Continuous-Time Negotiation Mechanism for Software Agents

Bo An, Kwang Mong Sim, Liang Gui Tang, Shuang Qing Li, and Dai Jie Cheng

***Abstract*—While there are several existing mechanisms and systems addressing the crucial and difficult issues of automated one-to-many negotiation, this paper develops a flexible one-to-many negotiation mechanism for software agents. Unlike the existing general one-to-many negotiation mechanism, in which an agent should wait until it has received proposals from all its trading partners before generating counterproposals, in the flexible one-to-many negotiation mechanism, an agent can make a proposal in a flexible way during negotiation, i.e., negotiation is conducted in continuous time. To decide when to make a proposal, two strategies based on fixed waiting time and a fixed waiting ratio are proposed. Results from a series of experiments suggest that, guided by the two strategies for deciding when to make a proposal, the flexible negotiation mechanism achieved more favorable trading outcomes as compared with the general one-to-many negotiation mechanism. To determine the amount of concession, negotiation agents are guided by four mathematical functions based on factors such as time, trading partners' strategies, negotiation situations of other threads, and competition. Experimental results show that agents guided by the four functions react to changing market situations by making prudent and appropriate rates of concession and achieve generally favorable negotiation outcomes.**

***Index Terms*—Automated negotiation, negotiation agents, one-to-many negotiation.**

I. INTRODUCTION

AUTOMATED negotiation [19], [21] among software agents is becoming increasingly important because automated interactions between agents [4], [8], [29] can occur in many different contexts (e.g., negotiation for resources [9]). In terms of the number of agents participating in negotiations, agent-based automated negotiation can be divided into three cases [6], namely: 1) one-to-one negotiation (bilateral negotiation); 2) many-to-many negotiation; and 3) one-to-many negotiation. Compared with auction mechanisms [18], one-to-many interactive negotiation is more flexible. For example, agents can adopt different negotiation strategies with different trading partners (alternatives), and negotiations can be taken under different negotiation environments and protocols [20].

In one-to-many negotiation (take the negotiation between a buyer and several sellers as an example), there are two alternatives: 1) buyer negotiates sequentially with all the sellers and 2) buyer negotiates concurrently with these sellers. Generally, the buyer gets more desirable negotiation outcomes when it negotiates concurrently with all the sellers in competitive situations in which there are information uncertainty and deadlines [16], [17]. In this paper, we assume that an agent negotiates concurrently with its trading partners.

Let a negotiation cycle be the time spent in a round of negotiations and the reaction time of a trading partner be the time from an agent's proposing to its receiving a counterproposal from the trading partner. In existing general one-to-many negotiation mechanisms and systems (e.g., [1], [7], [20], and [30]), taking the negotiation between a buyer and several sellers as an example, the buyer's negotiation with the set of sellers is divided into several rounds (indexed by $\{0, 1, 2, \ldots\}$), i.e., negotiation is conducted in discrete time. A problem with the general one-to-many negotiation mechanism [7], [12], [16], [20] is that during negotiation, no matter how long an agent has to wait and how many proposals have been received, the agent cannot propose until it has received proposals from all its trading partners. In actual negotiation environments, as agents may have different negotiation strategies, reasoning mechanisms, communication time, constraints, and preferences, an agent generally receives its trading partners' proposals at different times in each round after it sent proposals to all its trading partners at the same time (i.e., different trading partners have different reaction times).[1] Therefore, the general one-to-many negotiation mechanism is not flexible enough when negotiation agents are of different reaction times.

To overcome the limitation of the general one-to-many negotiation mechanism, this research focuses on developing a more flexible mechanism than the general one, where an agent can decide when to make a proposal according to the synchronization situations of negotiation, which are determined by the reaction time of each trading partner.

There are three critical issues in designing a flexible one-to-many negotiation mechanism.

1) *How to coordinate all the subnegotiation threads* (Section II): One-to-many negotiation can be treated as a series of subnegotiation threads, and different subnegotiation threads have different negotiation situations. A coordination strategy concerns issues such as whether all the subnegotiation threads are interactive or not and

Manuscript received July 11, 2005; revised December 10, 2005 and January 7, 2006. This paper was recommended by Associate Editor G. Skarmeta.

B. An, L. G. Tang, S. Q. Li, and D. J. Cheng are with the College of Computer Science, Chongqing University, Chongqing 400044, China (e-mail: boancqu@gmail.com).

K. M. Sim is with the Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, Hong Kong (e-mail: bsim@comp.hkbu.edu.hk).

Digital Object Identifier 10.1109/TSMCB.2006.874686

---

[1] The reasons that bring about different trading partners with different reaction times vary. For example, agents located in different positions in the network may have different communication distances. The factors affecting communication quality of service (QoS), e.g., bandwidth, congestion, and network failure, may result in different communication delays. Agents with different reasoning mechanisms and processing capabilities may have different processing speeds. According to strategies, agents may decide to postpone sending proposals, even though the proposals have already been generated successfully.

how to organize the negotiation of multiple threads. It is obvious that if an agent takes the information of all the subnegotiation threads into account while generating proposals, it will get more favorable negotiation outcomes.

2) *When to generate proposals* (Section III): In order to overcome the limitation of the general one-to-many negotiation mechanism, this paper develops two strategies for deciding when to make a proposal based on the synchronization situations of negotiation.
3) *Negotiation strategy* (Section IV): In order to build more flexible and sophisticated negotiation agents, Faratin *et al.* [3] have devised a negotiation model that defines a range of strategies and tactics for generating proposals. The strategies in [3] are based on time, resource, and behaviors of negotiators.

In Sim's (enhanced) market-driven agents (MDAs) [23]–[28], other essential factors, such as competition, trading alternatives, and differences of negotiators, are also considered. While designing negotiation strategies for agents conducting one-to-many negotiations, this paper considers factors that can significantly either enhance or diminish a negotiator's ability to achieve its objectives, namely: 1) time; 2) trading partners' strategies; 3) negotiation situations of other subnegotiation threads; and 4) competition. Additionally, the performance of the continuous-time one-to-many negotiation mechanism is evaluated by comparing it with the general negotiation mechanism and the desperate strategy in Section V. Section VI summarizes related work, and Section VII concludes this paper.

II. COORDINATION MECHANISMS

For ease of analysis, this paper focuses on single-issue (single-attribute, e.g., price-only) negotiation rather than multiple-issue negotiation (we leave multiple-issue negotiation, which is more complex and challenging than a single-issue negotiation [11], for future research). This paper adopts the alternating offers protocol [22, p.100] for each subnegotiation thread. In each round, an agent proposes according to its negotiation strategy. After receiving a proposal from one of its trading partners, the agent evaluates it. If it is acceptable, the agent accepts it, and the negotiation terminates. Otherwise, the agent puts forward a counterproposal according to its negotiation strategy, and the negotiation proceeds if the deadline is not reached.

While conducting negotiations with several trading partners concurrently, an agent needs to make a decision on the negotiation structure, which is used to organize negotiations, and the coordination strategy, which is used to control and coordinate multiple negotiation threads.

A. Negotiation Structure

1) Centralized Structure: The centralized structure consists of only a complex and powerful agent conducting the negotiation with several trading partners. It is simple and has high efficiency with a small number of trading partners. However, when the trading partners increase, the calculating cost of the agent increases, the processing speed slows down, and the negotiation efficiency decreases.

2) Hierarchy Structure: The hierarchy structure consists of a manager agent and several subagents, and negotiation is composed of multiple subnegotiation threads. When negotiation begins, the manager agent creates several subagents equal to the number of the trading partners, and then, each subagent negotiates with a trading partner. The manager agent manages and coordinates the negotiation of each subagent based on one of the coordination strategies described in Section II-B. Compared with the centralized structure, the hierarchy structure has the following advantages (this paper assumes the hierarchy structure).

1) Increase of trading partners almost has no effect on the negotiation process. When a seller enters a negotiation, the buyer's manager agent only needs to create a new subbuyer to negotiate with the seller agent.
2) System becomes much more robust, and the adjustment (even failure) of a subnegotiation thread will not result in the failure of the whole negotiation.
3) Distribution characteristics are obtained by the hierarchy structure. All the subagents can be distributed into any place throughout the network.

B. Coordination Strategy

Based on the hierarchy structure, an agent negotiating with a set of trading partners needs to make a decision on two levels of negotiation strategies, namely: 1) negotiation strategy for the manager agent and 2) negotiation strategy for the subnegotiators. The negotiation strategy for the manager agent refers to all the rules that control the whole negotiation and coordinate the multiple subnegotiation threads. The following two coordination strategies differentiated by whether the multiple negotiation threads are interactive can be exercised by the manager agent for controlling subnegotiation threads [20].

1) Desperate Strategy: All the subnegotiation threads are independent, i.e., subagents are not aware of the negotiation information of the other subnegotiation threads and the manager agent is anxious to end the negotiation, i.e., once a subagent finds an acceptable proposal, the manager agent accepts it and stops all the subnegotiation threads. If several acceptable proposals are found out at the same time, the manager agent chooses the one with the highest utility.

