which we refer from [25] to set $k^b = k^s = 0.1$ for both agents. Then several groups of experiments will be conducted according to different ranges of β, t^b_{max} and t^s_{max}. More specifically, there are 3 different relationships between t^b_{max} and t^s_{max}, i.e., $t^b_{max} > t^s_{max}$, $t^b_{max} < t^s_{max}$ and $t^b_{max} = t^s_{max}$, under which the different negotiations will be discussed in the following experiments.

In the experiments, we select a finite range of tactics because we cannot do infinite experiments (there are infinite time-dependent tactics because the range of β is infinite). For analytical tractability, we follow the setting of [25]: $0 < \beta < 50$, in which $0 < \beta < 1$ defines Boulware tactics, and $1 < \beta < 50$ defines Conceder tactics. At the very beginning, the seller chooses the value of β randomly to set its strategy curve, and then fixes its strategy at this value of β until the end of this negotiation. Meanwhile, the buyer runs the strategy selection algorithm to

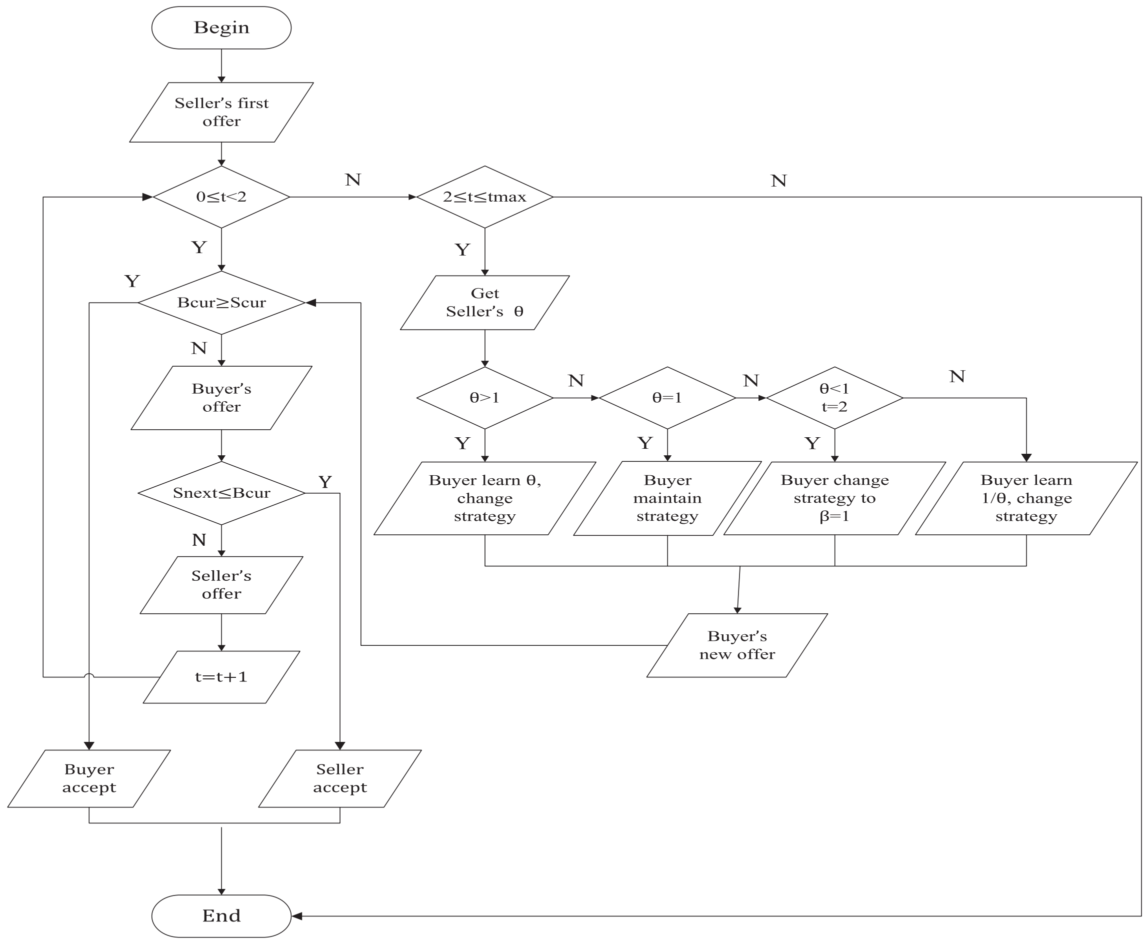


Fig. 4. Buyer's multi-strategy selection algorithm, where ***t*** is the time, ***θ*** is the concession rate of seller agent, ***Bcur*** and ***Scur*** are the current bid of the Buyer and Seller respectively; ***Snext*** is Seller's bid for the next round of negotiation.

choose an appropriate strategy among strategy candidates with different values of β to respond to the seller's different offers.

7.2. Experimental evaluation measurement

Referring to [26,32], we propose four measures to evaluate our model:

(1) *Success Rate* (SR). The most important factor to evaluate an automated negotiation is how much percentage of the negotiations using the system can reach an agreement. The success rate reflects the validation of the multi-strategy selection algorithm for all the negotiation experiments. Suppose among n times of experiments, n_1 times reach a deal. Then the success rate is:

$$SR = n_1/n \tag{4}$$

(2) *Intrinsic Utility* (IU). The intrinsic benefit is modeled as the agent's intrinsic utility for the negotiation's final outcome. This buyer's and seller's utilities are calculated for a price of $\boldsymbol{x}$ as follows:

$$U_s(x) = \frac{x - min^s}{max^s - min^s} \tag{5}$$

$$U_b(x) = \frac{max^b - x}{max^b - min^b} \tag{6}$$

In certain experiments, we compute the intrinsic utility only for cases in which deals are made.

