43

PARTNERING MECHANISM WITH ADAPTIVE MULTI-AGENT PROTOCOL FOR VIRTUAL ENTERPRISE

Toshiya Kaihara* and Susumu Fujii**

*Graduate School of Science and Technology, Kobe University
** Department of Computer and Systems Engineering, Kobe University, JAPAN
{kaihara, fujii}@cs.kobe-u.ac.jp

Although cooperation is the fundamental characteristic of Virtual Enterprise (VE) concept, due to its distributed environment and the autonomous and heterogeneous nature of the VE members, cooperation can only be succeed if a proper management of dependencies between activities is in place just like SCM. In this paper, we focus on negotiation process in VE formulation as a basic research to clarify the effective management for VE. Each enterprise in VE is defined as agent with multi-utilities and a framework of multi-agent paradigm with reinforcement learning algorithm, named Q-Learning, is newly proposed as an adaptive negotiation algorithm amongst the agents. We develop a computer simulation model to form VE through multiple negotiations amongst several potential members in the negotiation domain, and finally clarify the formulation dynamism on the negotiation process with adaptive behaviour.

1. INTRODUCTION

The introduction of information & telecommunications technologies and more recently, distributed object computing technology has enabled the creation of functioning Virtual Enterprises (VE), that do not have the geographic and structural restrictions that have traditionally constrained conventional enterprises. These technologies have enabled people to interact and collaborate effectively over distance as part of VE (Camarinha-Matos, 1999).

The ability to interact in these new ways raises two key areas of concern that must be addressed in order to effectively build collaboration systems that exploit distributed object computing. First, it is necessary to understand how people within and across organisations interact in a collaborative manner when undertaking tasks. As in many traditional enterprises and communities, people collaborate based on a set of informal, yet crucial procedures. These are often established in an ad-hoc fashion and without explicit enforcement mechanisms. Much work in the area of computer supported collaborative work (CSCW) has focused on capturing and supporting these kinds of interactions. The second and related area of concern is the need to understand and support the structured, competitive or even adversarial interactions that are characteristic to commerce, trade and inter-organisational interaction. This paper focuses primarily on the second issue dealing with structured interactions.

Traditionally, marketing, distribution, planning, manufacturing, and the purchasing organizations operated independently. These organizations have their own objectives and these are often conflicting. Marketing's objective of high customer service and maximum sales conflict with manufacturing and distribution goals. Many manufacturing operations are designed to maximize throughput and lower costs with little consideration for the impact on inventory levels and distribution capabilities. Purchasing contracts are often negotiated with very little information beyond historical buying patterns. The result of these factors is that there is not a single, integrated plan for the organization - there were as many plans as businesses. Clearly, there is a need for a mechanism through which these different functions can be integrated together. Although cooperation is the fundamental characteristic of VE concept, due to its distributed environment and the autonomous and heterogeneous nature of the VE members, cooperation can only be succeed if a proper management of dependencies between activities is in place just like Supply Chain Management (Fisher, 1994) (kaihara, 2002a).

In this paper, we focus on negotiation process in VE formulation as a basic research to clarify the effective management for VE. Each enterprise in VE is defined as agent with multi-utilities and a framework of multi-agent paradigm with reinforcement learning algorithm, named Q-Learning (Sutton, 1998), is newly proposed as an adaptive negotiation algorithm amongst the agents. Each enterprise unit is defined as agent in our VE model, and their decision makings are formulated as a game theoretic methodology (Von Neumann, 1947) with learning ability. We adopt Contract Net Protocol (CNP) (Smith, 1980) as the coordination and negotiation mechanism amongst the units. CNP models transfer of control in a distributed system with the metaphor of negotiation among autonomous intelligent beings. Nodes generally represent the distributed computing resources to be managed, correspond to "enterprises" in this paper. We develop a computer simulation model to form VE through multiple negotiations amongst several potential members in the negotiation domain, and finally clarify the formulation dynamism on the negotiation process with adaptive behaviour.