2) Optimized Strategy: The subnegotiation threads are interactive. During negotiation, each subagent is clearly aware of the negotiation information of the other subnegotiation threads and adjusts its proposals accordingly. Once a subagent finds an acceptable proposal, the manager agent stops all the subnegotiation threads.

Compared with the desperate strategy, the optimized strategy is much more complex. The optimized strategy regards that all the subnegotiation threads mutually influence one another and each subnegotiator can take the information of the other negotiation threads into account while generating new proposals. The following discussion assumes that the manager agent utilizes the optimized strategy.

Suppose that a buyer agent negotiates with n seller agents. While using the optimized strategy as the coordination strategy for the hierarchy structure, there will be information exchange among the manager buyer agent, the n subbuyer agents, and the n seller agents. During negotiation, each subbuyer reports its negotiation status (e.g., its trading partner's proposal) to the manager buyer agent; the manager buyer agent then decides when to make a proposal, decides whether to terminate negotiation, and informs each subbuyer of the negotiation status of the other negotiation threads. Assume that the manager agent exchanges information with all the subnegotiators once in each round, and each subnegotiator exchanges information with its trading partner once in each round. The communication complexity of the aforementioned negotiation scenario is $O(n \times m)$, where m represents how many rounds a negotiation takes. As both n and m are finite, the hierarchy structure with optimized coordination strategy has polynomial communication complexity.

III. WHEN TO MAKE A PROPOSAL

In contrast to the general one-to-many negotiation mechanism, an agent decides when to make a proposal according to the synchronization situations of negotiation in the flexible negotiation mechanism. After the discussion of the problem of evaluating the synchronization situation of a negotiation scenario, this section introduces two strategies for deciding when to make a proposal, namely: 1) fixed-waiting-time-based strategy and 2) fixed-waiting-ratio-based strategy.

A. Synchronization Situation Evaluation

The synchronization situation of a one-to-many negotiation is determined by a set of reaction times of all the trading partners. Let S be the synchronization situation of a one-to-many negotiation scenario, which takes the form

$$S = \frac{\sqrt{D(C)}}{E(C)} \tag{3.1}$$

where C is the set of reaction times of all the trading partners, $C = (C_1, C_2, \ldots, C_n)$, $E(C)$ is the expectation of C, and $D(C)$ is the variance of C. Because $E(C) > 0$ and $D(C) \geq 0$, it follows that $S \geq 0$.

The synchronization situation S increases with an increase of the variance $D(C)$. In terms of the value of the synchronization situation S, synchronization situations can be divided into three different levels, namely: 1) high ($S < 0.2$), 2) general ($0.2 \leq S < 0.5$), and 3) low ($S \geq 0.5$).

Case 1) If $\forall C_i, C_j \in C$ and $C_i = C_j$, then all the negotiation threads have the same reaction time, i.e., $D(C) = 0$. Hence, $S = 0$, i.e., the one-to-many negotiation is of the highest synchronization.

Case 2) If $\forall C_i \in C$ and $C_i - E(C) \to \infty$, then $D(C) \to +\infty$. Hence, $S \to +\infty$, i.e., the one-to-many negotiation is of the lowest synchronization.

While making the decision of when to make a proposal in round $t+1$, an agent may confront the following dilemma.

1) Agent will not propose until it has received the most counterproposals from trading partners (it becomes the general mechanism if the agent does not propose until it has received all the counterproposals). The agent will get more information for proposal generation after receiving more counterproposals. Therefore, the proposal for a trading partner after receiving more counterproposals will not be worse than the proposal after receiving less counterproposals. However, if the agent waits for more counterproposals before generating new proposals, it will take more time to complete a round of negotiations as "waiting" is always time consuming. Consequently, the agent may be not able to take several rounds of negotiation before the deadline approaches.
2) Agent will propose as soon as possible. Although this approach will decrease the time spent in a round of negotiations, the proposal for a trading partner may not be as good as the proposal generated after receiving more counterproposals. As the synchronization situation of a negotiation shows the difference of the set of reaction times of trading partners, it is intuitive to take the synchronization situations into account while deciding when to make a proposal.

A high synchronization situation level (e.g., $S = 0.01$) means that an agent can receive the proposals of all the trading partners almost at the same time. Accordingly, before proposal generation, the agent can wait for more trading partners' proposals after it receives the first proposal. In contrast, when the synchronization situation level is low (e.g., $S = 0.9$), it is intuitive that the agent does not have to wait for more trading partners' proposals after it receives the first proposal as it may take a long time to wait for all of their proposals. Based on the preceding intuitions, we introduce two flexible strategies for deciding when to make a proposal, namely: 1) fixed-waiting-time-based strategy in which the synchronization situations are used to determine the time to wait after an agent receives the first counterproposal (see Section III-B) and 2) fixed-waiting-ratio-based strategy in which the synchronization situations are used to determine the number of counterproposals to wait after an agent receives the first counterproposal (see Section III-C).

B. Fixed-Waiting-Time-Based Strategy

In each round, after a subagent first receives a proposal from its trading partner, it waits for a fixed waiting time, and then all the subagents that received proposals begin to generate new proposals.

Let T_{wait} be the fixed waiting time, which is given by

$$T_{\text{wait}} = [\max(C) - \min(C)] \times \frac{S_{\max} - S}{S_{\max}} \tag{3.2}$$

where $\max(C)$ is the maximum reaction time in C, $\min(C)$ is the minimum reaction time in C, and $S_{\max} = \max(C)/E(C)$. Because

$$\frac{dT_{\text{wait}}(S)}{dS} = \frac{-[\max(C) - \min(C)]}{S_{\max}}$$

and $\max(C) - \min(C)$ and $S_{\max}$ are nonnegative, the slope $dT_{\text{wait}}(S)/(dS)$ is always negative. Hence, if $\max(C) - \min(C)$ and $S_{\max}$ are determined and the negotiation is with a high synchronization level, the agent has to wait for a much longer time; otherwise, it has to wait for a much shorter time. The algorithm for deciding when to make a proposal according to the fixed-waiting-time-based strategy is given in Fig. 1. Additionally, when $T_{\text{wait}} = \max(C) - \min(C)$, the flexible mechanism is equivalent to the general one-to-many negotiation mechanism (Proposition 1).

```
Function: The Fixed Waiting Time Based Strategy
Input: The trading partners' proposals
Output: The set of trading partners to propose
/* This algorithm specifies the procedure for determining
the set of trading partners to propose in each round. */

Let $T_{wait}$ be the waiting time given by (3.2). Let $q$ be
a queue to store proposals of the trading partners. Let
$insert()$ be the function which loads proposals from the
head of the queue $q$ into the queue according to the
proposals' arriving order, just loading one proposal at a
time. But if several proposals arrive at the same time,
they will be loaded together. Let $T_{insert}$ be the executing
time of the function $insert()$. $reached = true$ means
that a consensus has been reached.

Begin
Set $reached = false$ and $q = null$
$insert()$, $t' = T_{insert}$
While $T_{insert} < (t' + T_{wait})$
  If $reached = true$
    Return $q = null$
  End-if
  $insert()$;
End-while
Return $q$
/*The agent proposes to every partner whose proposal is
in the queue $q$. In order to get the set of trading partners
to propose in the next round, the agent should continue
to run this algorithm until a consensus is reached or the
deadline has been reached.*/
End
```

Fig. 1. Algorithm for the fixed-waiting-time-based strategy.

Proposition 1: When $T_{\text{wait}} = \max(C) - \min(C)$, the flexible mechanism is equivalent to the general one-to-many negotiation mechanism.

Proof: In the general one-to-many negotiation mechanism, after a buyer sent proposals to all its trading partners in round t, it will wait until it has received all the counterproposals from all its trading partners before sending new proposals to all its trading partners in round $t+1$. For the flexible one-to-many negotiation mechanism, after sending proposals to its partners at the same time in round $t = 0$, the buyer agent first receives the proposal(s) of the seller agent(s) with reaction time $\min(C)$. Because the fixed waiting time is $T_{\text{wait}} = \max(C) - \min(C)$, the buyer can still receive proposals from agent(s) with reaction time $\max(C) - \min(C) + \min(C)$, i.e., the buyer agent last receives the proposal(s) of the seller(s) with reaction time $\max(C)$ before proposing in round $t = 1$. Then, the buyer agent will send counterproposals to all its trading partners in round $t = 1$ at the same time. Similarly, when $t = 1, 2, \ldots$, the buyer will still receive the proposal(s) of the seller(s) with reaction time $\max(C)$ last before proposing in round $t+1$. Thus, the flexible mechanism is equivalent to the general mechanism if $T_{\text{wait}} = \max(C) - \Xi\min(C)$. ■

C. Fixed-Waiting-Ratio-Based Strategy

In each negotiation round, after a subagent first receives a proposal from its trading partner, it waits until the number of subagents that received proposals reach a fixed ratio, and then all the subagents that received proposals begin to generate new proposals to their trading partners, respectively.

```
Function: The Fixed Waiting Ratio Based Strategy
Input: The trading partners' proposals
Output: The set of trading partners to propose
/* This algorithm specifies the procedure for determining
the set of trading partners to propose in each round. */

Let $R_{wait}$ be the waiting time given by (3.3) and $0 <
R_{wait} \leq 1$. $q$ and $insert()$ have the the same meaning as
that in Fig. 1. Let $n$ be the number of trading partners.
$reached = true$ means that a consensus has been
reached.

Begin
Set $reached = false$ and $q = null$
$insert()$
While $R_{wait} > Length_q/n$
  If $reached = true$
    Return $q = null$
  End-if
  $insert()$;
End-while
Return $q$
/*The agent proposes to every trading partner whose
proposal is in the queue $q$. In order to get the set of
trading partners to propose in the next round, the agent
should continue to run this algorithm until a consensus is
reached or the deadline has been reached.*/
End
```

Fig. 2. Algorithm for the fixed-waiting-ratio-based strategy.