(3) *Utility Product* (UP). The joint outcome of the negotiation for both sides is indicated by UP. Once and agreement $\boldsymbol{x}$ is achieved, the product of the utilities obtained by both sides is computed as follows:

$$UP = U_s(x) \times U_b(x) \tag{7}$$

(4) *Utility Difference* (UD). The distance between the buyer's and seller's utility is measured by UD. A good agreement offer is the one that has high UP and low UD [32]. Once an agreement $\boldsymbol{x}$ is achieved, the difference of the utilities obtained by both sides is computed as follows:

$$UD = |U_s(x) - U_b(x)| \tag{8}$$

In summary, SR with UP and UD describes the overall effect of the negotiation, while intrinsic utility indicates the impact of the final negotiation result on the buyer and seller agent.

7.3. Experimental hypotheses, procedure, and discussions

This section experimentally compares the effects of strategy-selection and no-strategy-selection.

7.3.1. Experimental hypotheses

The ensuing section elaborates on the hypotheses for the experiments.

According to the strategy selection model (see Section 6), a negotiating agent can select multiple offers in each round in the following way: it first detects the changes of the opponent's concession speed. If the opponent accelerates or decelerates concession, the agent will change its current strategy through the strategy selection algorithm to cater to the opponent's concession change. Thus, by selecting strategies in each round, the agent can approach more quickly to the opponent's offer than the agent that just adopts a fixed strategy, and thus increasing the future likelihood that the two sides meet in the feasible solution area. Therefore, over the negotiation rounds, there is a higher chance that the either party accepts the other party's offer, and consequently the overall success rate is more likely to be enhanced. Thus, we propose:

Hypothesis 1. An agent with the strategy-selection algorithm can achieve a higher success rate than those without strategy-selection algorithm (i.e., adopt a single negotiation strategy from beginning to end).

Before the buyer agent and the seller agent start a negotiation, their original strategy (Boulware or Conceder) should be checked out in advance. Intuitively, Boulware is a conservative and secure strategy, and accordingly can help the adopter ensuring not to lose much profit, though it might not lead to a deal; while the Conceder is much easier to reach an agreement with others as it concedes much at the beginning of the negotiation regardless of the possible profit loss. Therefore, using the Boulware tactic as the initial concession strategy would be a better choice for the agent to ensure its benefits. Thus, we have:

Hypothesis 2. In the case that one side's negotiation strategy is unknown to its opponent, Boulware tactic is a better choice for the other side because a better overall negotiation effect for both sides (described by SR, UD and UP) can be created.

As discussed in Section 2.2, the time factor can significantly impact on negotiators' behavior. Intuitively, by using our strategy selection algorithm, the agent is able to not only improve the success rate for making a deal as we assume in Hypothesis 1, but also get a better joint income. Although we expect that the strategy-selection mechanism can perform well in all the three cases: $t^b_{max} > t^s_{max}$, $t^b_{max} < t^s_{max}$ and $t^b_{max} = t^s_{max}$, it still needs to be verified one by one in our experiments. So, we have:

Hypothesis 3. In the case that the buyer's deadline is larger than the seller's, the buyer agent with our strategy-selection mechanism can get a better negotiation result than the one without the mechanism.

Hypothesis 4. In the case that the buyer's deadline is less than the seller's, the buyer agent with our strategy-selection mechanism can get a better negotiation result than the one without the mechanism.

Hypothesis 5. In the case that the buyer's deadline is equal to the seller's, the buyer agent with our strategy-selection mechanism can get a better negotiation result than the one without the mechanism.

7.3.2. Experimental procedures

The experiments are divided into two parts. The first part has no strategy selection, which is then used in the second part. In each part, the value of the seller's β is generated randomly in two intervals: (0, 1), and (1, 50). In each case, the deadline of the buyer and seller is divided into three groups: $t^b_{max} > t^s_{max}$, $t^b_{max} < t^s_{max}$ and $t^b_{max} = t^s_{max}$, in which the exact values of the maximum trading time are also randomly generated. Thus, we need to conduct 12 experiments. In each experiment, we run negotiation 200 times. In every negotiation process, the seller selects negotiation strategy randomly, and keeps that strategy until the end of negotiation. The buyer initially uses one of the Boulware tactics randomly (i.e., to choose a $\beta \in (0, 1)$), and then the strategy selection process is used to deal with the seller's offer. This ensures that the initial negotiation strategies for both sides are different in each experiment.

7.3.3. Experimental results and discussion

Consider Hypothesis 1 first. The data in Table 1 show that the adoption of strategy-selection can truly improve the success rate of the negotiation when other experiment environments are set to be consistent. Especially, when the seller adopts the Conceder strategy, the buyer's strategy-selection algorithm has an obvious enhancement in success rate. Empirically the experimental results support the hypothesis.

Moving onto Hypothesis 2, it can be divided into two situations as follows. Firstly, towards the buyer with no-strategy selection, in comparison with the seller with Conceder tactics, the seller with Boulware tactics can reach a higher success rate, higher utility product, and lower utility difference. The experiments basically support the hypothesis.

Secondly, when the buyer adopts strategy selection mechanism to negotiate with the seller, we can find out from Table 1 that in comparison with the seller with Boulware tactics, the seller with Conceder tactics reaches a higher success rate, but the utility product (UP) and the utility difference (UD) are worse. This implies that the conceder tactic can help the seller to get more agreements because it is an aggressive strategy sacrificing the agent's own utility as compensation. When we consider all the factors and situations, the Boulware tactic is a better choice for both sides to obtain a better outcome.

The typical situations of the experiments are shown in Fig. 5. The dotted lines denote buyer's each selected strategy curve, the solid line denotes buyer's final strategy curve, which leads the buyer to reach an agreement with the seller, and the star point denotes the final agreement price. The figure is divided into three columns based on the relation between t^b_{max} and t^s_{max}. In the first row (a, b and c) the seller adopts Boulware tactic, while in the second row (d, e and f) the seller adopts Conceder.

Taking Fig. 5(a) and (d) as two examples, we can clearly see the buyer's strategy selection process. The buyer's initial strategy curves are the nethermost one, which certainly cannot lead to an agreement because the expected agreement point is below the seller's reservation price. The buyer negotiates with the seller along its initial strategy curve for the first two offers till it receives the seller's third offer. The buyer then decides to change its negotiation strategy using the selection model. Every time the buyer changes its strategy, it can get a new offer against its opponent. Finally, they reaches an agreement that both sides can accept in the area of the feasible solutions. From Fig.(a) and (d) we can see the following: without the strategy selection, the seller and buyer cannot make a deal, or the deal made is not win–win in some cases, where the strategy selection model can help the agent improve the negotiation result.

