Time-Bounded Negotiation Framework for Multi-Agent Coordination

Kyoung Jun Lee[1] and Yong Sik Chang[2]

[1] School of Business, Korea University
Anam-Dong, Sungbuk-Ku, Seoul 136-701, Korea
leekj@kuba.korea.ac.kr
[2] Graduate School of Management, Korea Advanced Institute of Science and Technology
207-43 Cheongryangri-Dong, Dongdaemoon-Ku, Seoul 130-012, Korea
yschang@msd.kaist.ac.kr

Abstract. For the efficient and informative coordination of multiple agents, a time-bounded agent negotiation framework is proposed utilizing time-based commitment scheme. By attaching the commitment duration to agent messages, the traditional Contract Net Protocol is extended to a time-bounded environment, thereby giving rise to a *Time-Bounded Negotiation Framework* (TBNF). The proposed negotiation framework has a new message type to agree upon the extension of a commitment duration, and a novel commitment concept in the form of Negative Commitment. We interpret the semantics of the messages with the commitment duration, and then formally define and compare the three typical negotiation protocols - nothing-guaranteed protocol, acceptance-guaranteed protocol, and finite-time guarantee protocol - which can be incorporated into TBNF. The Time-Bounded Negotiation Framework should provide a background for efficient and effective agent coordination while accommodating each agent's adaptive negotiation strategy.

1 Introduction

Efficient coordination of multi-agents is very important for the performance of each agent and the whole system. The Contract Net Protocol [7, 14] has been the most commonly used method for coordinating agents in negotiation. The Contract Net Protocol specifies communication and control in a distributed problem solver [3], that is, how contract managers announce 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 task). The basic steps of the protocol is as follows.

- A manager issues a task announcement describing the task to be done and the criteria for bids.
- Contractors send bids to announce their willingness and ability to perform a task.

T. Ishida (Ed.): PRIMA'98, LNAI 1599, pp. 61-75, 1999.

- The manager sends the award message to a successful contractor
- The contractor sends an acknowledgement of the award, either accepting or rejecting it.

Since the Contract Net Protocol can be somewhat simple for a specific purpose and needs to be modified to satisfy various system requirements and improve performance, there have been a couple of researches on the extension of the protocol by:

- Introducing new speech acts such as temporal grant, temporal reject, definitive grant, definitive reject for the case when tasks exceed the capacity of a single agent. The context involves an extended Contract Net Protocol [5].
- Fuzzy theoretic method for determining next task announcement and the best fitting bid [17].
- Using directed contract and forgetting by case-based reasoning to reduce communication load [15].
- Enabling agents to choose level of commitment 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-bounded framework provides a good background for choosing a proper negotiation protocol especially for dynamic situation where desired tasks and available resources may be changing as the system is executing tasks. The following is the scenario from the delivery problem in a virtual shopping mall, which inspired this research. When a virtual shopping mall receives product orders from customers, it needs to make delivery orders automatically without human intervention, generate a request for proposal (RFP) and announce it to multiple delivery companies. 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, contractor agent (i.e. delivery company) is resource-constrained, and the status of agents is fast changing.

Example scenario from delivery problem in virtual shopping mall context

1. A shopping mall agent (SMA1) asks a delivery company agent (DCA) whether it can deliver a product (PA) to its buyer from a warehouse in three 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.
4. During the meantime, another shopping mall agent SMA2 asks the delivery company agent (DCA) whether DCA can deliver a product PB to its buyer from a warehouse in three days.
5. DCA tries to schedules its own facility for the delivery of PB and comes to know that it cannot deliver PB to the buyer for SMA2 on time without canceling the capacity reservation for SMA1.

In this case, what should DCA do? We can consider using three alternative protocols to cope with this problem: 1) *Nothing-Guaranteed Protocol*, 2) *Acceptance-*

Guaranteed Protocol, and 3) *Finite-time Guarantee Protocol*. The next section briefly explains each of them.

2 Three Candidate Protocols for the Scenario

2.1 Nothing-Guaranteed Protocol (NGP)

Most of protocols based on traditional Contract Net Protocol assume this [8,9,13,16]. In this protocol, no one has to take any responsibility before reaching mutual agreement on the task. In our scenario, DCA feels free to reply 'Yes' to SMA2, and waits for the confirming 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 agreement. Therefore, each agent feels free to send a message to another agent. However, since nothing is guaranteed until a mutual agreement is made, some agents can feel nervous and the global coordination performance can be degraded. For example, when SMA1 receives the bid from DCA and decides to award the bid to DCA, but in the meantime, DCA can contract with SMA2, then DCA cannot accept the bid awarded from SMA1. In this case, SMA1 should start negotiation from the beginning. Traditional Contract Net Protocol is a protocol which does not guarantee anything before the final agreement. 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 can be unsettled because it has no further information until the final agreement.