Let R_{wait} be the fixed waiting ratio, which is given by

$$R_{\text{wait}} = \frac{S_{\max} - S}{S_{\max}}. \tag{3.3}$$

Each parameter here has the same meaning as that in Section III-B. Because

$$\frac{dR_{\text{wait}}(S)}{dS} = -\frac{1}{S_{\max}}$$

and $S_{\max}$ is nonnegative, the slope $dR_{\text{wait}}(S)/dS$ is always negative. Therefore, an agent will wait for a much longer time with the increase of the synchronization level S.

The algorithm for deciding when to make a proposal according to the fixed-waiting-ratio-based strategy is given in Fig. 2. The trading partners for proposal generation are given after running the algorithm. In addition, when $R_{\text{wait}} = 1$, the flexible mechanism becomes the general one-to-many negotiation mechanism (Proposition 2).

Proposition 2: If $R_{\text{wait}} = 1$, the flexible mechanism becomes the general one-to-many negotiation mechanism.

Proof: When $R_{\text{wait}} = 1$, an agent negotiating with a set of trading partners will wait until it has received all the proposals from all its trading partners before sending counterproposals to these trading partners, which is similar to the general one-to-many negotiation mechanism. ■

IV. SUBAGENTS' NEGOTIATION STRATEGY

While adopting the hierarchy architecture, the one-to-many negotiation can be treated as multiple threads of bilateral negotiation. Given the condition that the manager agent takes the optimized strategy, each subagent needs to make a decision on its negotiation strategy.

While there are many factors affecting the agents' decision making (e.g., deadline, competition, trading opportunity, and eagerness) [3], [23]–[28], the following two issues should be taken into account in designing negotiation strategies for subagents conducting one-to-many negotiations: 1) every subagent needs to take the negotiation information of the other negotiation threads into account because the multiple negotiation threads mutually influence one another and 2) in concurrent one-to-many negotiation, an agent may have more than one trading partner to reach an agreement. After an analysis of the main factors affecting subagents' decision making, the subagents' strategies are determined by the following four decision functions, namely: 1) time-dependent function T; 2) trading-partners'-strategies-dependent function O; 3) other-negotiation-threads-dependent function P; and 4) competition-dependent function C.

Let $k_i^t[T]$, $k_i^t[O]$, $k_i^t[P]$, and $k_i^t[C]$ be the set of compromising degrees of the agent i in round t according to the decision functions T, O, P, and C, respectively. Letting k_i^t be the agent i's compromising degree in round t, we get

$$k_i^t = w_T k_i^t[T] + w_O k_i^t[O] + w_P k_i^t[P] + w_C k_i^t[C] \qquad (4.1)$$

where $0 \le w_T,\ w_O,\ w_P,\ w_C \le 1$, and $w_T + w_O + w_P + w_C = 1$. During negotiation, the agent can dynamically decide the values of w_T, w_O, w_P, and w_C.

If the agent i is a buyer, its proposal in round t is

$$b_i^t = \left(1 + k_i^t\right) \times b_i^{t-1}. \qquad (4.2)$$

If the agent i is a seller, its proposal in round t is

$$s_i^t = \left(1 - k_i^t\right) \times s_i^{t-1}. \qquad (4.3)$$

A. Time-Dependent Function T

Because a bargaining negotiation is fundamentally time dependent [8], [26], it is necessary to introduce a time variable for modeling market dynamics. Function T takes the remaining negotiation time's effect on negotiation strategies into account, i.e., agents' sensitivity to time. Agents' sensitivity to time embodies how the left negotiation time affects agents' negotiation strategies.

When an agent negotiates with several trading partners using the flexible mechanism, from its perspective, the negotiation process consists of several rounds, but each round may be of a different cycle time and the agent may only propose to some of its trading partners in each round. For the subnegotiation thread i, let B_i^t be the beginning time in round t and F_i^t be the finishing time in round t.[2] From the buyer's perspective when it negotiates with several sellers, the remaining time's effect on its negotiation strategy is given as follows:

$$b_i^t[T] = \left(b_{\max}^i - b_{\min}^i\right) \times \left(B_i^t / T_{\max}^i\right)^{\beta} + b_{\min}^i \qquad (4.4)$$

where $\beta > 0$ represents an agent's eagerness to complete a deal, $b_i^t[T]$ represents the subbuyer i's proposal in round t according to the decision function T, $T_{\max}$ represents the negotiation deadline of the buyer, $b_{\max}^t$ represents the buyer's reserve proposal, and $b_{\min}^t$ represents the buyer's desired proposal. The change of $b_i^t[T]$ with respect to the negotiation time has the following characteristics (as in Fig. 3).

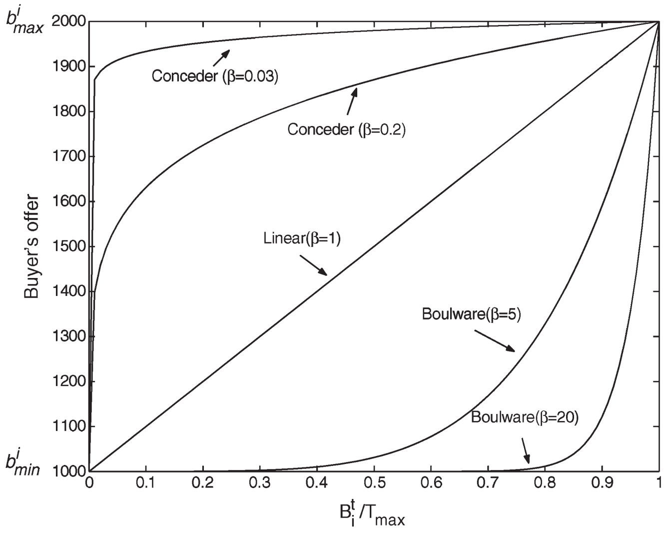

Fig. 3. Effect of the remaining time.

Case 1) $\beta = 1$. $b_i^t[T] = (b_{\max}^t - b_{\min}^t) \times B_i^t / T_{\max}^t + b_{\min}^t$. $b_i^t[T]$ linearly increases, which means that the remaining time has a consistent effect on the agent's negotiation strategy.

Case 2) $\beta > 1$. At the beginning of negotiation, $b_i^t[T]$ changes little. The agent makes little compromise at the beginning of negotiation, but makes large compromise when negotiation is to be closed.

Case 3) $0 < \beta < 1$. The change of the slope is decreasing. The agent is eager to reach an agreement as quickly as possible, and it makes large compromise at the beginning of negotiation. With the negotiation going on, the agent makes less and less compromise.

Similarly, when a seller i negotiates with several buyers, $s_i^t[T] = s_{\max}^i - (s_{\max}^i - s_{\min}^i) \times (B_i^t / T_{\max}^i)^{\beta}$ can be used to express the remaining negotiation time's effect on the seller's negotiation strategy, where $s_{\max}^i$ and $s_{\min}^i$ are the maximum and minimum proposal of the seller, respectively, and $s_i^t[T]$ represents the subseller i's proposal in round t according to the decision function T.

Let $\mathrm{kb}_i^t[T]$ be the compromising degree of the subbuyer i according to the function T in round t when the buyer negotiates with several sellers. Let $\mathrm{ks}_i^t[T]$ be the compromising degree of the subseller i according to the function T in round t when the seller negotiates with several buyers. Because $b_i^t[T] = (1 + \mathrm{kb}_i^t[T]) \times b_i^{t-1}[T]$ and $s_i^t[T] = (1 - \mathrm{ks}_i^t[T]) \times s_i^{t-1}[T]$, we have

$$\mathrm{kb}_i^t[T] = \frac{\left(b_{\max}^i - b_{\min}^i\right) \times \left(B_i^t / T_{\max}^i\right)^{\beta} + b_{\min}^i}{\left(b_{\max}^i - b_{\min}^i\right) \times \left(B_i^{t-1} / T_{\max}^i\right)^{\beta} + b_{\min}^i} - 1 \qquad (4.5\text{a})$$

$$\mathrm{ks}_i^t[T] = 1 - \frac{s_{\max}^i - \left(s_{\max}^i - s_{\min}^i\right) \times \left(B_i^t / T_{\max}^i\right)^{\beta}}{s_{\max}^i - \left(s_{\max}^i - s_{\min}^i\right) \times \left(B_i^{t-1} / T_{\max}^i\right)^{\beta}}. \qquad (4.5\text{b})$$

If shorter time is allowed, an agent faces greater pressure in making a final decision. Sim has analyzed different strategies that agents should adopt in response to different deadlines [23].