Through the above analysis, we can conclude basically that Hypothesis 2 holds.

From the viewpoint of Hypothesis 3, when the buyer's deadline is larger than the seller's, we can find from Table 1: our strategy-selection algorithm significantly raises up the negotiation success rate (SR), but affects less upon the both side's mean of intrinsic utility, UP and UD. This means that our algorithm can help the buyer agent get more deals although no more utility can be obtained from the negotiation for both sides. So, the experimental results support the hypothesis.

The typical situations of our negotiation experiments are shown in Fig. 6. Taking Fig. 6(a) and (c) as an example, the two figures depict two different negotiation situations in the same environment (i.e., the same value for [t^b_{max}, t^s_{max}, k^b, k^s, min^b_{price}, max^b_{price}, min^s_{price}, max^s_{price}]), plot (a) depicts the negotiation situations without our strategy selection mechanism, while (c) depicts the situation with our strategy selection mechanism. Obviously, the seller was more eager to complete the transaction, while the buyer was in no hurry, and had more space to adjust its strategy towards benefiting itself. So, when the buyer adopted the strategy-selection algorithm, the negotiation success rate increased significantly.

Table 1
Comparison of the experiment results.

Buyer's negotiation strategy	Seller's β	T	Success rate (SR)	Mean of buyer's intrinsic utility (BIU)	Mean of seller's intrinsic utility (SIU)	Mean of utility product (UP)	Mean of utility difference (UD)
No-selection $0<\beta<1$	$0<\beta<1$ (Boulware)	$t^b_{max} > t^s_{max}$	12.5%	0.4235	0.1099	0.0465	0.3136
		$t^b_{max} < t^s_{max}$	14.5%	0.1422	0.3912	0.0556	0.2490
		$t^b_{max} = t^s_{max}$	100%	0.3830	0.1503	0.0576	0.2327
	$1<\beta\le 50$ (Conceder)	$t^b_{max} > t^s_{max}$	3%	0.5288	0.0045	0.0024	0.5243
		$t^b_{max} < t^s_{max}$	49%	0.4794	0.0540	0.0259	0.4254
		$t^b_{max} = t^s_{max}$	7%	0.5195	0.0138	0.0072	0.5057
Selection $0<\beta<1$	$0<\beta<1$ (Boulware)	$t^b_{max} > t^s_{max}$	51%	0.4167	0.1166	0.0486	0.3001
		$t^b_{max} < t^s_{max}$	13%	0.1142	0.4191	0.0479	0.3049
		$t^b_{max} = t^s_{max}$	100%	0.3476	0.1858	0.0646	0.1618
	$1<\beta\le 50$ (Conceder)	$t^b_{max} > t^s_{max}$	95%	0.5235	0.0098	0.0051	0.5137
		$t^b_{max} < t^s_{max}$	97.5%	0.4345	0.0988	0.0429	0.3357
		$t^b_{max} = t^s_{max}$	100%	0.5063	0.0271	0.0137	0.4792

From the viewpoint of Hypothesis 4, when the buyer's deadline is less than the seller's, if the seller adopts Boulware tactics, our strategy-selection algorithm affects insignificantly upon the negotiation result. When the seller adopts Conceder tactics, the effect is fairly significant. Hence, Hypothesis 4 is supported partially. The experimental data shows that when the seller adopts the Boulware tactic, whether or not the buyer takes strategy-selection,it does not affect on the success rate. In fact, the buyer's intrinsic utility decreases obviously because the buyer is more eager to reach an agreement than the seller.

For Hypothesis 5, two sides have the same deadline, which is an ideal state that seldom happens in the real negotiation. When the seller adopts Boulware tactics, whether or not the buyer selects strategy, both sides can reach an agreement by the deadline. However, when seller adopts Conceder tactics, the strategy-selection algorithm has a significant effect upon the success rate, but not the overall negotiation effects in terms of UP and UD. The experimental results support Hypothesis 5 partially.

7.4. Implications

The above experimental results are significant to real-life negotiations and contribute to the development of the practical negotiation systems. The implication can be concluded into the following three aspects:

(1) Since our multi-strategy selection mechanism outperforms the fixed strategy, it should be put into consideration preferentially in the design and development of automated negotiation systems.
(2) According to the analysis of Hypothesis 2, when we set the initial negotiation strategy in the real-life negotiation or for an automated negotiation system, the Boulware tactics should be first considered because it is a safer strategy. Since Conceder is much riskier