2. ENTERPRISE AGENT

2.1 Virtual Enterprise Modelling Structure

Generally it is very important and difficult activity in forming a virtual enterprise to select appropriate business partners, i.e. partnering, because each enterprise considers not only pursuing its profit but also sharing the risk to join the virtual enterprise. The partnering is described as coordination activity amongst the enterprises, and some sophisticated coordination mechanism is required to realise efficient interactions.

The development of coordination mechanism in computer science can be found in the area of workflow management system, computer supported cooperative work (CSCW), and multi-agent systems. The area of multi-agent systems, especially when involving intelligent autonomous agents, has been discussing coordination issues and supporting mechanism (kaihara, 2002a). The interaction capability, both amongst agents and between agents and their environment, is one of the basic characteristics of an agent.

In this paper we focus on the CNP, that is one of the mechanism coming from the early works on multi-agent systems (Smith, 1980), as the coordination and negotiation mechanism amongst business units in VE.
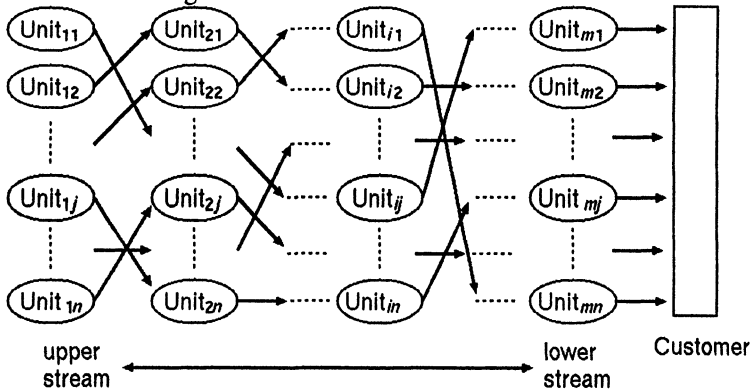

Figure 1 - VE Structure

Figure 1 shows the assumed VE structure in this paper. We call the enterprise as unit, and there exist m layers, which have m_n units in the VE model. The lowest level corresponds to consumers who can create original task requests to the VE. At first, the customer dispatches new order to all the units in layer m, and then several units, which are satisfied with the order, responds and circulates the order toward upper units in the VE model. Finally a VE with single supply chain will be established for the order as a consequence of their negotiations through all the layers. Regarding our precise negotiation algorithm with CNP, refer to (kaihara, 2002a) (kaihara, 2002b).

2.2 Basic Assumptions
There exist several situations in partnering amongst enterprise agents. In this paper it is assumed that the product demand is predictable in the negotiation under multipurpose criterion. That means order patterns are previously given and the negotiations start after the order reached to each enterprise agent. They should prepare robust solutions with maximum utilities against the order. We propose agent behaviours based on game theoretic approach with reinforcement learning algorithm according to this assumption.

2.3 Agent Behaviour
In this model, all the orders are clearly given before the negotiations. Agent behaviours are described in each negotiation step of CNP.
- Bidding

Each contractor (U_{ij}) has three attributes, such as cost, lead time and quality, in their bid for order k defined as follows:

$$Cost_{ij}^{k} = E_{ij}^{k} + D_{ij}^{k} + P_{ij}^{k} \tag{1}$$

$$Leadtime_{ij}^{k} = \lambda_{ij}^{k} / E_{ij}^{k} \tag{2}$$

$$Quality_{ij}^{k} = \mu_{ij}^{k} D_{ij}^{k} (1 - \exp^{-v_{ij}^{k} P_{ij}^{k}}) \tag{3}$$

where

Cost $_{ij}^{k}$: total cost for U_{ij} to process order k

Leadtime $_{ij}^{k}$: lead time for U_{ij} to process order k

Quality $_{ij}^{k}$: product quality for U_{ij} to process order k

$E_{ij}^{k}, D_{ij}^{k}, P_{ij}^{k}$: equipment / development / personnel cost

λ_{ij}^{k} : coefficient of leadtime

μ_{ij}^{k}, v_{ij}^{k} : coefficients of quality

Cost vs. lead time, and cost vs. quality, are in trade off relationship in those equations with reality.