2.2 Acceptance-Guaranteed Protocol (AGP)

In this protocol, an agent guarantees the positive response to the other agent's message. For example, a manager agent can guarantee the acceptance of a bid submission to a contractor agent when the manager agent announces a task, which means that the task announcer automatically will award the bid to the bidder on the condition that the submitted bid satisfies the constraints of the initial announcement. In addition, a contractor agent can guarantee the unconditional acceptance of the bid awarded from a manager agent. This protocol can be 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 the extreme case of audience restriction, which was discussed in [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 reducing the communication effort between agents. This protocol can be used for hierarchical (vertical) coordination between high-level agents and lower-level agents, where the high-level agents have only one partner in negotiation. If we apply this protocol to the above scenario, which assumes that DCA guarantees the acceptance of the bid

awarded from the SMA1, then DCA should reply 'No' to SMA2 because DCA already submitted the bid to SMA1.

2.3 Finite-Time Guarantee Protocol (FGP)

When a message has a kind of guarantee, that is, commitment, then the message needs to have a 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 its resource availability, it should send a message requesting the extension of the message life. The finite-time guarantee protocol addresses such needs by making every message have its own timed token over a valid duration. The message with the token is valid during the duration specified in the token and the timing token can be attached to request messages as well as reply messages. We may rewrite the above scenario using the protocol below.

1. SMA1 asks DCA whether it can deliver a product PA to its buyer in three days.
2. DCA schedules its own facility for the delivery of the product PA and replies 'Yes' to SMA1 with a timed token valid for 30 minutes.
3. SMA1 has not awarded the bid to DCA yet.
4. During the meantime, another shopping mall agent SMA2 asks DCA whether it can deliver a product PB to its buyer in three days.
5. DCA checks the current time.

– (a) If 30 minutes 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 timed token valid for 30 minutes. DCA need not feel any responsibility toward SMA1. (See (a) in Fig. 1)
– (b) If it is before the deadline (for example, it has taken 20 minutes after replying to SMA1): DCA should try to schedule its own facility for the delivery of PB with capacity reservation for SMA1. As the 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 try to request the order again after 10 minutes or wait 10 minutes to confirm message. (See (b) in Fig. 1)

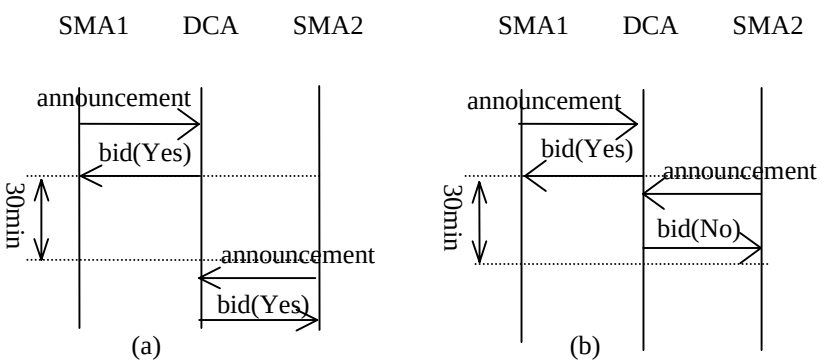

Fig. 1. Two situations when using finite-time guarantee protocol

3 Time-Bounded Negotiation Framework

To synthesize the above three protocols into a framework, we propose the Time-Bounded Negotiation Framework (TBNF) which is a meta-model of the Contract Net Protocol emphasizing the commitment duration attached to each negotiation message.

3.1 Definition of Time-Bounded Negotiation Framework (TBNF)

Time-Bounded Negotiation Framework (TBNF) is a negotiation framework where agents negotiate with messages with *commitment duration* (denoted by T in this paper). When a message has a zero-length commitment duration (T = 0), it has no commitment in it (*Zero-time commitment*). On the other hand, when a message has an infinite-length commitment duration (T = ∞), the message is interpreted to be valid eternally by the agent's commitment (*Infinite-time Commitment*). When a message has a finite-length commitment duration (T = α, $0 < \alpha < \infty$), the message is valid for the specified duration (*Finite-time commitment*).

