Time-bound negotiation framework for electronic commerce agents

Kyoung Jun Lee [a,*], Yong Sik Chang [b], Jae Kyu Lee [b]

[a] *School of Business, Korea University, 1,5Ka Anam-dong, Sungbuk-ku, Seoul 305-701, South Korea*
[b] *Graduate School of Management, Korea Advanced Institute of Science and Technology, Seoul, South Korea*

Abstract

For efficient and informative coordination of agents especially in electronic commerce environment, a time-bound agent negotiation framework is proposed utilizing a time-based commitment scheme. By attaching commitment duration to agent messages, the traditional contract net protocol is extended to a time-bound negotiation framework (TBNF). The proposed negotiation framework has a new message type which allows for parties to agree upon the extension of a commitment duration, and a novel commitment concept in the form of negative commitment. The semantics of the messages with the commitment duration are interpreted, and then the three typical negotiation protocols are formally defined and compared — *nothing-guaranteed protocol, acceptance-guaranteed protocol*, and *finite-time guarantee protocol* — which can be incorporated into TBNF. The TBNF should provide a background for efficient and effective electronic commerce negotiation while accommodating each agent's adaptive negotiation strategy. © 2000 Elsevier Science B.V. All rights reserved.

1. Introduction

As electronic commerce becomes more popular, the role of automated negotiation systems is expected to increase. For example, when a virtual shopping mall receives product orders from a cus-

tomer, it is necessary to make the delivery orders automatically without human intervention, generate a request for proposal (RFP) and announce it to multiple delivery companies (as shown in Fig. 1).

Then, the mall and delivery companies will negotiate over the price and quality (e.g., delivery date) of a specific delivery service. In this case, each agent is self-interested, the contractor agent (i.e., delivery company) is resource-constrained, and the status of agents constantly changes.

In such an electronic commerce environment, the efficient coordination of multi-agents is very important for the performance of each agent and the whole system. Contract Net Protocol [14] has been the most

---
* Corresponding author. Tel.: +82-2-3290-1952; fax: +82-2-922-4073.
E-mail addresses: leekj@kuba.korea.ac.kr (K.J. Lee), yschang@msd.kaist.ac.kr (Y.S. Chang), jklee@msd.kaist.ac.kr (J.K. Lee).
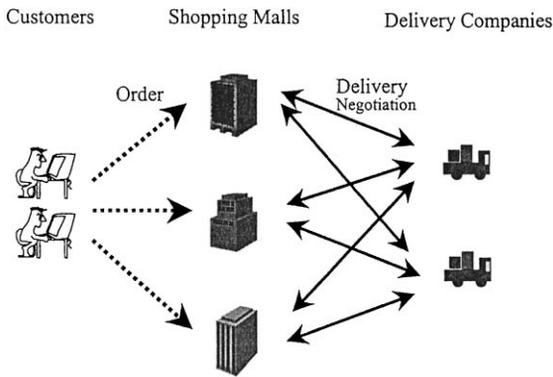
Fig. 1. Dynamic delivery negotiation in virtual shopping mall context.

commonly used method for coordinating agents in negotiation. Contract Net Protocol specifies communication and controls in a distributed problem solver [3]. This communication protocol concerns how contract managers announce the tasks to other agents, how potential contractors return bids to the manager, and how the manager then awards the contract (a manager is responsible for monitoring the execution of a task and processing the results of its execution, and a contractor is responsible for the actual execution of the tasks). The basic steps of the protocol are listed below and shown in Fig. 2.

- A manager issues a task announcement describing the task and the criteria for bids.
- Contractors send bids announcing their willingness and ability to perform a task.
- The manager sends the award message to a successful contractor.
- The contractor sends an acknowledgement of the award back to the manager, either accepting or rejecting it.

The original Contract Net Protocol can be somewhat simple for a specific purpose. However, it needs to be modified to satisfy various system requirements for improving performance. Therefore, there has been research conducted on the extension of the protocol by:

- Introducing new speech acts such as temporal grant, temporal reject, definitive grant, and defini-

tive reject when tasks exceed the capacity of a single agent. The context involves an extended Contract Net Protocol [5].
- Introducing fuzzy theoretic techniques for determining next task announcement and the best bid [17].
- Using directed contracts and forgetting by case-based reasoning to reduce communication loads [15].
- Enabling agents to choose levels of commitments dynamically for iterative task allocation negotiations [10].

In this paper, we propose a negotiation framework emphasizing the commitment duration attached to every message (commitment is an agreement or pledge to do something in the future [7,12]). The time-bound framework provides a good background for choosing a proper negotiation protocol especially for a dynamic situation where desired tasks and available resources may be changing as the system is executing tasks. The following is a scenario for a delivery problem in a virtual shopping mall, which inspires this research.