- Reward

After the bidding by contractors, managers compute following pay-off matrix according to their utilities against all the bids.

$$
\begin{array}{c}
\quad\quad\quad C \quad\quad L \quad\quad Q \\
\begin{array}{c}
Unit_{i1} \\
Unit_{i2} \\
... \\
Unit_{ij} \\
... \\
Unit_{in}
\end{array}
\left[
\begin{array}{ccc}
C_{i1}^{k} & L_{i1}^{k} & Q_{i1}^{k} \\
C_{i2}^{k} & L_{i2}^{k} & Q_{i2}^{k} \\
... & ... & ... \\
C_{ij}^{k} & L_{ij}^{k} & Q_{ij}^{k} \\
... & ... & ... \\
C_{in}^{k} & L_{im}^{k} & Q_{im}^{k}
\end{array}
\right]
\end{array}
\tag{4}
$$

where

$$C_{ij}^{k} = (\overline{c_i^k} - Cost_{ij}^{k})\,/\,s_{c_i^k} \tag{5}$$

$$L_{ij}^{k} = (\overline{l_i^k} - Leadtime_{ij}^{k})\,/\,s_{l_i^k} \tag{6}$$

$$Q_{ij}^{k} = (Quality_{ij}^{k} - \overline{q_i^k})\,/\,s_{q_i^k} \tag{7}$$

C_{ij}^{k} : utility on cost for U_{ij} to process order k

L_{ij}^{k} : utility on lead time for U_{ij} to process order k

Q_{ij}^{k} : utility on quality for U_{ij} to process order k

$\overline{c_i^k}, \overline{l_i^k}, \overline{q_i^k}$: average utility of all the bid on cost, lead time, quality

$s_{c_i^k}, s_{l_i^k}, s_{q_i^k}$: standard deviation of all the bid on cost, lead time, quality

Five strategies are defined as selection mechanism using the payoff matrix in (4).
- method 1: cost minimisation

$$\max_{j=1,2,...,\,n} = C_{ij}^{k} \tag{8}$$

- method 2: lead time minimisation

$$\max_{j=1,2,...,\,n} = L_{ij}^{k} \tag{9}$$

- method 3: quality maximisation

$$\max_{j=1,2,\dots,n} = Q_{ij}^k \tag{10}$$

- method 4: total utility maximisation

$$\max_{j=1,2,\dots,n} = C_{ij}^k + L_{ij}^k + Q_{ij}^k \tag{11}$$

- method 5: max-min strategy

$$\max_{j=1,2,\dots,n} \min\{\ C_{ij}^k, L_{ij}^k, Q_{ij}^k\} \tag{12}$$

Method 5 corresponds to "max-min strategy" in game theory, which has been proved to conduct Nash equilibrium solution by min-max theorem, if the game is in zero-sum situation like our formulation shown in (4).

2.4 Reinforcement Learning

In non-sequential tasks, a unit must learn a mapping of situations to actions that maximises the expected immediate payoff in general. Putting in a better context, units have their actions-to-reward manual by their side. Sequential tasks are more difficult because actions selected by units may influence its future situations and thus its future payoffs. In this case, the unit interacts with the environment over an extended period of time, and it needs to evaluate its actions on the basis of their long-term consequences. This involves a credit assignment problem, i.e. a whole sequence of actions takes place before long term consequences are known. This would be difficult because actions in a sequence may have different values with respect to the consequences.