3.2 New Message Type for Efficient Negotiation

The explicit use of the commitment duration leads to a new type of message for efficient negotiation. If an agent is computing its resource for decision making on an ongoing negotiation but the commitment duration given by its partner is almost expiring, it needs to send a message asking the extension of the commitment duration to its partner. Without this message type, the negotiation can be terminated despite both agents want to continue their negotiation process. We call this message as 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 Time-Bounded Negotiation Framework, we need to check each agent's motivation of using commitment duration for itself or demanding commitment duration to its partner. For example, in a cooperative and mediated environment, a central agent can ensure the 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 for self-interest as follows.

- *Committed task announcement*: Manager agent can use a committed announcement of a task to contact the contractor agents sequentially and select one of them.
- *Committed bid submission*: Manager agent can demand a committed submission to contractor agents for a safe choice among committed alternatives. On the other hand, contractor agents can use the committed bid submission for

1. increasing the probability of getting the award from manager agent by providing a safe choice or
2. maintaining internal consistency for itself and escaping the responsibility of the future possible rejection of the award from the manager agent.

- *Committed bid awarding*: Bid awarding is inherently a committed action. Manager can use finite-time commitment to contract sequentially with contractors or expedite the acknowledgement of the contractors.

4 Semantics of Each Message with Commitment Duration

By analyzing the meaning of the commitment duration, the semantics of any negotiation message can be precisely interpreted. Before the introduction of the commitment duration concept, strictly speaking, each message can only be an announcement without guarantee or commitment. Under TBNF, every message can have the 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 with the Commitment Duration

- *Common interpretation*: "Submit the bid about 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 = ∞)*: It says that contractor agent has only to submit the bid at any time. Awarding is guaranteed (Automatic Awarding). This scheme can be used between cooperating agents such as in internal hierarchical scheduling.
- *Finite-time commitment (T = α, 0 < α < ∞)*: It means that if a contractor agent submits the bid before α, the grant 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 manager, there is no guarantee it would be awarded. Interestingly, even using the expiration time of the task announcement message [14] corresponds to the case where 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 [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 multiple agents.

4.2 Bid Submission with the Commitment Duration

- *Common interpretation*: "I (Contractor) am submitting this bid to you and I'll reserve my resource for the task for the duration T".

- *Infinite-time commitment ($T = \infty$)*: The manager agent does not have to send the bid awarding 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 the manager agent awards the bid within α, the contractor agent can afford to use its resource 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 with the Commitment Duration

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 forever.
- *Finite-time commitment ($T = \alpha$, $0 < \alpha < \infty$)*: Though the contractor agent cannot submit the bid right now, but 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 Awarding with the Commitment Duration

For the resource-bounded contractors, the final step for the agreement should not be the bid awarding by the manager, but should be the acknowledgement of the bid awarding 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 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 awarding in traditional Contract Net Protocol can be viewed as an eternally committed message. Therefore, we can interpret that the traditional Contract Net Protocol is a special case protocol where the committed awarding message have infinite commitment duration and both the task announcement message and the bid submission message have zero-commitment duration. Under TBNF we can give the finite-time commitment duration to the bid awarding message in order to demand rejection or acceptance from the contractor in the specified time.

- *Infinite-time commitment ($T = \infty$)*: The contract is completed and the contractor agent do not have to reply to this.
- *Finite-time commitment ($T = \alpha$, $0 < \alpha < \infty$)*: For the final agreement, the contractor should send 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 the infinite commitment duration $(T = \infty)$

5 Comparison of the Three Protocols in Time-Bounded Negotiation Framework (TBNF)

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

1. Nothing-Guaranteed Protocol (NGP) is defined as the protocol where every negotiation message has zero-length commitment duration (T = 0).
2. Acceptance-Guaranteed Protocol (AGP) is defined as the protocol where the task announcement message or the bid submission message has infinite-length commitment duration $(T = \infty)$.
3. Finite Time-Guarantee Protocol (FGP) is defined as the protocol where one or more message types have finite-length commitment duration $(T = \infty)$.

The protocols have the comparative characteristics along the several criteria, which are summarized in Table 1.