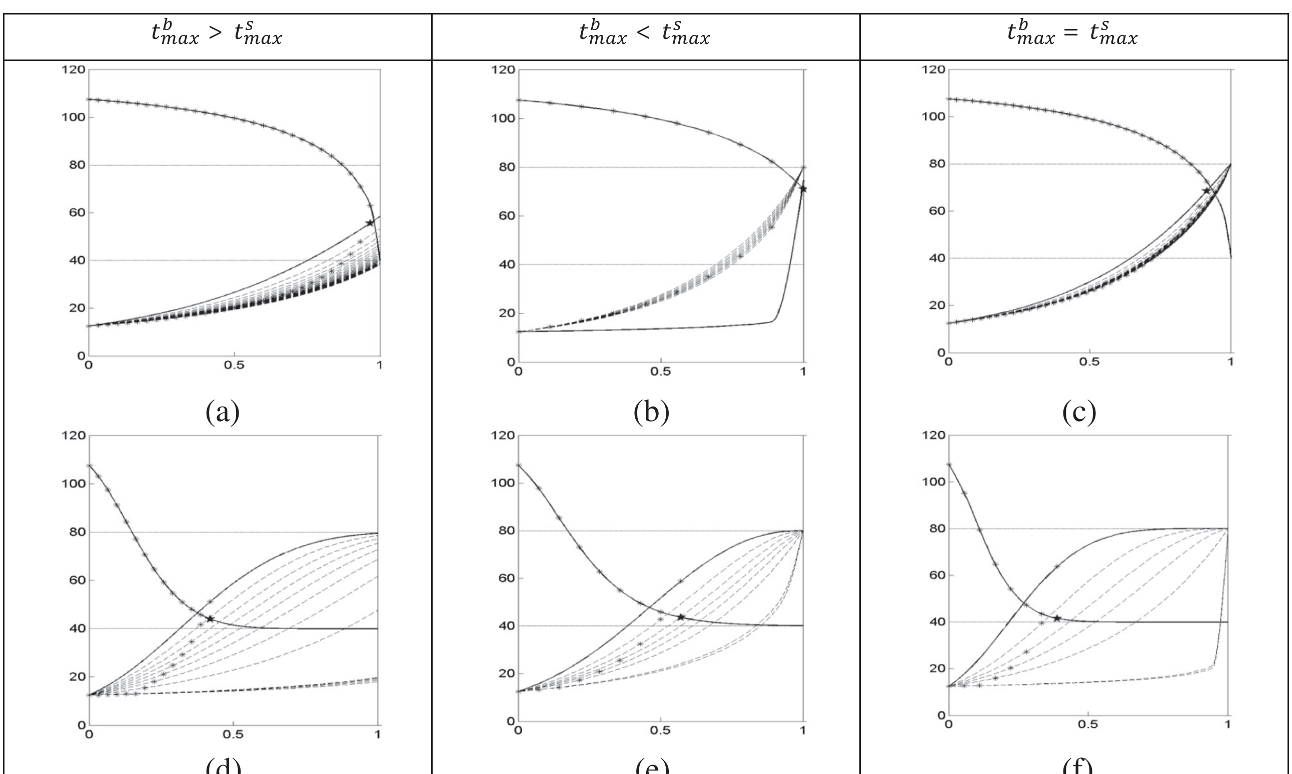
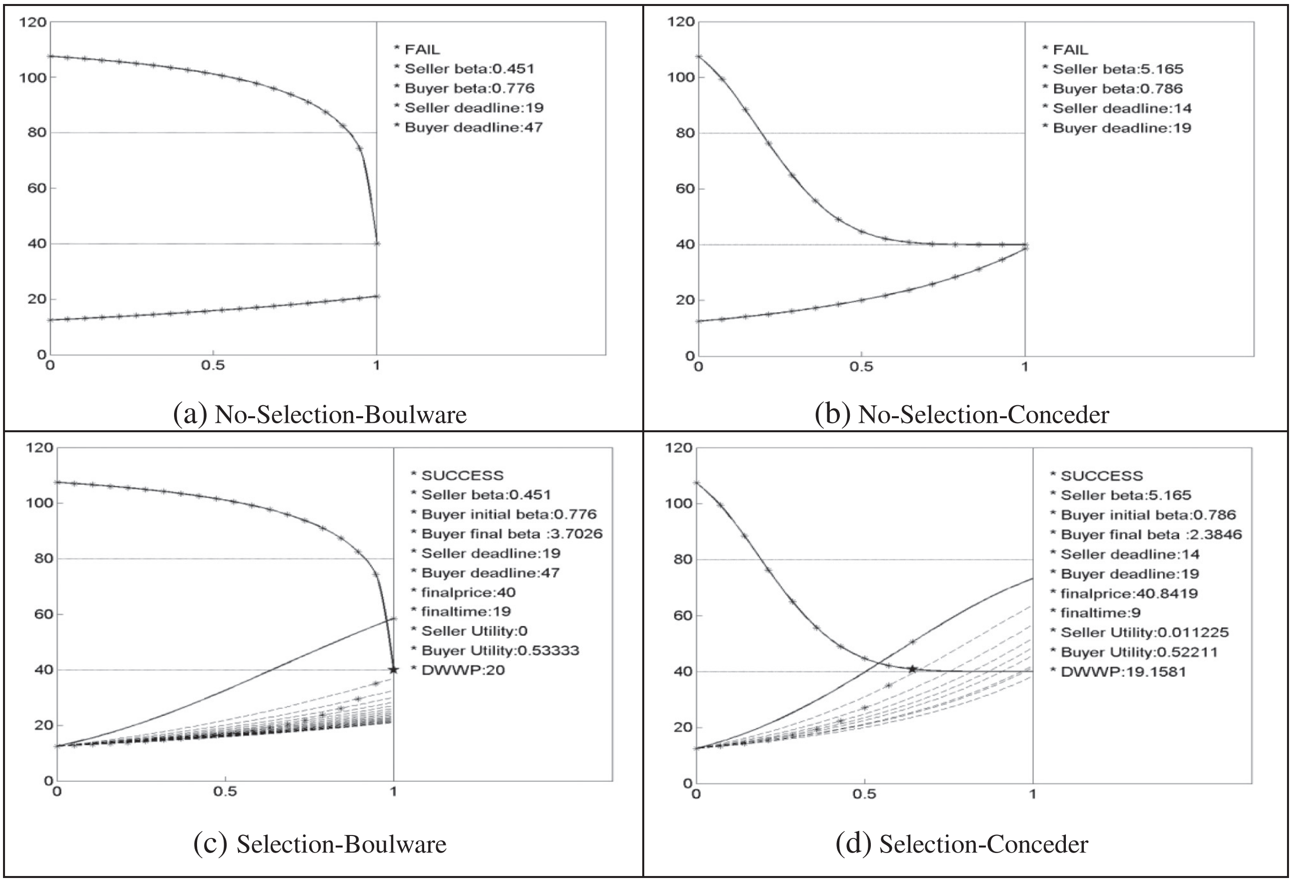


Fig. 5. Typical negotiation situation in case of buyer taking selection strategy.



(a) No-Selection-Boulware (b) No-Selection-Conceder (c) Selection-Boulware (d) Selection-Conceder

Fig. 6. Typical negotiation situation comparison in case of buyer's deadline larger than the seller's.

than Boulware, the Conceder tactic will be a good choice in the case of the negotiator hopes to make a deal quickly regardless of profit loss. As to its counterpart, adopting Boulware tactic as initial strategy is reasonable but should be used in conjunction with our strategy-selection mechanism to enable the negotiator responding to the opponent according to the ever-changing negotiation situation. This ensures successful transactions as well as better utilities.