---

[2] For the general one-to-many negotiation mechanism, if the negotiation cycle of each round is c, we get $B_i^t = c(t-1)$ and $F_i^t = ct$.

For longer deadlines, an agent may find it advantageous to adopt a conservative strategy [case 2] because it has plenty of time for negotiating deals. A conciliatory strategy [case 3] may be more suitable if an agent is coerced into completing a deal rapidly. Regardless of the deadline, linear strategies [case 1] are more likely to make deals than conservative strategies while achieving higher utility than conciliatory strategies.

B. Trading-Partners'-Strategies-Dependent Function O

For negotiation agents, maximizing utility is the most important objective. Thus, an agent may choose to use imitative tactics to protect itself from being exploited by other agents. Moreover, making large compromise to a conservative agent makes no sense. In addition, an agent may have more than one chance to reach an agreement in one-to-many negotiation. Therefore, it is reasonable for an agent to make a compromise based on the behaviors of its trading partners.

Let $\mathrm{kb}_i^t[O]$ be the compromising degree of the subbuyer i according to function O in round t $(t > 2)$ when the buyer negotiates with several sellers. Let $\mathrm{ks}_i^t[O]$ be the compromising degree of the subseller i according to function O in round t $(t > 2)$ when the seller negotiates with several buyers. Thus, we have

$$\mathrm{kb}_i^t[O] = \eta^n \times \left(1 - s_i^{t-1}/s_i^{t-2}\right) \tag{4.6a}$$

$$\mathrm{ks}_i^t[O] = \eta^n \times \left(b_i^{t-1}/b_i^{t-2} - 1\right) \tag{4.6b}$$

where $0 < \eta \leq 1$ and n is the number of trading partners. From (4.6a) and (4.6b), we can find that an agent partly, in percentage of η^n, reproduces the behavior that its trading partner performed. The parameter η reflects agents' optimism toward negotiation results when their trading partners increase. η with a small value (for example, 0.5) means that an agent is very optimistic about negotiation results with the increase of trading partners. In contrast, η with a larger value (close to 1) means that the agent is not optimistic about negotiation results with the increase of trading partners.

Faratin *et al.* [3] have proposed three kinds of trading-partners'-behavior-dependent tactics. The distinguishing feature of our decision function is that we consider agents' multiple choices. With more trading partners, an agent will concede much less (as η^n decreases with the increase of the number n of trading partners). The trading partners' strategies-dependent function O acts as a control mechanism to prevent an agent from making excessive compromise to a conservative trading partner and inadequate compromise to a trading partner who is likely to reach an agreement.

C. Other-Negotiation-Threads-Dependent Function P

When the manager agent adopts the optimized strategy, all the negotiation threads have mutual influence. While making a decision on its compromising degree, in order to avoid unnecessary negotiation process, a single subagent should take the information of the other negotiation threads into account. For example, a buyer negotiates with three sellers. In a certain round, the subbuyer b_1 receives the seller s_1's proposal of \$50, and it knows that the proposals of the seller s_2 and the seller s_3 are \$30 and \$40, respectively, i.e., the lowest proposal of the three sellers is \$30. In order to avoid unnecessary negotiation processes, the subbuyer b_1's proposal to the seller s_1 in the next round will be lower than \$30. Similarly, both the subbuyer b_2's and the subbuyer b_3's proposals will also be lower than \$30.

In the flexible one-to-many negotiation mechanism, when a buyer negotiates with n sellers, each subbuyer will not immediately begin to calculate the counterproposal to its trading partner after receiving its trading partner's proposal, but it will wait to get the negotiation information of other negotiation threads according to the two strategies for deciding when to make a proposal (see Sections III-B and C). Let nums_{t-1} be the number of the trading partners to whom the buyer proposes in round t (in other words, the number of proposals that the buyer has received in round $t-1$), $0 < \mathrm{nums}_{t-1} \leq n$, and $s_{\min}^{t-1}$ represent the lowest proposal that the buyer has received in round $t-1$. Similarly, when a seller negotiates with n buyers, $0 < \mathrm{numb}_{t-1} \leq n$ represents the number of proposals that the seller has received in round $t-1$, and $b_{\max}^{t-1}$ represents the highest proposal that the seller has received in round $t-1$

$$s_{\min}^{t-1} = \min_{i=1}^{\mathrm{nums}_{t-1}} s_i^{t-1}$$

$$b_{\max}^{t-1} = \max_{i=1}^{\mathrm{numb}_{t-1}} b_i^{t-1}.$$

Suppose that a buyer negotiates with several sellers. In round $t-1$, a subbuyer i receives a proposal s_i^{t-1} from its partner—the seller i. For the subbuyer i, if $s_{\min}^{t-1} > b_i^{t-1}$, it should make some compromise. In this case, if $s_{\min}^{t-1}/s_i^{t-1}$ is very small, i.e., the seller i's offer s_i^{t-1} is very high as compared with the other $\mathrm{nums}_{t-1} - 1$ proposals, the subbuyer i would make little compromise to the seller i because their negotiation seems "hopeless." If $s_{\min}^{t-1}/s_i^{t-1}$ is close to 1, i.e., the seller i's proposal s_i^{t-1} is very "favorable" as compared with the other trading partners' proposals, the subbuyer i would make compromise to the seller i according to the decision function O because the subnegotiation thread seems "hopeful." Therefore, the lower the seller i's proposal as compared with the other $\mathrm{nums}_{t-1} - 1$ proposals, the more compromise the subbuyer i will make. Otherwise, i.e., $s_{\min}^{t-1} \leq b_i^{t-1}$, because at least one of the trading partners' proposals is higher than its proposal in round $t-1$, the subbuyer i will choose to raise its expectation, i.e., let $b_i^t < s_{\min}^{t-1} \leq b_i^{t-1}$. Because $b_i^t = (1 + \mathrm{kb}_i^t[P]) \times b_i^{t-1}$, where $\mathrm{kb}_i^t[P]$ is the compromising degree of the subbuyer i according to the decision function P in round t, then $(1 + \mathrm{kb}_i^t[P]) \times b_i^{t-1} < s_{\min}^{t-1}$, thus, $\mathrm{kb}_i^t[P] < s_{\min}^{t-1}/b_i^{t-1} - 1$. Hence, we have $\mathrm{kb}_i^t[P] = s_{\min}^{t-1}/b_i^{t-1} - 1 - \sigma$, $\sigma > 0$, and the subbuyer i can decide the value of σ according to its desire to maximize utility (greed). σ with a large value means that the agent will greatly raise its expectation if $s_{\min}^{t-1} \leq b_i^{t-1}$.

Similarly, let $\mathrm{ks}_i^t[P]$ be the compromising degree of the subseller i according to the decision function P in round t when the seller negotiates with several buyers. We have

$$\mathrm{kb}_i^t[P] = \begin{cases} \min\left(\mathrm{kb}_i^t[O], s_{\min}^{t-1}/s_i^{t-1}\right), & \text{if } s_{\min}^{t-1} > b_i^{t-1} \\ s_{\min}^{t-1}/b_i^{t-1} - 1 - \sigma, & \text{otherwise} \end{cases} \tag{4.7a}$$

$$\mathrm{ks}_i^t[P] = \begin{cases} \min\left(\mathrm{ks}_i^t[O], b_i^{t-1}/b_{\max}^{t-1}\right), & \text{if } b_{\max}^{t-1} < s_i^{t-1} \\ 1 - b_{\max}^{t-1}/s_i^{t-1} - \sigma, & \text{otherwise.} \end{cases} \tag{4.7b}$$

Li *et al.* [12] consider the other threads' influence on an agent's reserve price. However, in this paper, we consider the other threads' influence on an agent's negotiation strategies.

With the other-negotiation-threads-dependent function P, an agent is aware of the negotiation situations of the all negotiation threads and then brings forward new proposals, which helps to avoid unnecessary negotiation processes and to maximize agents' utilities (by making little compromise to conservative trading partners and raising agents' expectations).

D. Competition-Dependent Function C

A negotiator's bargaining "power" is affected by the number of competitors and trading alternatives. Good options give a negotiator more power because the negotiating party need not pursue the negotiation with any sense of desperation. For instance, in a buyer's market, supply is greater than demand, and buyers have a bargaining advantage. Conversely, in a seller's market, demand is greater than supply, and sellers have a bargaining advantage.