Q-learning is a recent form of reinforcement learning algorithm that does not need a model of its environment and can be used on-line (Sutton, 1998). Therefore, it is very suited for repeated games against an unknown opponent. Q-learning algorithm works by estimating the values of state-action pairs. The value $Q(s,a)$ is defined to be the expected discounted sum of future payoffs obtained by taking action a from state s and following an optimal policy thereafter. Once these values have been learned, the optimal action from any state is the one with the highest Q-value. After being initialised to arbitrary numbers, Q-values are estimated on the basis of experience as follows:

From the current state s, select an action a. This will cause a receipt of an immediate payoff r, and arrival at a next state s'.

We update $Q(s,a)$ based upon this experience as follows:

$$Q(s,a) = (1 - \alpha)Q(s,a) + \alpha(r + \gamma \max Q(s',b)) \tag{13}$$

where α is the learning rate and $0 < \gamma < 1$ is the discount factor.

This algorithm is guaranteed to converge to the correct Q-values with the probability one if the environment is stationary and depends on the current state and the action taken in it; called Markovian, a lookup table is used to store the Q-values, every state-action pair continues to be visited, and the learning rate is decreased appropriately over time. This exploration strategy does not specify which action to select at each step. Experiments with Q-learning agent have been done in the past with favourable results.

It is quite natural to assume enterprise units have an adaptive behaviour in terms of negotiation process. We especially focus on the bid creation mechanism in CNP, and introduce the reinforcement learning algorithm into it. Contractor's behaviour in

each enterprise unit is improved by its experience so as to dispatch a better offer to attain the rewards from managers. The state and the action for Q-Learning, are defined in VE model as follows:

State: State is defined by two kinds of attributes, cost and lead time, because quality is automatically defined by these parameters in (7). State domain is divided by N^3 space with $E_{ij}^k, D_{ij}^k, P_{ij}^k$, and unit recognises its state in the state domain map. (Figure2)
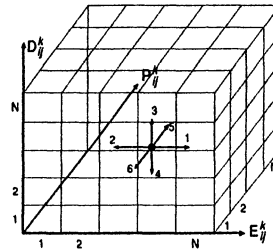

Figure 2 State Domain Map

Action: Seven types of actions are prepared to respond adaptively to the state as follows:

Action 1: Cost: fix ∩ Lead time: reduce
Action 2: Cost: fix ∩ Quality: increase
Action 3: Lead time: fix ∩ Cost: reduce
Action 4: Lead time: fix ∩ Quality: increase
Action 5: Quality: fix ∩ Cost: reduce
Action 6: Quality: fix ∩ Lead time: reduce
Action 7: no changes

3. EXPERIMENTAL RESULTS

3.1 Simulation Model

A 3-layered VE model for computer simulation was developed to clarify VE formulation dynamism with the proposed negotiation mechanism. Each layer consists of 5 enterprises in this simulation model described in Figure 1.

Simulation parameters are shown in Table 1. All the results are the average of 500 trials in each simulation scenario.

Table 1 Simulation parameters

m	n	N	E	D	P	λ	μ	ν
3	5	1	5-15*	5-15*	5-15*	300	5	0.1

* followed by uniformed random distribution

3.2 Simulation Results

Simulation results in terms of negotiation attributes and utilities are shown in Figure 3. Here we don't activate Q-Learning algorithm intentionally to analyse basic

dynamism of our VE model. We summarise the characteristics of each method as follows:

Method 1: Cost minimisation
  Since the negotiation amongst enterprises is cost-oriented in this method, cost parameter is the best of all the methods in Ave. It has been observed that all the utilities are small to minimise total cost in this method.
Method 2: Lead time minimisation
  Lead time-oriented method naturally conducts the minimal LeadTime in Ave. Cost parameter is not good, because Leadtime and E_{ij}^k is in trade-off relations.