(3) In respect to the negotiation deadline, no matter the counterparty adopts Boulware or Conceder, the party that adopts our strategy-selection algorithm should set a longer deadline than its opponent because the longer deadline can guarantee that the strategy-selection enabled agent has more chances to adjust its offer, so as to take the control of the negotiation situation, thus guaranteeing the success rate of the negotiation and also making the whole benefit of the both sides at a higher level. However, the subsequent question is how to make the agent's negotiation deadline longer than the human's in the human–computer negotiation where the agent acts as the seller on behalf of an online company, while the human is the buyer. After all, the deadline is very private information and is protected to be opaque with each other. Our solution is to set the agent's negotiation deadline initially as the system deadline, which is informed to both sides to obey at the beginning of the negotiation. This means that the end point of the negotiation occurs whether the participants reach an agreement or not. If the human's deadline is less than or equal to the system deadline, according to the analysis of Hypotheses 3 and 5, the system can guarantee the success rate of the negotiation and the benefit of the both sides. If a human intentionally stalls in order to wait for the agent make big concessions near the deadline, it is necessary to design a new mechanism that can help the agent to detect the human's real intention and adjust its critical negotiation parameters (e.g., reservation price or deadline) dynamically. This will be conducted in our future work.

As we discussed in the previous sections, human–computer negotiation systems are very promising in the future B2C e-commerce. Our multi-strategy selection model is extremely useful for building the negotiating agents in such systems because these systems need much smarter agents to handle the human's complicated price offers. According to the empirical analysis, when developing a human–computer negotiation system, the following principles should be followed: (1) The software agent acts as a seller (i.e., an online company) to provide products or services and the human side is a human buyer in a B2C e-commerce oriented negotiation system. (2) The seller agent has a built-in multi-strategy selection module and employs the Boulware tactic as its initial strategy no matter the human employs Boulware or Conceder, or any other random proposal curves. (3) The inner mechanism should guarantee that the agent's negotiation deadline is set to be equal to the system deadline.

7.5. Human–computer experiments

The results presented in this study refer only to the automated negotiation case, in which the buyer and the seller are both software agents. Although our final aim is to contribute to the development of human–computer negotiation system, the study of computer–computer negotiation is definitely the indispensible stage to test the validity and feasibility of the designed agent model, just as other prior studies did [18]. Considering that the human decision behaviors are generally uncertain and not easy to be substituted by an agent with a simple and fixed bargaining strategy, our experiment ran the agent–agent negotiation 2400 times. In each negotiation the seller agent adopted different fixed offer strategy in the strategy space, which is supposed to cover as much the human's possible offer curves as possible.

Nevertheless, we have also considered the case where humans propose their own offer against software agents. Based on our strategy selection model and the three principles for constructing human–computer negotiation system, we developed a prototype experimental platform for conducting some human–computer negotiation experiments. In order to process the human's uncertain negotiation behaviors, two improvements have been appended to the strategy selection function: (1) the new model is designed to handle the situation when the human's offer curve is non-monotonic, as the model proposed in this paper only can handle the seller's monotonic offers; (2) the new model is able to deal with the situation where the human has the intention of stalling time so that the agent can maintain its deadline superiority. In addition, new negotiation protocol was designed for facilitating the human-agent context. The full results are outside the scope of this paper, and will be comprehensively reported elsewhere in the future studies.

Here we provide only a brief synopsis of the results. Nearly 120 college students were asked to negotiate against our agents. Interestingly, the results show that comparing the performance of human and agent in the human–computer negotiation, the agent attending negotiation obtained higher success rate, while the agents received somewhat better results (the intrinsic utility) than the humans. It is important to note that almost all human players took the task of negotiation seriously, and quite a number of them attempted to find the limitations of the agent to defeat the agent.

8. Contributions, limitations and future work

8.1. Contributions

This research proposes a goal deliberated agent architecture equipped with a multi-strategy selection model for automated negotiation system, and experimentally evaluates its effects in the computer–computer negotiation. The significant contribution of this study lies in three aspects:

The first contribution is the goal deliberated agent architecture, which can support the agent to autonomously select an appropriate strategy to negotiate with the external environment without any human intervention once the negotiation starts. Unlike the multi-strategy selection mechanism proposed in [22] that is constructed upon subjective probability matrix, our architecture model excludes the human influents. Hence, our model accords with the main connotation of the agent theory, i.e., autonomy. Comparing with [23] which designed a negotiating agent architecture only from the buyer's viewpoint in a one-buyer-many-seller context, our approach goes beyond their spectrum as a more general and robust architecture model for both buyer and seller. Therefore our model has the ability to cope with a variety of negotiation situations in e-commerce, including one-to-one, one-to-many, and many-to-many. On the other hand, since implementing an autonomous agent architecture model is always a pending problem in the prior studies [21,34], we utilize goal deliberation technology to integrate strategy selection mechanism into the agent architecture from a theoretical layer. Furthermore, we elaborate in detail the concrete implementation method for the architecture model from a software engineering perspective, so it is possible to realize a practical agent system with strategy selection capability.