The competition situation of an agent is determined by the probability that it is (or is not) being considered as the most preferred trading partner [23], [26]. Suppose an agent b_1 has $m-1$ competitors $b_2, b_3, \ldots, b_m$ and n trading partners $s_1, s_2, \ldots, s_n$. The probability that b_1 is not the most preferred trading partner of any s_i $(i = 1, 2, \ldots, n)$ is $(m-1)/m$. The probability of agent b_1 not being the most preferred trading partner of all the trading partners is $[(m-1)/m]^n$. Hence, we have

$$\mathrm{kb}_i^t[C] = [(m-1)/m]^n \tag{4.8a}$$

$$\mathrm{ks}_i^t[C] = [(m-1)/m]^n \tag{4.8b}$$

where $\mathrm{kb}_i^t[C]$ is the compromising degree of the subbuyer i according to the decision function C in round t and $\mathrm{ks}_i^t[C]$ is the compromising degree of the subseller i according to the decision function C in round t. Because buyers and sellers can enter and leave the market at any time, the values m and n may constantly change with ongoing negotiation.

With respect to competition, a negotiation agent makes compromise according to the buyer–seller ratio in a market. In a favorable market, there are fewer competitors and more trading partners. Hence, an agent has stronger bargaining power and makes less compromise. In an unfavorable market, an agent's bargaining power decreases as it experiences more competition, and it may attempt to make more compromise. Using the competition-based function C, an agent strives to avoid making excessive compromise in favorable markets or inadequate compromise in unfavorable markets.

Although the four decision functions are implemented for one-to-many negotiation, most of these decision functions can be applied to one-to-one and many-to-many negotiations. All the four decision functions can be applied to many-to-many negotiation because one-to-many negotiation can be treated as a special kind of many-to-many negotiation. Time-dependent function T can be applied to one-to-one negotiation because the passage of time can affect an agent's negotiation strategy when it negotiates with another agent [10]. Trading-partners'-strategies-dependent function O can also be applied to one-to-one negotiation as it is still crucial to imitate trading partners' strategies in one-to-one negotiation. However, competition-dependent function C and other-negotiation-threads-dependent function P cannot be applied to one-to-one negotiations because there is no competition and there is only one negotiation thread.

TABLE I
INPUT DATA SOURCES

Input Data	**Possible Values**		
Market Type	*Favorable*	*Balanced*	*unfavorable*
P_{Buyer}	< 0.5	0.5	> 0.5
P_{Buyer}: *Probability* of an agent *being a buyer*			
Buyer-seller Ratio	{1:2, 1:4, 1:10}	1:1	{10:1, 4:1, 2:1}
Market Density	*Sparse*	*Moderate*	*Dense*
P_{gen}	0.25	0.5	1
P_{gen}: *Probability* of *generating an agent* per round			
Deadline	*Short*	*Moderate*	*Long*
T_{max}	$150-250s$	$250-400s$	$500-600s$
Eagerness	*Lower range*	*Mid-range*	*Upper range*
β	5	1	0.2
Optimism	*Lower range*	*Mid-range*	*Upper range*
η	0.98	0.93	0.8
Greed	*Lower range*	*Mid-range*	*Upper range*
σ	0.02	0.05	0.1

V. EVALUATION AND EXPERIMENTATION

A. Test Bed

To realize the idea of the flexible one-to-many negotiation mechanism, a simulation test bed consisting of a virtual e-Marketplace, which is a society of trading agents and a controller (manager), was implemented. The controller generates agents, randomly determines their parameters (e.g., their roles as buyers or sellers, initial proposals, reserve price, negotiation mechanisms, and deadlines), and simulates the entrance of agents to the virtual e-Marketplace. Using the test bed, a series of experiments was carried out in order to demonstrate the features of the flexible one-to-many negotiation mechanism and to evaluate the effectiveness of the negotiation strategies for subagents. In order to demonstrate the performance of the flexible one-to-many negotiation mechanism by comparison, another two mechanisms were evaluated as follows.

1) General One-to-Many Negotiation Mechanism: To evaluate the performance of the general mechanism in the flexible negotiation mechanism framework, we just need to let the waiting time $T_{\text{wait}} = \max(C) - \min(C)$ for the fixed-waiting-time-based strategy or the waiting ratio $R_{\text{wait}} = 1$ for the fixed-waiting-ratio-based strategy.

2) Desperate Strategy (Section II-B): Because all the subnegotiation threads are independent in the desperate strategy, the other-negotiation-threads-dependent function P is not suitable for agents using the desperate strategy, but agents can still use the other three decision functions. The general mechanism is utilized in negotiations with the desperate strategy.

To evaluate the performance of the three mechanisms in a wide variety of test environments, the agents are subject to different market densities, different market types, and other factors (e.g., deadline, eagerness, optimism, and greed) (Table I).

B. Experimental Settings

All the six input parameters in Table I are generated randomly following a uniform distribution. Both market density and market type depend on the probability of generating an agent in each round and the probability of the agent being a

TABLE II
PERFORMANCE MEASURE

Success Rate	$R_{success} = N_{success}/N_{total}$
Expected Utility	$U_{expected} = U_{success} \times R_{success} + U_{fail} \times (1 - R_{success}) = U_{success} \times R_{success}$
Average Negotiation Time	$R_{time} = \sum_{i=1}^{N_{total}} (T_{end}^{i}/T_{max}^{i})/N_{total}$
N_{total}	*Total number of agents*
$N_{success}$	*No. of agents that reached consensus*
$U_{success}$	*Average utility of agents that reached consensus*
$U_{fail} = 0$	*Average utility of agents that didn't reach consensus*
T_{end}^{i}	*The time spent in negotiation by the agent i*
T_{max}^{i}	*The deadline of the agent i*

buyer (or a seller). From a buyer agent's perspective, for a favorable (respectively, an unfavorable) market, an agent enters a market with lower (respectively, higher) probability of being a buyer agent and higher (respectively, lower) probability of being a seller. The life span of an agent, i.e., its deadline, is randomly selected from a range of 150–600 s. The range of 150–600 s for the deadline is adopted based on experimental tuning and the agents' behaviors. In the current experimental setting, the reaction time of every agent is about 5 s following a uniform distribution for convenience; it was found that for a very short deadline (< 150 s), very few agents could complete deals, and for deadlines > 600 s, there was little or no difference in the performance of agents. Hence, for the purpose of experimentation, a deadline between the ranges of 150–250, 250–400, and 450–600 s are considered short, moderate, and long, respectively. The value of eagerness is randomly generated from [0.1, 10], and the range values for eagerness follow the definition in Section IV-A. It was found that when $\beta > 10$ and $\beta < 0.1$, agents made little and large compromise, respectively, at the beginning of negotiation and there was little or no difference in the performance of agents. Hence, representative values of eagerness from the lower range (e.g., 5), the upper range (e.g., 0.2), and the midrange (e.g., 1) were used. The value of optimism is randomly generated from [0.7, 1]. Through experimentation, it was found that when the optimism value $\eta > 0.99$ or $\eta < 0.7$, there was little or no difference in the performance of negotiation agents. Thus, optimism values of 0.8, 0.98, and 0.93 are considered as upper range, lower range, and midrange, respectively. The value of greed is selected from the range of 0.01–0.15 based on experimental tuning. It was found that when the greed value $\sigma > 0.15$, most agents failed to reach agreements, and when the greed value $\sigma < 0.01$, there was little or no difference in the performance of negotiation agents. Therefore, greed values of 0.1, 0.02, and 0.05 are considered as upper range, lower range, and midrange, respectively.

C. Performance Measurement

The performance measurements include expected utility, success rate, and average negotiation time (Table II). Other than optimizing agents' utility, enhancing the success rate is also an important evaluation criteria for designing negotiation agents, as pointed out in [5] and [15, p.130]. Because negotiation results are uncertain (i.e., there are two possibilities, namely: 1) eventually reaching a consensus and 2) not reaching a consensus), it seems more prudent to use expected utility [2] rather than average utility as a performance measure because it takes into consideration the probability distribution over the two different outcomes [27]. Average negotiation time examines the average amount of time spent in negotiation.

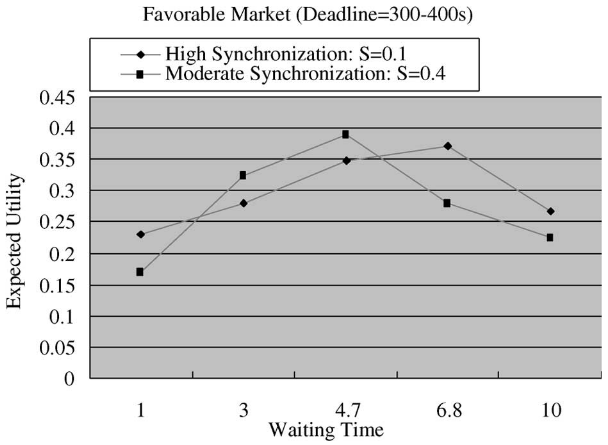

Fig. 4. Fixed-waiting-time-based strategy.