1.1. A Delivery Problem in a Virtual Shopping Mall

1. A shopping mall agent (SMA1) asks a delivery company agent (DCA) if a product PA can be delivered to its buyer from a warehouse in 3 days.
2. DCA schedules its own facility (e.g., trucks) for the delivery of the product PA and replies 'Yes' to SMA1.
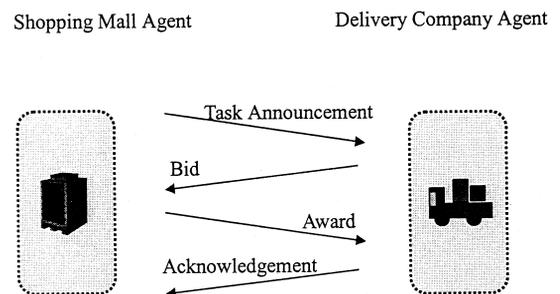3. However, SMA1 has not yet awarded the bid to DCA.

Fig. 2. Message flows between a manager and a contractor.

4. In the meantime, another shopping mall agent, SMA2 asks the (DCA) whether DCA can deliver a product PB to its buyer from a warehouse in 3 days.
5. DCA tries to schedule its own facility for the delivery of PB and then learns that the product PB can not be delivered to the buyer for SMA2 on time without canceling the capacity reservation for SMA1.

In this case, what should DCA do? Three alternative protocols can be used to cope with these problems: (1) *Nothing-guaranteed protocol*, (NGP) (2) *Acceptance-guaranteed protocol*, (AGP) and (3) *Finite-time guarantee protocol* (FGP). The next section briefly explains each of them.

2. Three possible protocols for dealing with the scenario

2.1. Nothing-guaranteed protocol

Most protocols based on traditional Contract Net Protocol use nothing-guaranteed protocol [8,9,13,16]. In this protocol, no one has to take any responsibilities before reaching mutual agreement on completing tasks. In the scenario, DCA feels free to reply 'Yes' to SMA2, and waits for the confirmation messages from either SMA1 or SMA2. Under this protocol, the messages need not have any time concept because there is no commitment or responsibility before the final agreements. Therefore, each agent feels free to send a message to another agent. However, since nothing is guaranteed until an agreement is made, some agents feel nervous and the global coordination performance is degraded. For example, SMA1 receives the bid from DCA and decides to award the bid to DCA. In the meantime, DCA can contract with SMA2, then DCA cannot accept the bid it received from SMA1. In this case, SMA1 should start negotiations again to find a new partner.

Traditional Contract Net Protocol is a protocol which does not guarantee anything before the final agreements. Even though the manager awards the bid immediately after receiving it, the contractor's rejection is inherently possible, which can result in an inefficient negotiation process. In addition, each agent is potentially unsettled because there is no further information until the final agreements.

2.2. Acceptance-guaranteed protocol

In this protocol, an agent guarantees that he will fulfill the order when he initially responds to the other agent's message. For example, a manager agent can guarantee the acceptance of a bid submission for the contractor agent when the manager agent announces a task, which means that the task announcer automatically awards the bid to the bidder on condition that the submitted bid satisfies the constraints of the initial announcement. In addition, a contractor agent can guarantee unconditional acceptance of the bid awarded from a manager agent. This protocol is useful when a task announcer agent contacts only one bidder agent or a bidder agent contacts only one manager. This situation can be interpreted as an unusual case, called 'audience restriction' discussed in Ref. [16]. For example, there might be only one bidder, or the announcer could have preferences among the bidders. One of the merits of the acceptance-guaranteed protocol is the reduction of required communication between the agents. This protocol can be used for hierarchical (vertical) coordination between a high-level agent and lower-level agent, where the high-level agent has only one partner in negotiation. If we apply this protocol to the above scenario, DCA guarantees the acceptance of the bid awarded from the SMA1, so DCA should reply 'no' to SMA2 because DCA already submitted the bid to SMA1.

2.3. Finite-time guarantee protocol

When a message has a certain type of guarantee, i.e., commitment, then the message needs to have a specified lifetime because an agent cannot wait a long time to establish a contract with the other agent. In addition, if an agent wants to hold a task but needs more time to confirm that it can fill the order, it should send a message requesting an extension of the life of the message. Finite-time guarantee protocol addresses this by giving every message its own time-token over a valid duration. The message, then, is valid for the duration specified in the token. The time-token can be attached to messages making re-

quests as well as messages replying to requests. We may rewrite the above scenario using the protocol below.

1. SMA1 asks DCA if a product PA can be delivered to its buyer in 3 days.
2. DCA schedules its own facilities for the delivery of the product PA and replies 'Yes' to SMA1 with a time-token valid for 30 min.
3. SMA1 has not awarded the bid to DCA yet.
4. In the meantime, another shopping mall agent SMA2 asks DCA if the product PB can be delivered to its buyer in 3 days.
5. DCA checks the current time.

(a) If 30 min have passed, DCA schedules its own facility for the delivery of PB without reserving its resources for SMA1 and replies 'Yes' to SMA2 with the time-token valid for 30 min. DCA has no obligation with SMA1.

(b) If it is before the deadline (for example, 20 min has passed since replying to SMA1), DCA should try to schedule its own facility for the delivery of PB by reserving capacity for SMA1. As a result, DCA knows that it cannot deliver PA for SMA2 on time without canceling the capacity reservation for SMA1. DCA replies 'No' to SMA2 but it asks SMA2 to make the request again after 10 min, or wait for 10 min to confirm the message.

