24

ENTERPRISE NEGOTIATION ALGORITHM WITH WALRASIAN VIRTUAL MARKET

Toshiya Kaihara[1] and Susumu Fujii[2]
[1] *Graduate School of Science and Technology, Kobe University*
[2] *Department of Computer and Systems Engineering, Kobe University*
{kaihara, fujii}@cs.kobe-u.ac.jp
JAPAN

Virtual enterprise (VE) is now recognised as one of the effective means by which SMEs can make instant improvements to their business strategies and operations. In a large-scaled VE environment, system modules represent independent business units with conflicting and competing resource requirements, and may possess localised information relevant to their tasks. To recognise this independence, we treat the modules as agents, ascribing each of them autonomy to decide how to deploy resources under their control in service of their interests. To explore the use of market mechanism for the coordination of distributed planning module in VE circumstances, we have developed a prototype environment for specifying and simulating computational markets with multi-commodity situations. It has been confirmed that the proposed algorithm successfully calculates Pareto optimal solutions in the VE resource problem by comparing an analytic approach.

1. INTRODUCTION

During the last few years the focus has shifted from factory level to enterprise level due to the increasing global presence of the companies. Virtual enterprise (VE) is now recognised as one of the effective means by which SMEs can make instant improvements to their business strategies and operations (Camarinha-Matos, 1999) (Kaihara, 1999 & 2002). This in turn can lead to increase competitive advantages. Solving resource allocation problems under autonomous VE situations, with and for distributed computing systems presents particular challenges attributable to the distributed nature of the computation. In a large-scaled VE environment, system modules represent independent business units with conflicting and competing resource requirements, and may possess localised information relevant to their tasks. To recognise this independence, we treat the modules as agents, ascribing each of them autonomy to decide how to deploy resources under their control in service of their interests.

On the other hand, market price systems constitute a well-understood class of mechanisms that provide effective decentralisation of decision making with minimal communication overhead. In a market-oriented programming approach to distributed problem solving, the optimal resource allocation for a set of computational agents is derived by computing general equilibrium (Shoven, 1992) of an artificial economy. Market mechanism can provide several advantages on resource allocation in VE:

- Markets are naturally distributed and agents make their own decisions about how to bid based on the prices and their own utilities of the goods.
- Communication is limited to the exchange of bids and process between agents and the market mechanism.

So as to facilitate the optimised VE management with e-commerce infrastructure, a sophisticated business matching mechanism is required to manage such a large-scaled environment (Kaihara, 2001a & 2001b). General equilibrium theory in microeconomics endorses Pareto optimality of the product allocation in perfect competitive market. By applying VM architecture into VE negotiation mechanism, this optimality plays quite an important role to facilitate sophisticated & efficient VE operation. In this paper we construct Walrasian type Virtual Market, that is a principal market model in microeconomics (Kreps 1990), and try to confirm the Pareto optimality in our market model by comparing the solutions with an analytic approaches, named fixed-point algorithm.

2. WALRASIAN VIRTUAL MARKET

There exists a market-oriented programming to construct a computational market (i. e. virtual market), which consists of several heterogeneous agents (Wellman, 1996) (Kaihara, 1999). Agent activities in terms of products required and supplied are defined so as to reduce an agent's decision problem to evaluate the tradeoffs of acquiring different products in the market-oriented programming. These tradeoffs are represented in terms of market prices, which define common scale of value across the various products. The problem for designers of computational markets is to specify the mechanism by which agent interactions determine prices.

Market-oriented programming is the general approach of deriving solutions to distributed resource allocation problems by computing the competitive equilibrium of an artificial economy. It involves an iterative adjustment of prices based on the reactions of the agent in the market. General concept of the negotiation mechanism in market-oriented programming is shown in Figure 1. Definitions of the virtual market are based on general equilibrium concept in perfect competitive market, and that means it satisfies a necessary condition of Walrasian type virtual market.

Supply / demand functions represent agent's willingness to sell / buy resources, respectively. They are defined as the relationship between price and quantity of the trading resource. Let $Pt(s)$ be the price of resource s at time t. $\alpha_t ms$ and $\beta_t ns$ represent the supply function of supplier m on resource s at time t and the demand function of demander n on resource s at time t, respectively. The bidding mechanism computes an equilibrium price in each separate market. It involves an iterative adjustment of prices based on reactions of agents in the market. Agent s submits supply and demand functions ($\alpha_t ms$ and $\beta_t ns$) and the auction adjusts individual prices to clear, rather than adjusting the entire price vector by some increment. The mechanism associates an auction with each distinct resource. Agents act in the market by submitting bids to auctions. In this paper bids specify a correspondence between prices and quantities of the resource that the agent offers to demand or supply as a basic study. Given bids from all interested agents, the auction derives a market-clearing price.