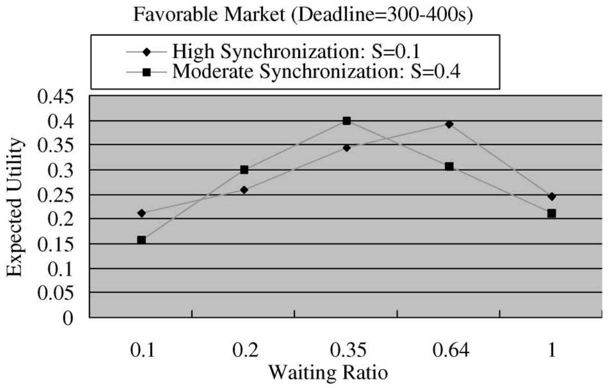

Fig. 5. Fixed-waiting-ratio-based strategy.

D. Results

An extensive amount of stochastic simulations were carried out for all the combinations of market density (dense, moderately dense, and sparse), market type (favorable, almost balanced, and unfavorable), and other agents' constraints (deadline, eagerness, optimism, and greed). Even though experiments were carried out for all 729 $(3 \times 3 \times 3 \times 3 \times 3 \times 3)$ combinations of the input data, due to space limitation and the main objective of the experimental simulation, which is to show the advantages of the flexible mechanism over the other two mechanisms, only some representative results are presented in Figs. 4–11. In these experiments, the market is of moderate density, the value of eagerness is 0.8, the optimism value is 0.8, and the greed value is 0.1.

E. Observation 1

The negotiation outcomes with the waiting time (respectively, waiting ratio) given by the fixed-waiting-time-based strategy (respectively, fixed-waiting-ratio-based strategy) are more desirable than the outcomes with any other waiting time (respectively, waiting ratio).

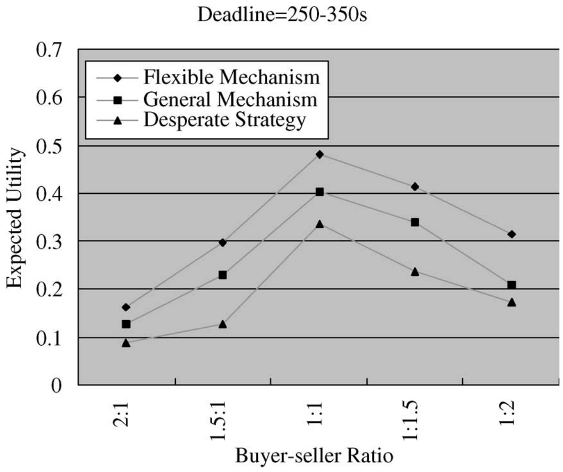

Fig. 6. Expected utility and market type.

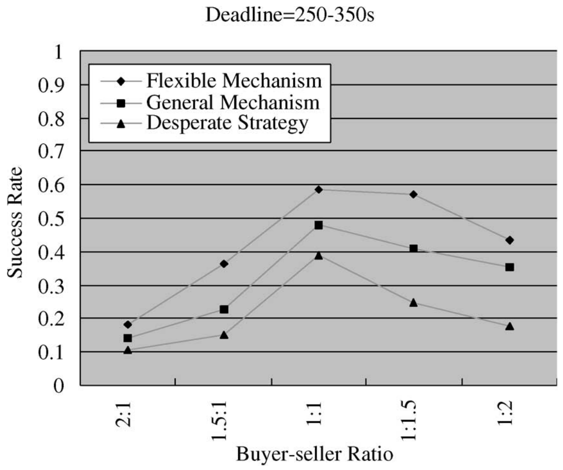

Fig. 7. Success rate and market type.

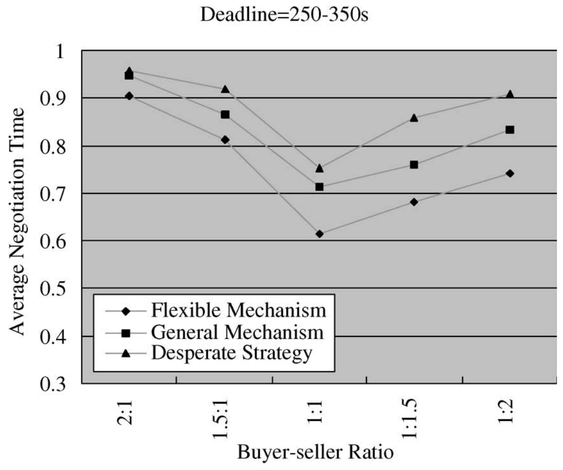

Fig. 8. Average negotiation time and market type.

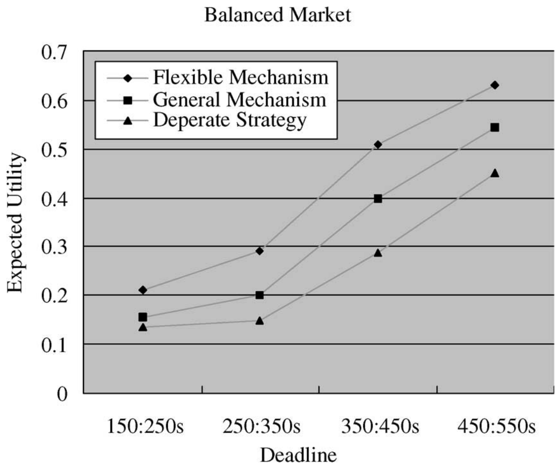

Fig. 9. Expected utility and negotiation deadline.

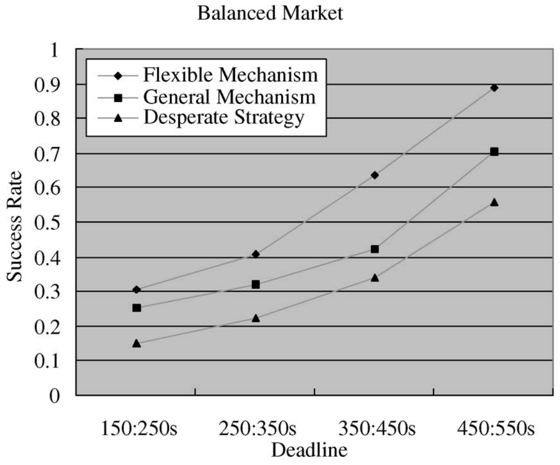

Fig. 10. Success rate and negotiation deadline.

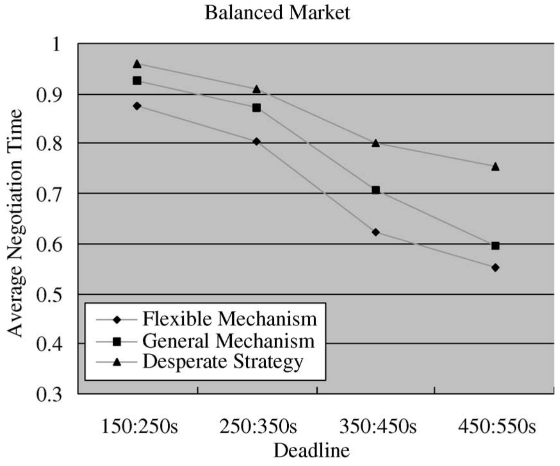

Fig. 11. Average negotiation time and negotiation deadline.

In Figs. 4 and 5, it can be found that the market is favorable for buyers, and the deadline is between 300 and 400 s. A set of experiments with different synchronization levels (e.g., $S_1 = 0.1$ and $S_2 = 0.4$) was conducted. In Fig. 4, $\max(C) - \min(C) = 10$ s. According to (3.2) (Section III-B), the waiting times for the two different synchronization levels are 6.8 and 4.7 s, respectively. Results from simulations show that the negotiation results with the waiting time given by (3.2) are more desirable than the result with any other waiting time. In Fig. 4, when $S_1 = 0.1$ and using the waiting time 6.8 s given by (3.2), the expected utility is 0.37, which is higher than the outcome with any other waiting time (e.g., waiting time = 1, 3, 4.7, and 10 s); when $S_2 = 0.4$, the waiting time given by (3.2) is 4.7 s, and the expected utility is 0.39, which is also higher than the outcome when the waiting time is 1, 3, 6.8, or 10 s. Simulation results show that the negotiation results with the waiting ratio given by (3.3) are also more desirable than the result with any other waiting ratio. According to (3.3) (Section III-C), the fixed waiting ratios for the two different synchronization levels are 0.64 and 0.35, respectively. In Fig. 5, when $S_1 = 0.1$ and using the waiting ratio 0.64 given by (3.3), the expected utility is 0.39, which is higher than the outcome with any other waiting ratio; when $S_2 = 0.4$, the waiting ratio given by (3.3) is 0.35, and the expected utility is 0.40, which is also higher than the outcome with any other waiting ratio.

F. Observation 2

With moderate deadline (e.g., 250–350 s), the flexible mechanism achieved higher U_{expected}, higher R_{success}, and lower R_{time} than the general mechanism and the desperate strategy.