Method 3: Quality maximisation
  It is obvious that quality maximisation strategy caused the worst in Cost, and enterprises don't pay any attention to LeadTime shown in (2).
Method 4: Total utility maximisation
  Generally the result is moderate in the balance amongst 3 parameters by trying to maximise total utility. It has been observed that they are relatively better in its LeadTime and Quality, but worse in its Cost.
Method 5: Max-min strategy
  Acquired result is completely well-balanced in Ave. It has also been confirmed that this strategy conducts minimal in St. Dev., and their negotiation is stable and robust, because each agent tried to minimise the risk with min-max theory.
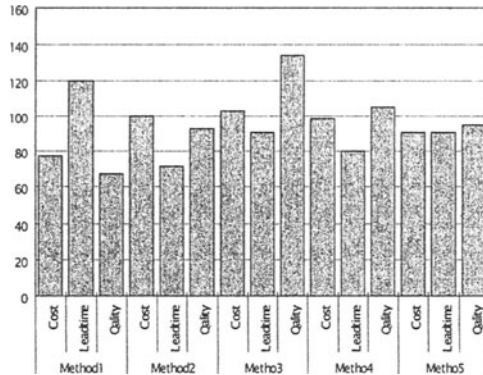

Figure 3 Method Comparison

3.3 Learning effects
Here we activate Q-Learning algorithm, and Q-Learning parameters are as follows:
  $\alpha : 0.5, \quad \gamma : 0.2$
  the number of tasks for initial learning   : 2,000
Simulation results in terms of negotiation attributes and utilities are shown in Table 2. All the results are also shown in the average (Ave.) and the standard distribution (St. Dev.) in Table 2.

Table 2 Q-Learning effects

	Conventional		Q-Learning	
	Ave.	St. Dev.	Ave.	St. Dev.
Cost	1509	124.80	1496	100.60

Lead time	59.48	4.36	59.38	3.76
Quality	298.20	30.62	295.50	24.81

It is obvious that the standard deviation is smaller in Q-Learning based negotiation compared with the conventional negotiations (without learning algorithm), although there isn't so much difference in the average.

A manager in a unit has to select one contractor amongst several bids at least by relative evaluations in our negotiation mechanism. Therefore it is quite natural that the randomness of the attributes in bid causes large standard deviation in their compromised trade. On the other hand, contractor's bidding strategy based on Q-Learning algorithm produces more acceptable attributes in their bid, and that concludes little randomness with small standard deviation. It is obvious that Q-Learning based bidding strategy leads stable trading between VE units.

Although our investigations have clearly not been exhaustive, it is already apparent that the agent behaviours with learning algorithm have great influence on autonomously formulated VE structure as a basic study. Proposed game theoretic formulation with Q-Learning in multi-agent paradigm is quite reasonable to realise rational negotiation process amongst enterprises.

4. CONCLUSIONS

In this paper, we focused on negotiation process in VE formulation as a basic research. Each enterprise in VE was defined as agent with multi-utilities and a framework of multi-agent programming with game theoretic methodology and Q-Learning algorithm was newly proposed as negotiation algorithm. Simulation results have proved that the proposed game theoretic formulation on agent decision mechanism with multi-agent paradigm is quite reasonable to facilitate rational negotiation process amongst enterprises.

5. REFERENCES

1. Camarinha-Matos, L. M. et al., The virtual enterprise concept, Infrastructures for virtual enterprises, Kluwer academic publishers, Boston, pp.3-14, 1999.
2. Fisher M. L., Making supply meet demand in uncertain world, Harvard Business Review, May/Jun, 1994.
3. Kaihara, T. and S. Fujii, A proposal on negotiation methodology in VE, Collaborative Business Ecosystems and Virtual Enterprises, Kluwer Academic Publishers, Boston, pp125-132, 2002a.
4. Kaihara, T. and S. Fujii, IT based Virtual Enterprise Coalition Strategy for Agile Manufacturing Environment, Proc. of the 35th CIRP Int. Seminar on Manufacturing Systems, pp32-37, 2002b.
5. Sutton, R. and A. G. Barto, Reinforcement Learning, MIT Press, Cambridge, MA, 1998.
6. Von Neumann, J. et al., Theory of Games and Economic Behavior, Princeton University Press, 1947.
7. Smith, R., The contract net protocol, IEEE Transaction on Computers, C-29, pp.1104-1113, 1980.