Table 1. Protocol comparison summary

Protocols	NGP	AGP	FGP
Contract process	simple	simple	complex
Negotiation efficiency	inefficient when highly resource-constrained	efficient	efficient when highly resource-constrained
Predictability & 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

- *Contract process complexity*: In terms of the contract process complexity, Finite-time Guarantee Protocol is the most complex because it has to deal with the new message types such as ReqECD and ResECD and the concept of the negative commitment.
- *Negotiation efficiency*: Since agents in Nothing-Guaranteed Protocol 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, Finite-time Guarantee Protocol can be effective in highly constrained situation as we see in the example scenario. Acceptance-Guaranteed Protocol has a simple contract process and is efficient in negotiation, but it can be used only for a special situation such as the hierarchical or cooperative coordination of agents.

- *Predictability and informativeness*: In Finite-Guarantee Protocol, agents can enjoy more predictability (e.g. Managers can select the pre-committed bid). Finite-time Guarantee Protocol is the most informative protocol among the three protocols since each agent can infer the resource status or the intention of the other agent from the commitment duration. The informativeness can expedite the agreement between participating agents. Agents in Nothing-Guaranteed Protocol cannot have sufficient information on the status of other agents, which can lead to inefficiency in negotiation.
- *Alternative availability*: In Acceptance-Guaranteed Protocol, alternatives for the contract partner of an agent are much reduced because it should communicate with the agents to which it can guarantee the acceptance of the offer from its partner. On the other hand, in Finite-time Guarantee Protocol , agents can enjoy more availability. Manager agents can use a finite-time commitment in 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 bid awarding message.
- *Communication overhead*: In highly constrained and dynamic situation, the communication overhead can be high from the negotiation failure especially under Nothing-Guaranteed Protocol. Under a Finite-time Guarantee Protocol, too short a commitment duration can lead to unnecessary communication for confirmation and computation for feasibility checking. On the other hand, in a Finite-time Guarantee Protocol with too long a commitment duration, agents may respond negatively to requests from their partners. Therefore, it is necessary to find an optimal lengrh of the 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 (Request for Extending Commitment Duration), which increases the amount of communication.
- *Implementation complexity*: In Finite-time Guarantee Protocol, 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 a kind of anytime algorithm capability [1]. In addition, Finite-time Guarantee Protocol, 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 electronic commerce transactions that does not require any third-party enforcement.

- *Strategic variety*: Finite-time Guarantee Protocol has more strategic varieties over the other two protocols. For example, contractor agents can use the finite-time guarantee to promote bid award from manager agents.

6 Experiment in Electronic Commerce Context

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 the experiment which finds an optimal commitment duration in bidding message when the contractor's resource is highly constrained.

6.1 Design of Experiment

For the experiment, we assume that we have three manager agents (i.e. shopping mall agents) and two contractor agents (i.e. delivery company agents). Each delivery company is assumed to have one unit resource (i.e. a truck), which is used up when a contract is set-up and becomes available when a certain time (i.e. delivery time is randomly distributed from 6 seconds and 10 seconds in this experiment) is 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 task announcement and awarding to zero. The acknowledgement message inherently has the infinite commitment duration. All these message flows in the contract process is depicted in Fig. 2.

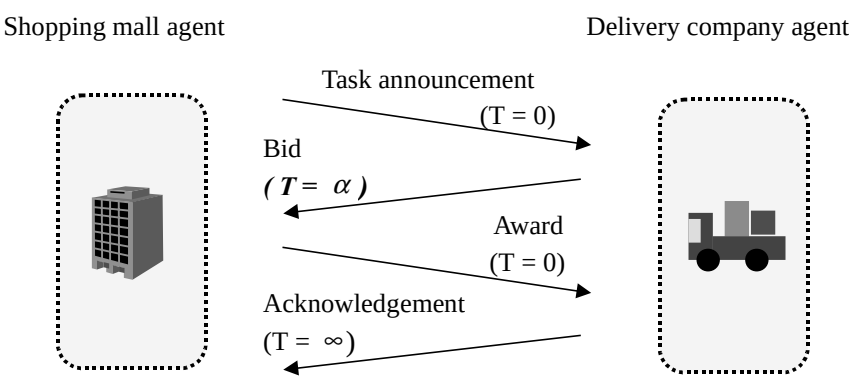

Fig. 2. Commitment duration in bidding message

When the contract process begins between the two agents, 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 see the frequency of each result by changing the commitment duration in bid submission. The optimal commitment duration will be the duration which maximizes the number of successful contracts.

Table 2. Contract success and rejection types

Negotiation types	B		Q		X		Y	
Agent	S	D	S	D	S	D	S	D
Task announcement	O.K.		O.K.		O.K.		O.K.	
Bid		No		O.K.		O.K.		O.K.
Award			No		O.K.		O.K.	
Acknowledgement						No		O.K.