The experimental results in Fig. 6 show that the flexible mechanism always gets better negotiation results than the general mechanism and the desperate strategy, and the general mechanism always gets better negotiation results than the desperate strategy. In Fig. 6, when the buyer–seller ratio is $1:1$, the expected utilities for the flexible mechanism, general mechanism, and desperate strategy are 0.49, 0.40, and 0.34, respectively.

The simulation results in Fig. 7 suggest that the flexible mechanism always gets higher success rates than the general mechanism and the desperate strategy with different buyer–seller ratios. In Fig. 7, the success rates for the flexible mechanism, general mechanism, and desperate strategy are 0.59, 0.48, and 0.39, respectively, when the buyer–seller ratio is $1:1$.

From the experimental results in Fig. 8, we can find that the negotiation with the flexible mechanism always leads to a shorter average negotiation time as compared with the general mechanism and the desperate strategy. For example, when the buyer–seller ratio is $1:1$, the average negotiation times for the flexible mechanism, general mechanism, and desperate strategy are 0.61, 0.71, and 0.75, respectively.

G. Observation 3

In an almost balanced market, the flexible mechanism achieved higher U_{expected}, higher R_{success}, and lower R_{time} than the general mechanism and the desperate strategy.

The experimental results in Fig. 9 show that the negotiation results become more favorable with the increase of the deadline in the three situations. For example, when agents utilize the flexible mechanism, the expected utilities are 0.21, 0.29, 0.51, and 0.63, respectively, with the deadline varying from 150 to 550 s. Furthermore, the flexible mechanism always gets better negotiation results than the general mechanism and the desperate strategy with the same deadline.

With the increase of negotiation deadline, agents tend to make relatively less compromise and will have more time to bargain. The simulation outcomes in Fig. 10 show that the success rate increases with the increase of the deadline in the three situations. We can also find that the flexible mechanism always gets higher success rates than the general mechanism and the desperate strategy with the same deadline.

The simulation results in Fig. 11 show that the average negotiation time decreases with the increase of the deadline in the three situations. The results correspond to the intuition that an agent with a short (respectively, long) deadline has to spend almost all (respectively, a part of) the permitted time to reach an agreement. Taking the flexible mechanism as an example, the average negotiation times when the deadline is in the ranges of 150–250, 250–350, 350–450, and 450–550 s are 0.88, 0.80, 0.62, and 0.55, respectively. Furthermore, the flexible mechanism always leads to a shorter average negotiation time than that of the general mechanism and the desperate strategy.

VI. RELATED WORK

The literature of one-to-many negotiation mechanisms and systems (e.g., [1], [7], [20], and [30]) and negotiation strategies for agents conducting one-to-many negotiation (e.g., [3], [8], [23], [24], [26], and [27]) forms a very huge collection, and space limitations preclude introducing all of them here. For a survey on automated negotiation, see [5] and [14]. This section only introduces and discusses some important negotiation systems and negotiation strategies supporting one-to-many negotiation.

An intelligent trading agency (ITA) [7], [20] is a framework for one-to-many negotiation by means of conducting a number of concurrent coordinated one-to-one negotiations. In the framework, a number of agents, all working on behalf of one party, negotiate individually with other parties. After each negotiation cycle, these agents report back to a coordinating agent that evaluates how well each agent has done and issues new instructions accordingly. Each individual agent conducts reasoning by using constraint-based techniques. Negotiation strategies of individual subnegotiators include take it or leave it, no compromise, fixed compromise, and better deal strategies.

Negotiating agents for load management (NALM) [1] is a component-based multiagent system capable of negotiation for load management. In NALM, a model has been designed with a transparent component-based structure based on explicit and formal specifications of the knowledge of the negotiation process at a conceptual level. In NALM, the utility agent always starts the negotiation process as soon as a peak in the electricity consumption is predicted. The utility agent communicates an announcement to all customer agents to which they can respond by making a bid. The utility agent may then put forward another announcement depending on the bids the customer agents have made, and this goes on until an agreement is finally established.

Vetter and Pitsch [30] have presented competitive agents for secure business applications (CASBA), a project devoted to developing an electronic marketplace to improve the quality of existing electronic commerce services by introducing higher flexibility and automating trading processes. Technically, the CASBA system consists of a server and three applets. The characteristics of CASBA are given as follows: 1) negotiation protocol described in CASBA is one of alternating proposals; 2) agents' roles are defined in advance, and the agents are assumed to be able to evaluate a proposal by computing its value in terms of a given (private) utility function; 3) environment is static; and 4) no timeouts or requests for information are present, and bids are private.

Nguyen and Jennings [16], [17] have developed and evaluated a heuristic model that enables an agent to participate in multiple concurrent bilateral encounters in competitive situations in which there are information uncertainty and deadlines. The main findings through empirical evaluation are given as follows: 1) time to complete the negotiation is less for the concurrent model than for the sequential one; 2) to realize the benefits of concurrent negotiation, the buyer agent's deadline must not be too short; 3) final agreements reached by the concurrent model have, on average, higher or equal utility for the buyer than those of the sequential model; 4) changing the strategy in response to the agent's assessment of the ongoing negotiation is equal or better than not doing so; 5) to improve the performance of the concurrent model, the analysis time should be moderately early but not too early; and 6) the tougher the buyer negotiates, the better the overall outcome it obtains.

Li *et al.* [12], [13] present a model for bilateral contract negotiation that considers the uncertain and dynamic outside options. Outside options affect the negotiation strategies via their impact on the reserve price. The model is composed of three modules, namely: 1) single-threaded negotiation; 2) synchronized multithreaded negotiation; and 3) dynamic multithreaded negotiation. The single-threaded negotiation model provides negotiation strategies without specifically considering outside options. The model of synchronized multithreaded negotiation builds on the single-threaded negotiation model and considers the presence of concurrently existing outside options. The model of dynamic multithreaded negotiation expands the synchronized multithreaded model by considering the uncertain outside options that may come dynamically in the future.

Unlike some one-to-many negotiation systems (e.g., ITA [7], [20], NALM [1], and CASBA [30]), where one-to-many negotiation is conducted in discrete time, this paper presents a continuous-time one-to-many negotiation mechanism. In the continuous negotiation mechanism, an agent can decide when to propose according to the synchronization situations of negotiation. In Nguyen and Jennings' concurrent one-to-many negotiation model [16], [17], the multiple negotiation threads still have the same negotiation cycle. However, this paper focuses on proposing a flexible mechanism. Moreover, the heuristic method used by the coordinator of negotiation in [16] and [17] strongly depends on history information about trading partners and negotiation environment. This paper also presents several decision functions for agents' proposal generation. In the discrete-time concurrent one-to-many negotiation model by Li *et al.* [12], [13], the multiple threads mutually influence one another, and the impact of the other threads' influence on an agent's reserve price is considered. In this paper, we also consider the other subnegotiation threads' influence on an agent's negotiation strategies. Moreover, their model strongly depends on history information, e.g., the probability distribution of partners' reserve price.

VII. CONCLUSION

This paper investigates a flexible continuous-time one-to-many negotiation mechanism for software agents. The main contributions of this research include: 1) flexible one-to-many negotiation mechanism; 2) two strategies for deciding when to make a proposal; 3) four decision functions supporting agents' proposal generation; and 4) evaluation of the flexible one-to-many negotiation mechanism through experimentation. Compared with general one-to-many negotiation mechanisms including various auction mechanisms (e.g., [7], [12], [16], and [20]), the flexible mechanism can conduct one-to-many negotiations in a more flexible and interactive way, e.g., an agent with flexible mechanism does not have to wait until it has received proposals from all the trading partners before generating proposals; an agent can take different negotiation strategies for different trading partners. The two strategies for deciding when to make a proposal are based on the practical synchronization situations, which help to get more favorable negotiation results as validated in the simulation experiments (see Section V-E). After an analysis of the special characteristics of one-to-many negotiations and the crucial factors affecting negotiation performance, the four decision functions based on time, trading partners' strategies, negotiation situations of other negotiation threads, and competition are given. However, this research makes no claim that our strategies are sufficiently accurate or powerful to solve all or most of the problems in one-to-many negotiation. When agents with our negotiation strategies are engineered with some elements that model how negotiators in reality may behave, it is not the intention of this research to create an exact replica of negotiators in realistic markets. Because market situations in reality are very complex, the factors considered in this research highlight some of the most essential and fundamental elements that affect the negotiators' decision making in one-to-many negotiation.

The advantages of the flexible mechanism over the general mechanism are summarized as follows.