In addition to contributing to the system architecture, the second contribution this paper presents is a multi-strategy selection model complementing the research of negotiation strategy. There are two major approaches to designing the negotiation strategy: the heuristic-based approach and the machine learning approach [35]. (1) The heuristic-based approach abides by a fixed concession function to implement the concession process, e.g., [18,25,26,36]. However, different from the previous studies, the multi-strategy selection model proposed in this paper enables the agent to select an appropriate offer strategy in the time-dependent strategy space, so that it can deal with the ever-changing negotiation situation according to the opponent's offer. The experimental results show that, comparing with the benchmark work [25], our model leads to a higher negotiation success rate. (2) The machine learning approach, on the other hand, mainly predicts the opponent's future negotiation behavior relying on the availability of past negotiation history as a training set [37] or requiring a large number of rounds of offer exchanges in a negotiation episode [30] before the agent can build an effective learning model. The proposed strategy selection model in this paper is not to predict but to imitate the opponent's negotiation behavior so that it can better adapt to the opponent's ever changing offers, consequently improve the negotiation success rate. Moreover, the machine learning approach needs rather more historical data to complete the prediction process [27–29]. In our model, however, only 3 rounds of past negotiation data are needed to create effective feedback against the current negotiation progress. More significantly, in terms of the theory and technology of automated negotiation, our multi-strategy selection model actually creates a novel concession mode, which is the main method for the both sides to reach an agreement. The extant method normally utilized a preset concession mode, usually a monotonic [25] or segmented [18] concession function, to realize the concession process. Beyond the prior studies, our strategy selection mechanism has no preset mode and the concession offer curve is completely generated dynamically, thereby increasing the flexibility and robustness of the negotiation system to a maximum extent. As such, our mechanism provides a new thought for the study of concession model in automated negotiation.

On the more practical side, the third contribution is that, through massive experiments, valuable empirical knowledge (including agent's initial settings for negotiation strategy, reservation price and deadline) for building and using the human–computer negotiation system has been acquired, hence representing a step close to more realistic practical e-commerce agent-based negotiations. Our study proves that the ability to dynamically change and adjust the negotiation strategy according to the opponent's offer is a required function for a negotiating agent. That can significantly help the practical design and implementation of the construction and application of a human–computer negotiation system.

8.2. Limitations and future work

Despite the noteworthy contributions, the findings of this research should be interpreted within the defined problem scope. It is hoped that several avenues open up for future research.

Firstly, our current research is mainly based on the Faratin's time-dependent and behavior-dependent tactics. The underlying reason for why the other strategy models are not considered so far is because theoretically the classic time-dependent strategy model covers almost all possible offer curves as shown in Fig. 3. Nevertheless, revised Faratin's model and more novel negotiation strategies and protocols have been proposed [18,38]. It should be of interest as a topic for future research to challenge our boundaries and examine the proposed models in more complex negotiation settings and new strategy context.

Secondly, we only consider a single attribute (i.e., price) in our multi-strategy selection model while the negotiators may have to process several attributes of the product simultaneously. In terms of the multi-attribute negotiation, the common solution is a combination of trade-off and concession. As discussed in [11], buyer and seller firstly trade-off among different negotiation issues at a same utility level. When they cannot find a common interest at the level, they will proceed to compromise on the utility of the multi-attributes according to predefined concession strategies. Therefore, the multi-attribute negotiation problem eventually comes down to a concession problem on the overall utility over multi issues, which has the same effect as concession on a single price. We discuss our multi-strategy selection model only in the single attribute negotiation context because the context does not affect the evaluation of a concession model's function. As we discussed above, one of our significant contributions is creating a novel multi-strategy selection model, which mainly provides a new idea for designing concession model. Therefore, at current stage, we limit our scope in the single price concession, so that the result can be adapted to multi-attribute negotiation. Since the multi-attribute negotiation is more general in real-life, it is necessary that the future studies should improve our multi-strategy selection model to fit the multi attribute negotiation.

Thirdly, this paper's work mainly focuses on agent–agent negotiation. So we propose to conduct human-agent negotiation experiments in the future. We have already extended our computer–computer negotiation into a human–computer negotiation experimental platform, and also conducted human–computer negotiation experiments in order to present the critical empirical data and experiences for developing and applying the human–computer negotiation systems, which will widely be used in the future B2C e-commerce.

9. Conclusions

The central idea of this paper is that the strategy selection is a novel negotiation concession model, and should be considered as a requisite component in negotiating agent architecture. Extending this line of research, this study is built upon the goal deliberation mechanism to enable a negotiating agent to select appropriate strategy dynamically to deal with the ever-changing opponent's offer and get agreement successfully. Experimental results confirmed that, compared with the conventional fixed strategy, the proposed multi-strategy selection mechanism leads to a higher counterpart acceptance ratio, greater counterpart utility, and joint utility. The contribution of this study leads to further valuable empirical experiences for constructing a human–computer negotiation system. This study is expected to bridge the gap between the theoretical and practical aspects of the negotiating agent development.

Based on the experimental findings, we feel confident in recommending that a negotiating agent incorporating the proposed multi-strategy selection model be deployed to complement existing automated negotiation technologies for facilitating computer–computer or human–computer negotiations. Our theory-informed design and experimental research demonstrate the significant potential of negotiating agent technologies in enhancing the efficiency of the negotiated transactions and increasing the negotiation success rate, thus takes a step closer to fulfill the promise of the human–computer dynamic transactions for the next generation of e-commerce.

Acknowledgement

The authors would like to thank reviewers and editors for their helpful comments and suggestions. This research was supported by the National Natural Science Foundation of China under Grant 70902042, and Mountain Sea Foundation under Grant 2013221029.

Appendix 1. The derivation process for getting the new strategy from the known concession rate

For $\theta > 1$, we have:

$$\theta = \frac{x^t_{b\to s} - x^{t-1}_{b\to s}}{x^{t-1}_{b\to s} - x^{t-2}_{b\to s}} \tag{A.1}$$

Please note the difference between formulas (3) and (A.1): $x^t_{s\to b}$ for Eq. (3), and $x^t_{b\to s}$ for (A.1).