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

We implemented the test bed of agents for the experiment using Oracle Database and C language in UNIX environment and the user interface on the WWW. The current architecture (Fig. 3) is simplified for the experiment but has a scalable structure.

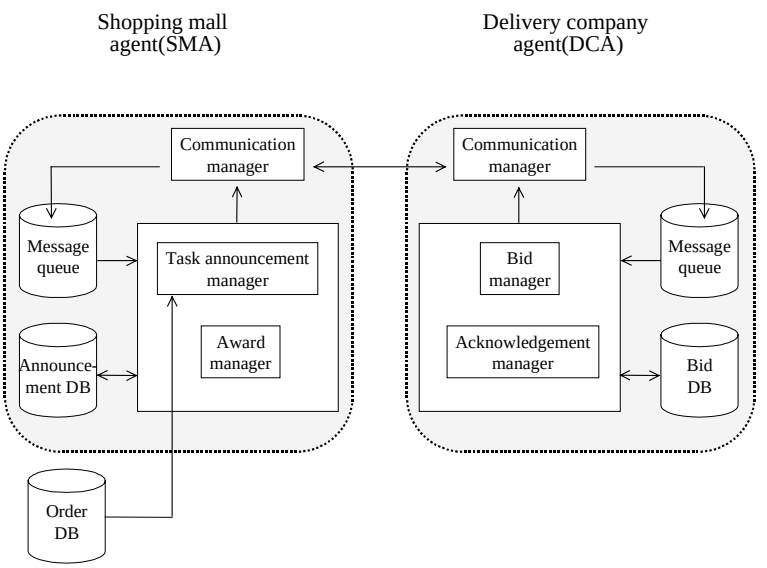

Fig. 3. Architecture of agents for experiment

The multiagent system is composed of shopping mall agent and delivery company agent. Each has a communication manager sending and receiving messages and a message queue storing the incoming messages. The shopping mall agent has a task announcement manager building and announcing task, a task announcement history database, and an award manager which awards the bid to a selected delivery company agent. The delivery company agent has 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 such an agent system operating successfully under real electronic commerce environment, each agent needs a kind of action scheduling module which can optimally allocate the time and resources for his decision making [1]. To improve the performance of TBNF, it is important to implement the optimal decision making on 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 message queue by FIFO (First-In First-Out) method.

6.2 Result of the Experiment

We observed the trend of frequencies of each result in the contract process while increasing the commitment duration of the bidding message by 5 seconds per experiment from zero to 20 seconds. Table 3 summarizes the result and Fig. 4 shows the trend of the contract result.

Table 3. Result of the experiment

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

(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

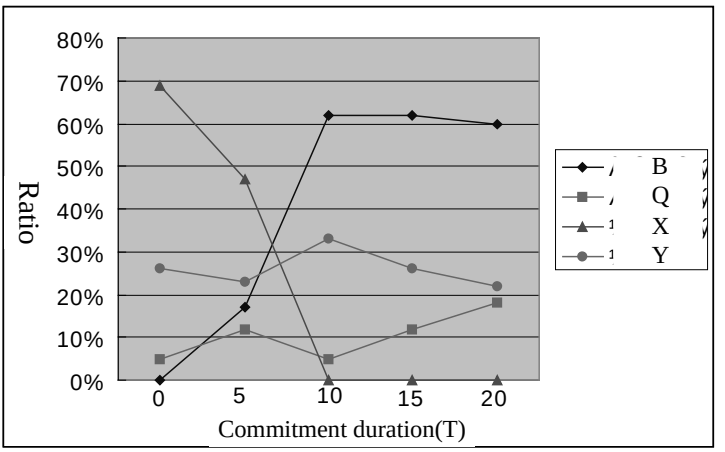

Fig. 4. Trend of contract result

As we see in Fig. 4, negotiation success ratio depends on the length of bid commitment duration; it is not good when the bid commitment duration is too short or too long. That is, there exists the optimal length of the commitment duration in bidding message, which maximizes the contract success rate. It means that a proper use of a commitment duration in messages can result in better result than the nothing-guaranteed protocol which has been frequently used so far, and for the best performance it is necessary to find the optimal commitment duration for a specific situation. The trend of the bid rejection confirms the plausible fact that the longer the commitment duration of bidding message the more conservative in bid submission. On the contrary, we can see that the acknowledgement rejection decreases as the commitment duration of bidding increases.