3. Time-bound negotiation framework (TBNF)

To synthesize the above three protocols into a framework, we propose the TBNF which is a meta-model of the Contract Net Protocol emphasizing commitment duration attached to each negotiation's message.

3.1. Definition of TBNF

TBNF is a negotiation framework where agents negotiate through messages with *commitment duration* (denoted by T in this paper). When a message has a zero-length commitment duration ($T = 0$), the

agent has no commitment (*zero-time commitment*). On the other hand, when a message has an infinite-length commitment duration ($T = \infty$), the message has eternal commitment (*infinite-time commitment*). When a message has a finite-length commitment duration ($T = \alpha, 0 < \alpha < \infty$), the message is valid for the specified duration (*finite-time commitment*).

3.2. New message type for efficient negotiation

Explicit use of commitment duration leads to a new type of message for efficient negotiation. If an agent is computing its resource for decision-making in an ongoing negotiation but the commitment duration given by its partner is almost expiring, it needs to send a message that asks for an extension of the commitment duration. Without this message type, the negotiation may be terminated despite both agents wanting to continue their negotiation process. We call this message request for extending commitment duration (ReqECD) and its response is named as response to extending commitment duration (ResECD).

3.3. Motivations of using and demanding commitment duration

To justify the real world applicability of the TBNF, we need to check each agent's motivation in using commitment duration itself, or in demanding commitment duration of its partner. For example, in a cooperative and mediated environment, a central agent can ensure global performance by enforcing the use of commitment duration. On the other hand, in a self-interested environment, each agent can generate a message with commitment duration only in its self-interest. Here are some examples:

Committed task announcement. A manager agent can use a committed announcement of a task to contact the contractor agents sequentially and select one of them.

Committed bid submission. A manager agent can demand a committed submission from contractor agents to make a safe choice among the committed alternatives, and contractor agents can use the committed bid submission for: (1) increasing the probability of getting the award from the manager agent

by providing a safe choice or (2) maintaining internal consistency for itself and escaping responsibility for possible future rejection of the awards from the manager agent.

Committed bid awarding. Bid awarding is inherently a committed action. A manager can use finite-time commitment to contract sequentially with contractors or expedite the acknowledgements for the contractors.

4. Semantics of each message with commitment duration

By analyzing the meaning of the commitment duration, the semantics of any negotiation messages can be precisely interpreted. Before the introduction of the commitment duration concept, strictly speaking, each message can only be an announcement without a guarantee or commitment. Under TBNF, every message has an option regarding whether it commits or not to a specific task, and a new commitment concept, *negative commitment*, is suggested. The following is the interpretation of each message.

4.1. Task announcement (from manager to contractor) with commitment duration T

Common interpretation. ''Submit the bid for this task to me (manager) within duration T. If you submit the bid within T then I will award the grant to you''.

Infinite-time commitment $(T = \infty)$. A contractor agent has only to submit the bid at any time. The award is guaranteed (automatic awarding). This scheme can be used between cooperating agents such as in internal hierarchical scheduling.

Finite-time commitment $(T = \alpha, 0 < \alpha < \infty)$. If a contractor agent submits the bid before α, the contract should be awarded to the contractor agent.

Zero-time commitment $(T = 0)$. It is implicitly interpreted that even if a contractor submits the bid to a manager, there is no guarantee the bid will be awarded. Interestingly, the use of expiration time of the task announcement message [14] is the same as when the commitment duration is zero because the expiration time has nothing to do with the manager's commitment. The commitment duration is not the same as the deadline for receiving bids. As pointed out in Ref. [10], it should be noted that when the commitment duration is longer than zero, the number of recipients of the announcement message should be one since the same task set cannot be awarded exclusively to the multiple agents.

4.2. Bid submission (from contractor to manager) with the commitment duration T

Common interpretation. ''I (contractor) am submitting this bid to you and I'll reserve my resources for the task for the duration T''.

Infinite-time commitment $(T = \infty)$. A manager agent does not have to send the bid award message to a contractor agent because the contractor agent already committed its resource for the bid.

Finite-time commitment $(T = \alpha, 0 < \alpha < \infty)$. If a manager agent awards the bid within α, the contractor agent can use its resources for the bid.

Zero-time commitment $(T = 0)$. It is implicitly interpreted that even though the manager agent awards the bid as soon as it receives the submission, there is actually no guarantee that the contractor will commit its resources for the bid.

4.3. Bid rejection (from contractor to manager) with the commitment duration T

It is interesting to see that even the message of the rejection of a bid has a concept of commitment. We may call this *negative commitment*.

Infinite-time commitment $(T = \infty)$. Due to the limited capacity of the contractor agent, it will not be able to submit the bid for the task.

Finite-time commitment $(T = \alpha, 0 < \alpha < \infty)$. Although the contractor agent cannot submit the bid right now, after the duration α the contractor agent may be able to submit the bid.

Zero-time commitment $(T = 0)$. It is implicitly interpreted that the contractor agent cannot submit the bid right now, but it may be able to submit the bid soon.