For $\theta < 1$, we have:

$$\frac{1}{\theta} = \frac{x^t_{b\to s} - x^{t-1}_{b\to s}}{x^{t-1}_{b\to s} - x^{t-2}_{b\to s}} \tag{A.2}$$

The work of [25] defines the vector of values for issue j proposed by agent a to agent b at time t by:

$$x^t_{a\to b}[j] = \begin{cases} \min^a_j + \alpha^a_j(t)\left(\max^a_j - \min^b_j\right) & \text{if } V^a_j \text{ is decreasing} \\ \min^a_j + \left(1-\alpha^a_j(t)\right)\left(\max^a_j - \min^a_j\right) & \text{if } V^a_j \text{ is increasing} \end{cases} \tag{A.3}$$

where the range of the values acceptable to agent a for issue j is interval $[min^a_j, max^a_j]$, and V^a_j is the score that agent a assigns to a value of issue j in the range of its acceptable values. Since only the buyer's monotonically increasing offer price is involved in the negotiation, we can simplify formula (A.3) as:

$$x^t_{b\to s} = \min^b + \left(1-\alpha^b(t)\right)\left(\max^b - \min^b\right) \tag{A.4}$$

The above formula means the agent's offer price $x^t_{b\to s}$ at the current time t is a point on the strategy curve, which is solely determined by $\alpha^b(t)$ in formula (A.4). As a result, the subsequent work is to solve $\alpha^b(t)$ and then proceed to get the agent's offer price $x^t_{b\to s}$ at time t. In the same way of $x^t_{b\to s}$, we can get $x^{t-1}_{b\to s}$ and $x^{t-2}_{b\to s}$. By substituting $x^t_{b\to s}$, $x^{t-1}_{b\to s}$ and $x^{t-2}_{b\to s}$ into formulas (A.1) and (A.2), we obtain:

$$\alpha^b(t) = \begin{cases} (\theta+1)\alpha^b(t-1) - \theta\alpha^b(t-2) & \text{if } \theta > 1 \\ \left(\frac{1}{\theta}+1\right)\alpha^b(t-1) - \frac{1}{\theta}\alpha^b(t-2) & \text{if } \theta < 1 \end{cases} \tag{A.5}$$

From formula (A.4), we can obtain

$$\alpha^b(t-1) = \frac{\max^b - x^{t-1}_{b\to s}}{\max^b - \min^b} \tag{A.6}$$

Similarly, we can get $\alpha^b(t-2)$. Then, in formula (A.5), all the items are known except $\alpha^b(t)$. Thus, we can calculate it as a specific value of A. Integrating A with formula (1) for the value of $\alpha^b(t)$, we get:

$$A = \exp^{\left(1 - \frac{\min\left(t, t^b_{\max}\right)}{t^b_{\max}}\right)^{\beta} \ln K^b} \tag{A.7}$$

From (A.7), the buyer can obtain the new value of β and thus find a new strategy against the seller by:

$$\beta = \ln\left(\frac{\ln A}{\ln K^b}\right) \Big/ \ln\left(1 - \frac{\min\left(t, t^b_{\max}\right)}{t^b_{\max}}\right) \tag{A.8}$$

Then, we can get the buyer's new negotiation strategy as follows:

$$x_{b\to s} = \min^b + \left(1 - \exp^{\left(1 - \frac{\min\left(t, t^b_{\max}\right)}{t^b_{\max}}\right)^{\beta} \ln K^b}\right)\left(\max^b - \min^b\right) \tag{A.9}$$

where t is independent time variable and $x_{b\to s}$ is the dependent offer price variable.

References

[1] X. Luo, et al., A fuzzy constraint based model for bilateral, multi-issue negotiations in semi-competitive environments, Artificial Intelligence 148 (1) (2003) 53–102.