1) A flexible mechanism allows an agent to decide when to make a proposal based on the synchronization situations of negotiation. This research introduces two strategies used for deciding when to make a proposal. The simulation results show that the waiting time (or waiting ratio) given by the two strategies leads to the most favorable outcomes (see Section V-E). Furthermore, the flexible mechanism can be easily transformed to the general mechanism (see Section III-B and C). As agents can propose in a flexible way, our flexible mechanism is especially useful for one-to-many negotiation between software agents that have different reaction times.
2) A flexible mechanism always leads to more favorable negotiation outcomes as compared with the general mechanism. The simulation outcomes in Section V-F and G suggest that the flexible mechanism always gets results of higher utility (respectively, higher success rate and shorter average negotiation time) with different market types and deadlines. Experimental results also show that the four decision making functions help with agents' reacting to changing market situations by making prudent and appropriate rates of concession and achieving favorable negotiation outcomes.

In summary, in conducting one-to-many negotiation in a more flexible way, the proposed flexible one-to-many negotiation mechanism can be applied in open dynamic complicated negotiation environments (such as a service-oriented grid, supply chain, and workflow). Because most multiagent systems work in a heterogeneous and distributed environment, our strategies and mechanism still need to be validated in practical applications. Finally, a future agenda of this paper is engineering behavior-based tactics [3] and learning techniques [31] into agents conducting one-to-many negotiation.

ACKNOWLEDGMENT

The authors would like to thank the associate editor and three anonymous referees for their comments and suggestions for improving this paper.

REFERENCES

[1] F. M. T. Brazier, F. Cornelissen, R. Gustavsson, C. M. Jonker, O. Lindeberg, B. Polak, and J. Treur, "A multi-agent system performing one-to-many negotiation for load balancing of electricity use," *Electron. Commerce Res. Appl.*, vol. 1, no. 2, pp. 208–224, 2002.

[2] P. K. Dutta, *Strategies and Games: Theory and Practice*. Cambridge, MA: MIT Press, 1999.

[3] P. Faratin, C. Sierra, and N. R. Jennings, "Negotiation decision functions for autonomous agents," *Int. J. Robot. Auton. Syst.*, vol. 24, no. 3/4, pp. 159–182, 1998.

[4] R. H. Guttman, A. G. Moukas, and P. Maes, "Agent-mediated electronic commerce: A survey," *Knowl. Eng. Rev.*, vol. 13, no. 2, pp. 143–152, Jun. 1998.

[5] N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, C. Sierra, and M. Wooldridge, "Automated negotiation: Prospects, methods and challenges," *Int. J. Group Decis. Negot.*, vol. 10, no. 2, pp. 199–215, 2001.

[6] J. B. Kim and A. Segev, "A framework for dynamic ebusiness negotiation processes," in *Proc. IEEE Int. Conf. E-Commerce*, Jan. 2003, pp. 24–27.

[7] R. Kowalczyk and V. Bui, "On constraint-based reasoning in negotiation agents," in *Proc. Agent-Mediated E-Commerce III*, vol. 2003, *Lecture Notes in Computer Science*, 2003, pp. 31–46.

[8] S. Kraus, "Automated negotiation and decision making in multi-agent environments," in *Multi-Agents Syst. Appl.*, vol. 2086, Lecture Notes in Computer Science. Berlin, Germany: Springer-Verlag, 2001.

[9] ——, *Strategic Negotiation in Multiagent Environments*. Cambridge, MA: MIT Press, 2001.

[10] S. Kraus, J. Wilkenfeld, and G. Zlotkin, "Multiagent negotiation under time constraints," *Artif. Intell.*, vol. 75, no. 2, pp. 297–345, Jun. 1995.

[11] G. Lai, C. Li, K. Sycara, and J. Giampapa, "Literature review on multi-attribute negotiations," Robotics Inst., Carnegie Mellon Univ., Pittsburgh, PA, Tech. Rep. CMU-RI-TR-04-66, Dec. 2004.

[12] C. Li, J. Giampapa, and K. Sycara, "Bilateral negotiation decisions with uncertain dynamic outside options," in *Proc. 1st IEEE Int. Workshop Electron. Contract.*, 2004, pp. 54–61.

[13] ——, "Bilateral negotiation decisions with uncertain dynamic outside options," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 36, no. 1, pp. 31–44, Jan. 2006.

[14] A. R. Lomuscio, M. Wooldridge, and N. R. Jennings, "A classification scheme for negotiation in electronic commerce," *Int. J. Group Decis. Negot.*, vol. 12, no. 1, pp. 31–56, 2003.

[15] M. Wooldridge, *Introduction to Multiagent Systems*. New York: Wiley, 2002.

[16] T. D. Nguyen and N. R. Jennings, "Concurrent bi-lateral negotiation in agent systems," in *Proc. 4th DEXA Workshop e-Negotiations*, 2003, pp. 839–844.

[17] ——, "A heuristic model of concurrent bi-lateral negotiations in incomplete information settings," in *Proc. 18th Int. Joint Conf. Artif. Intell.*, 2003, pp. 1467–1469.

[18] D. C. Parkes, L. H. Ungar, and D. Foster, "Accounting for cognitive costs in on-line auction design," in *Proc. Agent Mediated Electronic Commerce*, 1999, vol. 1571, pp. 25–40.

[19] D. Pruitt, *Negotiation Behavior*. New York: Academic, 1981.

[20] I. Rahwan, R. Kowalczyk, and H. H. Pham, "Intelligent agents for automated one-to-many e-commerce negotiation," *Aust. Comput. Sci. Commun.*, vol. 24, no. 1, pp. 197–204, Jan./Feb. 2002.

[21] J. Rosenschein and G. Zlotkin, *Rules of Encounter: Designing Conventions for Automated Negotiation Among Computers*. Cambridge, MA: MIT Press, 1994.

[22] A. Rubinstein, "Perfect equilibrium in a bargaining model," *Econometrica*, vol. 50, no. 1, pp. 97–109.

[23] K. M. Sim, "A market-driven model for designing negotiation agents," *Comput. Intell.*, vol. 18, no. 4, pp. 618–637, 2002.

[24] ——, "Negotiation agents that make prudent compromises and are slightly flexible in reaching consensus," *Comput. Intell.*, vol. 20, no. 4, pp. 643–662, 2004.

[25] ——, "Equilibria, prudent compromises, and the 'waiting' game," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 33, no. 3, pp. 712–724, Jun. 2005.

[26] K. M. Sim and C. Y. Choi, "Agents that react to changing market situations," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 33, no. 2, pp. 188–201, Apr. 2003.

[27] K. M. Sim and S. Wang, "Flexible negotiation agent with relaxed decision rules," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 3, pp. 1602–1608, Jun. 2004.

[28] K. M. Sim and E. Wong, "Towards market-driven agents for electronic auction," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 31, no. 6, pp. 474–484, Nov. 2001.

[29] K. Sycara, "Multi-agent systems," *Artif. Intell.*, vol. 19, no. 2, pp. 79–92, 1998.

[30] M. Vetter and S. Pitsch, "Towards a flexible trading process over the internet," in *Proc. Agentlink*, vol. 1991, *Lecture Notes in Computer Science*, 2001, pp. 148–162.

[31] D. Zeng and K. Sycara, "Bayesian learning in negotiation," *Int. J. Human Comput. Syst.*, vol. 48, no. 3/4, pp. 125–141, 1998.

Bo An received the B.Sc. degree in computer science from Chongqing University, Chongqing, China, in 2003. He is currently working toward the M.Sc. degree in computer science at the College of Computer Science at Chongqing University.

His research interests are autonomous agent and multiagent systems, especially automated negotiation and coalition formation.

Kwang Mong Sim received the B.Sc. degree (summa cum laude) from the University of Ottawa, Ottawa, ON, Canada, and the M.Sc. and Ph.D. degrees from the University of Calgary, Calgary, AB, Canada.

He is currently an Associate Professor in the Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, and was an Associate Professor at the Chinese University of Hong Kong. He served as a Session Chairman and PC Member in many agent conferences and supervised over 20 researchers/graduate students in this area.

Dr. Sim is an Associate Editor for the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C and a Guest Editor for several special journal issues on automated negotiation and agent-based grid computing, including the special issue on *Game Theoretic Analysis and Stochastic Simulation of Negotiation Agents* in the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C.

Liang Gui Tang received the B.Sc. degree in mathematics and computer science from Chongqing Normal University, Chongqing, China, in 1998, and the M.Sc. degree in computer science from Chongqing University, Chongqing, in 2003. He is currently working toward the Ph.D. degree in computer science at Chongqing University.

His research interests include intelligent e-commerce.

Shuang Qing Li received the B.Sc. degree in mathematics and computer science from the National University of Defense Technology, Hunan, China, in 1990, the M.Sc. degree in computer science from the University of Electronic Science and Technology of China, Chengdu, in 1994, and the Ph.D. degree in computer science from Chongqing University, Chongqing, China, in 2003.

He is currently an Associate Professor in the College of Computer Science, Chongqing University. His research interests are intelligent e-commerce and networks.

Dai Jie Cheng received the B.Sc. degree in mathematics from the Beijing University of Aeronautics and Astronautics, Beijing, China, in 1958.

He is currently a Professor in the College of Computer Science, Chongqing University, Chongqing, China. His research interests are parallel computing, networks, intelligent e-commerce, and data mining.