4.4. Bid award (from manager to contractor) with the commitment duration T

For resource-bounded contractors, the final step for the agreement should not be the bid award by the

manager, rather, it should be the acknowledgement of the bid award by the contractor because the bid submission cannot guarantee the acceptance of a bid award without any pre-commitment. In the traditional Contract Net Protocol, it is assumed that if the manager awards the bid to a contractor, then the contractor can accept or reject it, and the manager accepts the response unconditionally. So, the bid award in traditional Contract Net Protocol can be viewed as an eternally committed message. Therefore, we can interpret the traditional Contract Net Protocol as being a special case protocol where the committed award message has an infinite commitment duration and both the task announcement message and the bid submission message have the zero-commitment duration. Under TBNF we can use finite-time commitment duration in the bid award message in order to demand rejection or acceptance from the contractor in the specified time.

Infinite-time commitment ($T = \alpha$). The contract is completed and the contractor agent does not have to reply.

Finite-time commitment ($T = \alpha, 0 < \alpha < \infty$). For the final agreement, the contractor should send an acknowledgement within α. If not, the award can be canceled.

Zero-time commitment ($T = 0$). Irrelevant to the semantics of bid awarding.

4.5. Acknowledgement (award acceptance / rejection)

The final confirmation message inherently has infinite commitment duration ($T = \infty$).

5. Comparison of the three protocols in TBNF

By employing the length of the commitment duration in TBNF, we can formally define the above three typical negotiation protocols.

1. NGP is defined as the protocol where every negotiation message has zero-length commitment duration ($T = 0$).
2. AGP is defined as the protocol where the task announcement message or the bid submission message has infinite-length commitment duration ($T = \infty$).
3. FGP is defined as the protocol where one or more message types have finite-length commitment duration ($T = \infty$).

The protocols have comparative characteristics which are summarized in Table 1.

5.1. Contract process complexity

In terms of the contract process complexity, FGP is the most complex. It has to deal with the concept of the negative commitment and new message types such as ReqECD and ResECD.

5.2. Negotiation efficiency

Since agents in NGP do not guarantee anything until the final agreement, the negotiation process can be inefficient when resources are highly constrained or the decision-making time (deliberation time) of agents is relatively long. On the other hand, FGP can be effective in a highly constrained situation as we saw in our example scenario. AGP has a simple contract process and is efficient in negotiation, but it

Table 1
Protocol comparison summary

Protocols	NGP	AGP	FGP
Contract process complexity	simple	simple	complex
Negotiation efficiency	inefficient when highly resource-constrained	efficient	efficient when highly resource-constrained
Predictability and informativeness	low	high	high
Alternative availability	high	low	high
Communication overhead	high when highly resource-constrained	low	high when not optimized
Implementation complexity	simple	simple	complex
Strategic variety	low	low	high

can be used only for a special situation such as the hierarchical or cooperative coordination of agents.

5.3. Predictability and informativeness

In FGP, agents can enjoy more predictability (e.g., managers can select the pre-committed bid). FGP is the most informative protocol among the three protocols since each agent can infer the resource status or the intention of other agents from the commitment duration. This information can expedite the agreement between participating agents. Agents in NGP cannot have sufficient information on the status of other agents, which can lead to inefficiency in negotiation.

5.4. Alternative availability

In AGP, alternatives for the contract partner of an agent are greatly reduced because it should communicate with the agents from which it can get a guarantee for the acceptance of the offer. On the other hand, in FGP, agents can enjoy more availability. Manager agents can use a finite-time commitment in a task announcement message to choose a good candidate by sending the task announcement sequentially. In addition, manager agents can contact sequentially the contractor agents who submitted the bid by using the finite-time commitment in a bid award message.

5.5. Communication overload

In a highly constrained and dynamic situation, communication load can be high because of negotiation failure especially under NGP. Under FGP, too short a commitment duration can lead to unnecessary communication because of confirmation and computation for feasibility checking. On the other hand, in FGP with too long a commitment duration, agents may respond negatively to requests from their partners. Therefore, it is necessary to find an optimal commitment duration. The optimal lifetime of messages can be determined for each agent's performance or for global performance.

In addition, if the computation time of an agent is unpredictable, it should send many ReqECD, which increases the amount of communication.

5.6. Implementation complexity

In FGP, each agent should have a sophisticated action scheduling mechanism and a message management procedure such as the local scheduler [4] for managing message transmission and its own resources. To react in such a time-constrained situation, each agent may need some type of anytime-algorithm capability [1]. In addition, FGP, a time-dependent scheme, requires that the sending or receiving time of a message be verified by both parties [10]. That is, there should be a mechanism for all agents to agree on the message arrival time. To solve such a problem, a method has been proposed [11] which can carry out the electronic commerce transaction without requiring any third-party enforcement.

5.7. Strategic variety

FGP has more strategic varieties than the other two protocols. For example, the contractor agents can use the finite-time guarantee to promote a bid award from a manager agent.

6. Experiment for validating the framework

One of the benefits from using TBNF is that it provides a background for finding an appropriate architecture and protocol for a specific domain and situation. In this section, we show the usefulness of TBNF by an experiment which finds an optimal commitment duration in a bidding message when the contractor's resources are highly constrained.