[2] X. Luo, K.M. Sim, M. He, *A knowledge based system of principled negotiation for complex business contract*, in *Knowledge Science*, Engineering and Management, Springer, 2013. 263–279.
[3] G. Adomavicius, A. Gupta, Toward comprehensive real-time bidder support in iterative combinatorial auctions, Information Systems Research 16 (2) (2005) 169–185.
[4] M. Bichler, A. Gupta, W. Ketter, Research commentary—designing smart markets, Information Systems Research 21 (4) (2010) 688–699.
[5] A.R. Lomuscio, M. Wooldridge, N.R. Jennings, A classification scheme for negotiation in electronic commerce, Group Decision and Negotiation 12 (1) (2003) 31–56.
[6] G.E. Kersten, H. Lai, Negotiation support and e-negotiation systems: an overview, Group Decision and Negotiation 16 (6) (2007) 553–586.
[7] T. Baarslag, et al., Evaluating practical negotiating agents: results and analysis of the 2011 international competition, Artificial Intelligence 198 (5) (2013) 73–103.
[8] E. de la Hoz,, M.A. López-Carmona, I. Marsá-Maestre, *Trends in Multiagent Negotiation: From Bilateral Bargaining to Consensus Policies*, in *Agreement Technologies*, Springer, 2013, pp. 405–415.
[9] F. Lopes, M. Wooldridge, A.Q. Novais, Negotiation among autonomous computational agents: principles, analysis and challenges, Artificial Intelligence Review 29 (1) (2008) 1–44.
[10] W. Ketter, et al., Real-time tactical and strategic sales management for intelligent agents guided by economic regimes, Information Systems Research 23 (4) (2012) 1263–1283.
[11] Y. Yang, S. Sharad, C.X. Yunjie, Alternate strategies for a win–win seeking agent in agent–human negotiations, Journal of Management Information Systems 29 (3) (2013) 223–255.
[12] G. Adomavicius, A. Gupta, D. Zhdanov, Designing intelligent software agents for auctions with limited information feedback, Information Systems Research 20 (4) (2009) 507–526.
[13] X. Luo, et al., KEMNAD: a knowledge engineering methodology for negotiating agent development, Computational Intelligence 28 (1) (2012) 51–105.
[14] R. Lin, S. Kraus, Can automated agents proficiently negotiate with humans? Communications of the ACM 53 (1) (2010) 78–88.
[15] R. Lin, et al., Training with automated agents improves people's behavior in negotiation and coordination tasks, Decision Support Systems 60 (IS) (2014) 1–9.
[16] T. Bosse, C.M. Jonke, Human vs. computer behavior in multi-issue negotiation, Rational, Robust, and Secure Negotiation Mechanisms in Multi-Agent Systems2005.
[17] K.-J. Lin, E-commerce technology: back to a prominent future, Internet Computing, IEEE 12 (1) (2008) 60–65.
[18] L. Pan, et al., A two-stage win–win multi-attribute negotiation model: optimization and then concession, Computational Intelligence 29 (4) (2013) 577–626.
[19] X. Luo, N.R. Jennings, N. Shadbolt, Acquiring user tradeoff strategies and preferences for negotiating agents: a default-then-adjust method, International Journal of Human-Computer Studies 64 (4) (2006) 304–321.
[20] M.E. Bratman, What is intention, Intentions in Communication (1990) 15–32.
[21] A. Fabregues, C. Sierra, HANA: a human-aware negotiation architecture, Decision Support Systems 60 (2014) 18–28.
[22] T.D. Nguyen, N.R. Jennings, Coordinating multiple concurrent negotiations, Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, vol. 3, 2004.
[23] T.D. Nguyen, N.R. Jennings, Managing commitments in multiple concurrent negotiations, Electronic Commerce Research and Applications 4 (4) (2006) 362–376.
[24] T. Skylogiannis, et al., DR-NEGOTIATE—a system for automated agent negotiation with defeasible logic-based strategies, Data & Knowledge Engineering 63 (2) (2007) 362–380.
[25] P. Faratin, C. Sierra, N.R. Jennings, Negotiation decision functions for autonomous agents, Robotics and Autonomous Systems 24 (3) (1998) 159–182.
[26] C.-F. Lee, P.-L. Chang, Evaluations of tactics for automated negotiations, Group Decision and Negotiation 17 (6) (2008) 515–539.
[27] J. Brzostowski, R. Kowalczyk, Modelling partner's behaviour in agent negotiation, AI 2005: Advances in Artificial Intelligence, 2005, pp. 653–663.
[28] M. Oprea, An adaptive negotiation model for agent-based electronic commerce, Studies in informatics and Control 11 (3) (2002) 271–279.
[29] A. Monteserin, A. Amandi, A reinforcement learning approach to improve the argument selection effectiveness in argumentation-based negotiation, Expert Systems with Applications 40 (6) (2012) 2182–2188.
[30] I. Roussaki, I. Papaioannou, M. Anangostou, Building automated negotiation strategies enhanced by MLP and GR neural networks for opponent agent behaviour prognosis, Computational and Ambient, Intelligence, 2007, pp. 152–161.
[31] C. Camerer, Behavioral Game Theory: Experiments in Strategic Interaction, Princeton University Press, 2003.
[32] R. Ros, C. Sierra, A negotiation meta strategy combining trade-off and concession moves, Autonomous Agents and Multi-Agent Systems 12 (2) (2006) 163–181.
[33] P. Busetta, et al., Structuring BDI agents in functional clusters, Intelligent Agents VI. Agent Theories, Architectures, and Languages, Springer, 2000. 277–289.
[34] M. Cao, M.Y. Kiang, BDI agent architecture for multi-strategy selection in automated negotiation, Journal of Universal Computer Science 18 (10) (2012) 1379–1404.
[35] N.R. Jennings, et al., Automated negotiation: prospects, methods and challenges, Group Decision and Negotiation 10 (2) (2001) 199–215.
[36] P. Faratin, C. Sierra, N.R. Jennings, Using similarity criteria to make issue trade-offs in automated negotiations, Artificial Intelligence 142 (2) (2002) 205–237.
[37] R.Y. Lau, et al., Mining trading partners' preferences for efficient multi-issue bargaining in e-business, Journal of Management Information Systems 25 (1) (2008) 79–104.
[38] J. Zhan, et al., *A Fuzzy Logic Based Model of a Bargaining Game*, in *Knowledge Science*, Engineering and Management (2013) 387–403.

Dr. Mukun Cao is an Associate Professor of Management Science in the School of Management at the Xiamen University, China. He earned his Ph.D. in Information Systems from Harbin Institute of Technology, China. He has been a visiting scholar to the University of Illinois, Urbana and Champaign, California State University, Long Beach, and Teaching Entrepreneurial Thought and Action training in Babson College. His research interests include supply chain management oriented collaboration, coordination and negotiation theory and technology, electronic commerce oriented negotiation support system and automated negotiation system theory and technology, multi-agent system theory and software agent technology. He has abundant occupational experience in industry. He was a software engineer in the China Aerospace Science and Technology Corporation (Beijing), Strong Software Co., Ltd. He has been certified as the International Project Management Professional (IPMP) by the International Project Management Association (IPMA). He is also invited to be the special advisor for some corporations in China, including Kingdee International Software Group (China) Co., Ltd. for ERP, Shanghai Power Technology Co., Ltd. for project management. He has published many research articles in academic journals and international conferences. He is currently serving as a managing Editor for Service Science journal, and serves as reviewer for several international and domestic journals, including Electronic Commerce Research and Application, Journal of Management, and Journal of Information Systems.

Dr. Xudong Luo is currently a professor of Artificial Intelligence at Sun Yat-sen University, China. He published one book and more than 140 papers including 2 in top journal {\em Artificial Intelligence}, one of which has been highly cited by, for example, MIT and CMU research groups. Prof. Luo has international recognized reputation: co-chair and (senior) PC members of about 60 international conferences or workshops, including major conferences IJCAI and AAMAS, and referees for many international journals such as top journal {\it Artificial Intelligence}. He is also invited to make a presentation of his work in more than 10 universities internationally, including Imperial College. His research focus is on the areas of agent-based computing, fuzzy sets and systems, decision theory, game theory, and knowledge engineering. Prof. Luo has supervised or co-supervised about 30 master students, Ph.D. students and research fellows.

Dr. Xin (Robert) Luo is an Associate Professor of Management Information Systems and Information Assurance in the Anderson School of Management at the University of New Mexico, USA. He earned his Ph.D. in Information Systems from Mississippi State University. His research interests include information assurance, innovative technologies for strategic decision-making, and cross-cultural management. His research has been published in journals including *Journal of the Association for Information Systems, European Journal of Information Systems, Communications of the ACM, Decision Support Systems, Information Systems Journal, Journal of Strategic Information Systems, Communication of the Association for Information Systems, and Computers & Security*, etc. He is currently serving as an ad-hoc Associate Editor for *MIS Quarterly*, Associate Editor for *European Journal of Information Systems, Journal of Electronic Commerce Research*, and *International Conference on Information Systems*, etc.

Xiaopei Dai, born in 1989, is currently a PHD student of Xiamen University, Xiamen, China. Her major is Management Science and Engineering. Her current research interest is the simulation and artificial intelligence.