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 various commitment methods with the penalties of decommiting. If the negotiation partner has not answered by that 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 decomitting in this paper. Instead, we try to formalize the CNP (contract net protocol) based negotiation framework by emphasizing the concept of the commitment duration, introducing new message types such as ReqECD (Request for Extending Commitment Duration) and creating new commitment concept in

bid rejection, that is, *Negative Commitment* to promote the successful completion of the negotiation between agents.

In [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=∞), while TBNF provides the opportunity to employ an adaptive strategy between moderate alternatives (such as T=0 and T=α, T=α and T=β, T=β and T=∞, 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 duration depending on its various characteristics and each agent can change the length of the commitment duration at its own and employ 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 without 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 TBNF will be suited well to choose a good protocol for the situation where self-interested and resource-constrained agents negotiate in a dynamic situation.

In summary, the merits of TBNF we can consider are as follows:

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 strategy 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 agent's adaptive negotiation strategy.

The future research topics related to TBNF include the research on the desirable architecture of agents under the framework and the optimal commitment scheme for competition and cooperation between agents. In addition, the research on a time stamping mechanism is necessary 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.

References

1. Boddy, M., Dean, T.: Solving Time-Dependent Planning Problems, Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (1989) 979-984
2. Collins, J., Jamison, S., Gini, M., and Mobasher, B.: Temporal Strategies in a Multi-Agent Contracting Protocol, Proceedings of AAAI-97 Workshop on Using AI in Electronic Commerce, Virtual Organizations, Enterprise Knowledge Management to Reengineer the Corporation (1997) 50-56
3. Davis, R., Smith, R.: Negotiation as a metaphor for distributed problem solving, Artificial Intelligence 20(1) (1983) 63-109
4. Decker, K.: Environment Centered Analysis & Design of Coordination Mechanisms, Ph.D. Dissertation. University of Massachusetts at Amherst, Department of Computer Science (1995)
5. Fisher, K., Muller, J., Pischel, M., Schier, D.: A Model for Cooperative Transportation Scheduling, Proceedings of the First International Conference on Multi-Agent Systems (1995) 109-116
6. Gu, C., Ishida, T.: Analyzing the Social Behavior of Contract Net Protocol, Agents Breaking Away; MAAMAW'96, Lecture Notes in Artificial Intelligence, Vol. 1038. Springer-Verlag, Berlin Heidelberg New York (1996) 116-127
7. Jennings, N.: Commitments and Conventions: The Foundation of Coordination in Multi-Agent Systems, Knowledge Engineering Review 8(3) (1993) 223-250
8. Lin, G., Solberg, J.: Integrated Shop Floor Control Using Autonomous Agents, IIE Transactions 24(3) (1992) 57-71
9. Malone, T., Fikes, R., Grant, K., Howard, M.: Enterprise: A Market-like Task Scheduler for Distributed Computing Environments, The Ecology of Computation, North-Holland (1988) 177-205
10. Sandholm, T., Lesser, V.: Issues in Automated Negotiation and Electronic Commerce: Extending the Contract Net Framework, Proceedings of the First International Conference on Multi-Agent Systems (1995) 328-335
11. Sandholm, T.: Unenforced E-Commerce Transactions, IEEE Internet Computing, (1997 Nov/Dec) 47-54
12. Sen, S., Durfee, E.: The Role of Commitment in Cooperative Negotiation, International Journal of Intelligent and Cooperative Information Systems 3(1) (1994) 67-81
13. Shaw, M., Whinston, A.: Task Bidding and Distributed Planning in Flexible Manufacturing, Proceedings of the Second IEEE Conference on Artificial Intelligence Applications, Miami (1985) 184-189
14. Smith, R.: The contract net protocol: High-level communication and control in a distributed problem solver, IEEE Transactions on Computer 29 (1980) 1104-1113
15. Takuya, O., Kazuo, H., Yuichiro, A.: Reducing Communication Load on Contract Net by Case-Based Reasoning -- Extension with Directed Contract and Forgetting --, Proceedings of the Second International Conference on Multi-Agent Systems (1996) 244-251
16. Van Dyke Parunak, H.: Manufacturing experience with the contract net. In Proceedings of the Distributed Artificial Intelligence Workshop (1985) 67-91
17. Vojdani, N.: Distributed Manufacturing Control Using Fuzzy Contract Net. In: Jamshidi, M., Zadeh, L. (eds.): Applications of Fuzzy Logic, Prentice Hall Canada (1997)