6.1. Design of the experiment

For an experiment, we assume that we have three manager agents (i.e., SMAs) and two contractor agents (i.e., DCAs). Each delivery company is assumed to have one unit resource (i.e., a truck), which is used up when the contract is set-up and becomes available when a certain time (i.e., delivery time is randomly distributed from 6 to 10 s in this experiment) has passed after the contract.

To simplify the experiment and see the pure effect of the commitment duration in bidding message, we

set each commitment duration in the task announcement and award at zero. The acknowledgement message inherently has infinite commitment duration. The flow of the messages in the contract process is depicted in Fig. 3.

When the contract process between two agents begins, as described in Table 2, there can be four kinds of results such as bid rejection by contractor (B), award rejection by manager (Q), acknowledgement rejection by contractor (X), and successful contract (Y). In the experiment, we will examine the frequency of each result by changing the commitment duration in bid submission. Optimal commitment duration will be the duration which maximizes the number of successful contracts.

We developed the agent testbed for the experiment using Oracle Database and C language in a UNIX environment and the user interface on the World Wide Web. The current architecture (Fig. 4) is simplified for the experiment but it has a scalable structure.

The multi-agent system is composed of a shopping mall agent and a delivery company agent. Each has a communication manager sending and receiving messages and a message queue for storing incoming messages. The shopping mall agent has a task announcement manager which builds and announces a task, a history database of a task announcement, and an award manager which awards the bid to a selected delivery company agent. The delivery company agent contains a bid manager which analyzes an announcement and submits a bid to the shopping mall agent, a bidding history database, and an acknowledgement manager which receives an award and finally sends a confirmation message to the shopping mall agent. In order to implement an agent system which operates

Table 2
Contract success and rejection types
B: Bid rejection by contractor (delivery company), Q: award rejection by manager (shopping mall) after bid, X: acknowledgement rejection by contractor (delivery company) after award, Y: successful contract, S: shopping mall agent, D: delivery company agent.

Negotiation types	B		Q		X		Y	
Agent	S	D	S	D	S	D	S	D
Task announcement	OK		OK		OK		OK	
Bid		No		OK		OK		OK
Award			No		OK		OK	
Acknowledgement							No	OK

successfully under the real electronic commerce environment, each agent needs an action scheduling module which can optimally allocate time and resources for his decision-making [1]. To improve the performance of TBNF, it is important to implement the optimal decision-making for message prioritizing and reallocating the decision resources for new arriving messages. However, for this experiment, we implemented a simple system where each agent processes messages in a message queue by an FIFO method.

6.2. Agent knowledge for experiment

In this experiment, each agent uses KQML as the communication language and UNIK-OBJECT as knowledge representation for internal reasoning. The following shows the KQML-type messages for task announcement and bid rejection :

· Announcement
  (evaluate
    : **sender** SHOPPING MALL AGENT
    : **receiver** DELIVERY AGENT
    : **reply-with** msg_981119_O1
    : **ontology** Agent-Based Commerce
    : **language** UNIK-OBJECT
    : **content**
      ((title RFP)
        (order
          (product
            (item_no 01)
            (product_id CD-ROM 32X)
            (quantity 100))
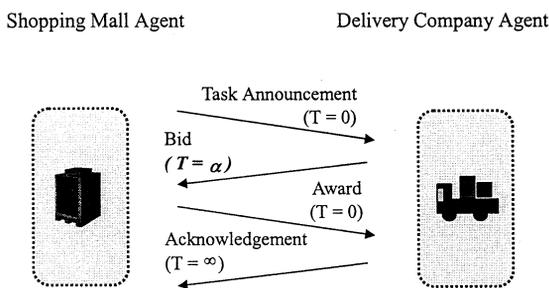

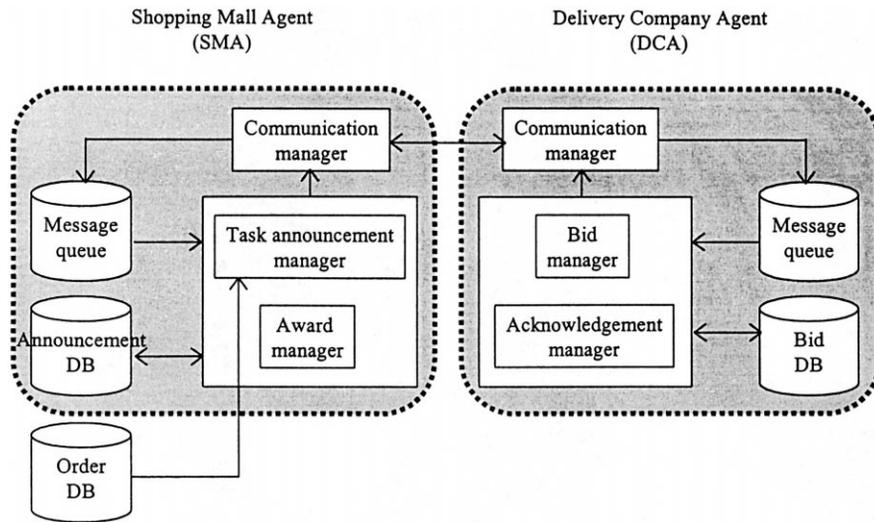Fig. 3. Commitment duration in bidding message.

Fig. 4. Architecture of agents for experiment.

```
(delivery
  (recipient
    (name SHOPPING MALL)
    (address Seoul, Korea))
  (delivery_date 10:00 NOV 23,
  1998)
  (delivery_method truck)))))
```
· Bid: Negative commitment
```
(reply
  : sender DELIVERY AGENT
  : receiver SHOPPING MALL AGENT
  : reply-with msg_981119_O1
  : ontology Agent-Based Commerce
  : language UNIK-OBJECT
  : content
    ((title BID)
      (bid
        (product
          (item_no 01)
          (product_id CD-ROM 32X))
        (Y/N No)
        (Committed_Until 13:00 NOV 20,
        1998)))))
```

The internal knowledge for each agent is different according to application domain. For the delivery problem in this paper, the internal decision-making knowledge for the shopping mall agent and the delivery company agent can be summarized as in Table 3. Since the aim of this experiment is to validate the usefulness of TBNF, we implemented simplified knowledge and performed the experiment.

6.3. Result of the Experiment

We observed the trend of frequencies for each result in the contract process while increasing the commitment duration of the bidding message by 5 s per experimental run from zero to 20 s. Table 4 summarizes the results and Fig. 5 shows the trend of the contract results.

As we see in Fig. 5, the success ratio depends on the length of bid commitment duration; we observe that there exists an optimal commitment duration in a bidding message, which maximizes the contract success rate. It means that proper use of commitment duration in messages gives better results than the nothing-guaranteed protocol which is now frequently used. For the best performance it is necessary to find the optimal commitment duration for each specific situation. The trend of bid rejections confirms the plausible fact that the longer the commitment duration of the bidding message the more conservative

Table 3
Internal knowledge for each agent

	Message type	Knowledge type and descriptions
All agents	All messages	Length of commitment duration:
		decide on the length of commitment duration of message
		Request to extend commitment duration:
		decide on whether requesting or not for extending commitment duration of messages
		Response to extension request:
		decide on whether accepting or rejecting requests for extending commitment duration
Shopping mall agent	Task announcement	Order clustering:
		cluster customer's orders considering location of recipient and delivery date
		Delivery company selection:
		select delivery companies suitable for the clustered jobs
	Award	Optimal bid combination:
		compose the optimal bid combination among submitted bids
Delivery company agent	Bid submission	Delivery scheduling:
		schedule the resource of delivery company with task announcements
		Shopping mall company selection:
		select shopping mall companies to submit a bid
	Acknowledgement	Decision on acknowledgement:
		decide on final confirmation of award message

agents are in bid submission. On the other hand, we see that the acknowledgement rejection decreases as the commitment duration of the bidding process increases.

6.4. Variables for future experiment

In this paper, we performed an experiment using the length of the bid commitment duration as the

Table 4
Experimental results
The numbers in parentheses denote the frequencies. B: bid rejection by contractor (delivery company), Q: award rejection by manager (shopping mall) after bid, X: acknowledgement rejection by contractor (delivery company) after award, Y: successful contract.

Commitment duration (T)	B(%)	Q(%)	X(%)	Y(%)	Total
0	0	5	69	26	
	(0)	(1)	(13)	(5)	(19)
5	17	12	47	23	
	(3)	(2)	(8)	(4)	(17)
10	62	5	0	33	
	(49)	(4)	(0)	(26)	(79)
15	62	12	0	26	
	(50)	(10)	(0)	(22)	(82)
20	60	18	0	22	
	(24)	(7)	(0)	(9)	(40)

only variable. In future experiments, the following variables can be used (we include expected results, as well):

Degree of resource constrainedness. As the resource is the more constrained, the more the optimal commitment duration of the total system will tend to increase.

Time for checking the resource availability. If the time to check the resource availability is unstable, the performance of FGP will tend to decrease. As this time gets longer, the optimal commitment duration will tend to increase.

Use and length of commitment duration in each message (task announcement, bid, and award). This
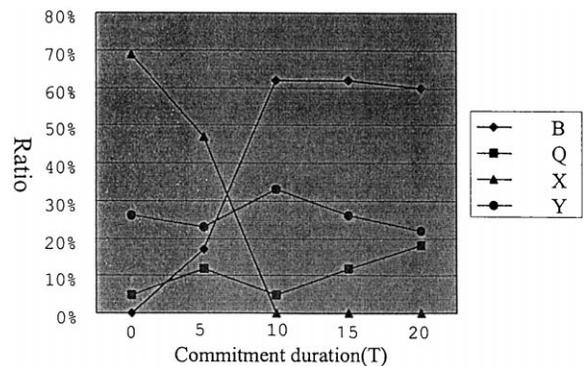

Fig. 5. Trend of contract results.

variable is to confirm the differences of each agent's performance according to the strategies that each agent chooses as well as the performance of the total system.

7. Related research

A field similar to the commitment duration 'Time Valid Through' was suggested in the extension of the Contract Net Protocol [10]. This field describes how long an offer on an alternative is valid and suggested as one of the various commitment methods with the penalties of decommiting. If the negotiation partner has not answered by the specified time, the sender of the message gets decommitted from that alternative. While the research deals with a rational decommiting scheme based on marginal cost calculation, we do not include the issues of decommitting in this paper. Instead, we try to formalize the Contract Net Protocol based on a negotiation framework by emphasizing the concept of the commitment duration, introducing new message types such as

ReqECD and creating new commitment concept in bid rejection, (*negative commitment*) to promote the successful completion of the negotiation between the agents.

In Ref. [12], the commitment concept is studied at the negotiation agent's strategy level while TBNF treats it at the architecture and protocol levels. We can say that they used the 'oscillation-type' of adaptive strategy between two extremes ($T = 0$ and $T = \infty$), while TBNF provides the opportunity to employ an adaptive strategy between moderate alternatives (such as $T = 0$ and $T = \alpha, T = \alpha$ and $T = \beta, T = \beta$ and $T = \infty$, when $\alpha < \beta$). We expect that a moderate strategy can outperform the oscillatory strategy in many situations. Furthermore, we can give each heterogeneous agent different commitment durations depending on various characteristics and each agent can change the length of the commitment duration on its own and employ the flexible strategies.

Collins et al. [2] study the temporal strategies in Contract Net Protocol and show how the selection of the timing elements within the protocol can affect the behaviors of the agents involved in the negotiation. However, a temporal strategy of an agent with-
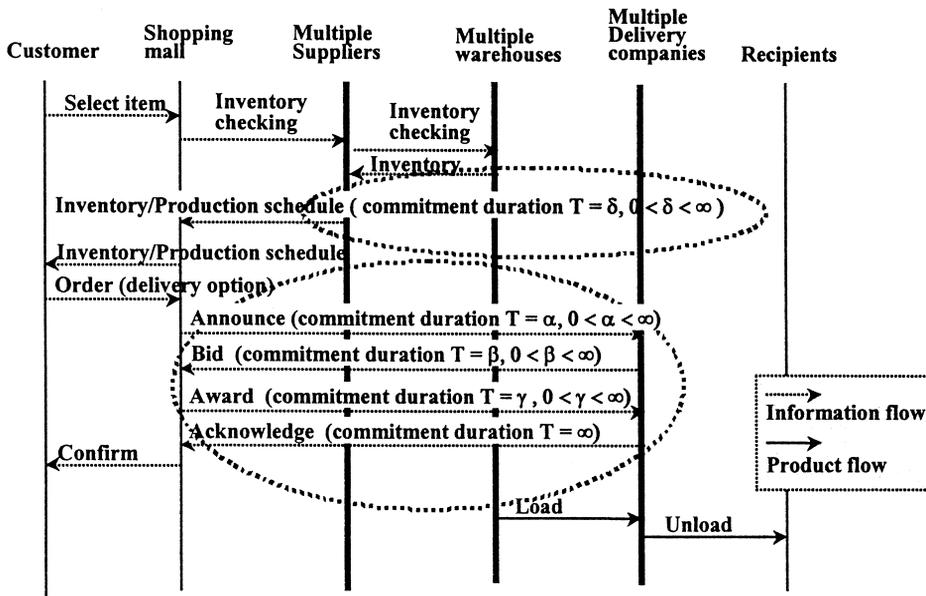

Fig. 6. Message flows in supply chain management for virtual shopping malls.

out commitment is only a declaration, so it does not have any enforcing mechanism for the contract between two agents. Therefore, in their scheme a mendacious agent can have an advantage over the other honest agents.

8. Conclusion

We expect that TBNF will be well suited for choosing a good protocol for the situation where self-interested and resource-constrained agents negotiate in a dynamic situation such as in a full-scale supply chain management for virtual shopping malls. Fig. 6 shows message flows in the context where the messages are exchanged among shopping mall agents, supplier agents, and delivery company agents etc. In summary, the merits of TBNF are.

1. TBNF provides a more informative framework with richer semantics.
2. TBNF provides the framework for promoting and expediting the negotiation process by allowing agents more strategic alternatives.
3. TBNF provides the background for finding an appropriate architecture and protocol for a specific domain and situation.
4. TBNF provides the background for efficient and effective multi-agent coordination while accommodating each agent's adaptive negotiation strategy.

Future research topics related to TBNF include research on the desirable architecture of agents under a specified framework, the optimal commitment scheme for competition, and how agents cooperate. In addition, research should be conducted on a time-stamping mechanism for all agents to verify and agree on the sending or receiving time of a message, which is regarded as one of the functions of certificate authorities for electronic commerce [6].

References

[1] M. Boddy, T. Dean, Solving time-dependent planning problems, in: Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, 1989, pp. 979–984.

[2] J. Collins, S. Jamison, M. Gini, B. Mobasher, Temporal strategies in a multi-agent contracting potocol, in: Proceedings of AAAI-97 Workshop on Using AI in Electronic Commerce, Virtual Organizations, Enterprise Knowledge Management to Reengineer the Corporation, 1997, pp. 50–56.

[3] R. Davis, R. Smith, Negotiation as a metaphor for distributed problem solving, Artificial Intelligence 20 (1) (1983) 63–109.

[4] K. Decker, Environment Centered Analysis & Design of Coordination Mechanisms, Ph.D. Dissertation. University of Massachusetts at Amherst, Department of Computer Science (Advisor: Dr. Victor R. Lesser) 1995

[5] K. Fisher, J. Muller, M. Pischel, D. Schier, A Model for Cooperative Transportation Scheduling, in: Proceedings of the First International Conference on Multi-Agent Systems, 1995, pp. 109–116.

[6] A. Froomkin, The Essential role of trusted tird prties in electronic commerce, in: R. Kalakota, A. Whinston (Eds.), Readings in Electronic Commerce, Addison-Wesley, 1997, pp. 119–178, Chap. 6.

[7] N. Jennings, Commitments and conventions: the foundation of coordination in multi-agent systems, Knowledge Engineering Review 8 (3) (1993) 223–250.

[8] G. Lin, J. Solberg, Integrated shop floor control using autonomous agents, IIE Transactions 24 (3) (1992) 57–71.

[9] T. Malone, R. Fikes, K. Grant, M. Howard, Enterprise: A Market-like Task Scheduler for Distributed Computing Environments, in: The Ecology of Computation North-Holland, Amsterdam, 1988, pp. 177–205.

[10] T. Sandholm, V. Lesser, Issues in automated negotiation and electronic commerce: extending the contract net framework, in: Proceedings of the First International Conference on Multi-Agent Systems, 1995, pp. 328–335.

[11] T. Sandholm, Unenforced e-commerce Transactions, IEEE Internet Computing (1997) 47–54.

[12] S. Sen, E. Durfee, The role of commitment in cooperative negotiation, International Journal of Intelligent and Cooperative Information Systems 3 (1) (1994) 67–81.

[13] M. Shaw, A. Whinston, Task bidding and distributed planning in flexible manufacturing, in: Proceedings of the Second IEEE Conference on Artificial Intelligence Applications, Miami, 1985, pp. 184–189.

[14] R. Smith, The contract net protocol: high-level communication and control in a distributed problem solver, IEEE Transactions on Computer 29 (1980) 1104–1113.

[15] O. Takuya, H. Kazuo, A. Yuichiro, Reducing communication load on contract net by case-based reasoning–extension with directed contract and forgetting, in: Proceedings of the Second International Conference on Multi-Agent Systems, 1996, pp. 244–251.

[16] H. Van Dyke Parunak, Manufacturing experience with the contract net, in: Proceedings of the Distributed Artificial Intelligence Workshop, 1985, pp. 67–91, December.

[17] N. Vojdani, Distributed manufacturing control using fuzzy contract net, in: M. Jamshidi, L. Zadeh (Eds.), Applications of Fuzzy Logic, Prentice-Hall, Canada, 1997.

Kyoung Jun Lee is an Assistant Professor of MIS at the School of Business of Korea University. He received his BS (1990), MS (1992), and PhD (1995) in Management Science from the Korea Advanced Institute of Science and Technology (KAIST). From 1996 to 1997 he worked as a visiting scientist of the Robotics Institute of the Carnegie Mellon University. He has twice won the Innovative Applications of Artificial Intelligence (IAAI) Award in 1995 and 1997. He was a senior researcher of the International Center for Electronic Commerce (ICEC) from 1997 to 1998. He has published papers in AI Magazine, Expert Systems with Applications, and the European Journal of Operational Researches. His curret research interests are on developing intelligent infromation systems for electronic commerce, supply chain management, and planning/scheduling problems etc.

Yong Sik Chang is a PhD candidate at the Graduate School of Management at the Korea Advanced Institute of Science and Technology (KAIST) and a senior researcher at the International Center for Electronic Commerce (ICEC). He received his BS (1988) in Physics from Sogang University and MS (1991) in Physics from Pohang Institute of Science and Technology (POSTECH). He has experience in developing MIS, intranet and electronic commerce apllications for POSDATA, Infoware, and ICEC. His current research focuses on electronic commerce, supply chain management and multi-agent systems.

Jae Kyu Lee is a Professor of Management Information Systems at the Korea Advanced Institute of Science and Technology and Director of the International Center for Electronic Commerce. He received a BA from Seoul National University, a MS from the Korea Advanced Institute of Science and Technology and a PhD from the Wharton School, University of Pennsylvania. He has previously served as an exchange professor at Carnegie-Mellon University (1989) and University of Texas at Austin (1994). He was the chair of the '98 International Conference on Electronic Commerce (ICEC) and the 3rd World Congress on Expert Systems (1996). He has authored several books on Electronic Commerce and Expert Systems and published numerous papers in the following journals: Management Science, Decision Support Systems, Expert Systems with Applications, Expert Systems, Decision Science and many others. Currently, he is an editor of various international journals like Decision support Systems, Expert Systems with Applications, and International Journal of Electronic Commerce. His main research interests are in the fields of Electronic Commerce and Intelligent Management Information Systems.