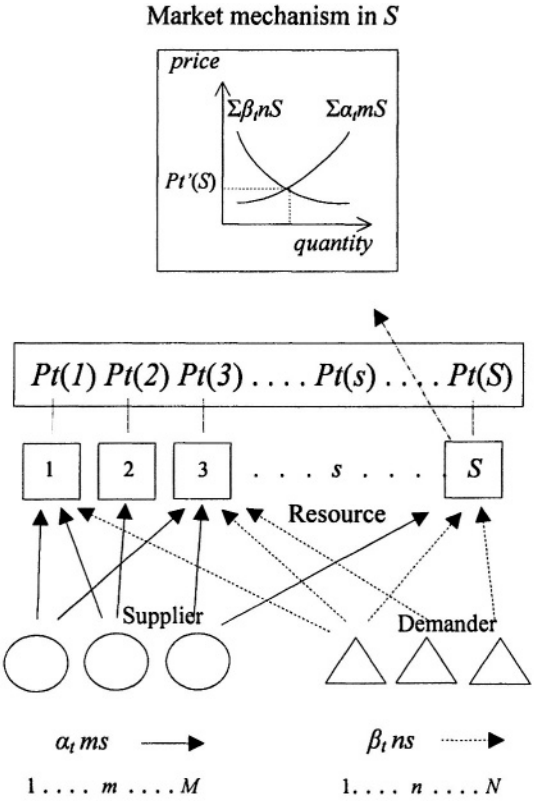


Figure 1 Negotiation mechanism

3. AGENT DEFINITIONS

We describe consumers (i.e. demanders) as c_m $(m = 1,2,\ldots,M)$, and producers (i.e. supplier) as s_n $(n = 1,2,\ldots,N)$. The number of kind of goods is assumed as I in our Walrasian VM.

3.1 Demand agent (Consumer agent)

3.1.1 Demand utility

Suppose demand agent c_m has utility function u^{c_m} , which is described with in equation (1). In this equation $x_i^{c_m}$ represents the demand quantity for resource *i*:

$$u^{c_m} = a^{c_m} \prod_{i=1}^{I} (x_i^{c_m})^{b_i^{c_m}} \tag{1}$$

where $$\sum_{i=1}^{I} b_i^{c_m} = 1 \qquad (0 < a^{c_m}, b_i^{c_m})$$

In this paper we adopt Cobb-Douglas function (Kreps 1990) as a demand function described in equation (1), because the Cobb-Douglas function is one of the primitive functions in microeconomics, which handles economical scale in the market by index constant *b*.

3.1.2 Budget

Budget of demand agent c_m is formulated by initial quantity of resource (*i*): $e_i^{c_m}$, and their price: p_i as follows:

$$B^{c_m} = \sum_{i=1}^{I} p_i e_i^{c_m} + r^{c_m} \tag{2}$$

In this equation r^{c_m} represents supplier's profit, which suppliers return to demanders under zero -profit conditions in the general equilibrium theory.

3.1.3 Bidding functions

Demand agents send their bid to their target resources in the market, and the bid is formulated as demand function. The function is obtained as the optimal solution as maximising problem of equation (1) under the constraints described in (2). The following demand function is calculated by Lagrange's method of (indeterminate) multiplier in this research.

$$x_i^{c_m}(p_i) = \frac{b_i^{c_m} B^{c_m}}{p_i} \qquad (i = 1, \ldots, I) \tag{3}$$

We assume demander c_m supplies all the initial resources into the market according to the principle of microeconomics, and demander's supply function of resource *j* is defined as follows:

$$y_j^{c_m} = e_j^{c_m} \tag{4}$$

3.2 Supply agent (Producer agent)

3.2.1 Production function

As described in demand agent definitions, Cobb-Douglas function is basic functions which handles economical scale in the market easily. In micro-economics production function is assumed to be convex function, and that means market prices are established at a predictable level in the general equilibrium theory in convex shape production function.

We also formulate production function of supply agent s_n to resource *j* as Cobb-Douglas function to satisfy the assumption, shown as equation (5). Cobb-Douglas function is defined as a convex function in $0< \beta <1$ in this equation.

$$y_j^{s_n} = \alpha^{s_n} (x_j^{s_n})^{\beta^{s_n}} \qquad (0 < \alpha^{s_n}, 0 < \beta^{s_n} < 1) \tag{5}$$

3.2.2 Profit

According to microeconomics assumption, supply agents have no initial resources. They can earn their profit π^{s_n} by producing value added resources from purchased resources. The profit function is defined as follows:

$$\pi^{s_n} = p_j y_j^{s_n} - p_i x_i^{s_n} \tag{6}$$

3.2.3 Bidding functions

Supply agents send supply functions to production resources, and demand functions to purchase resources, respectively. They maximise their profit by solving maximising problem of equation (6) under the constraint in equation (5). We also solve the problem by Lagrange's method of multiplier in this research, and obtain the following demand function and supply function in equation (7) and (8), respectively:

$$x_i^{s_n}(p_i) = \left(\frac{p_i}{\alpha^{s_n}\beta^{s_n}p_j}\right)^{\frac{1}{\beta^{s_n}-1}} \tag{7}$$

$$y_j^{s_n}(p_j) = \left(\frac{p_i^{\beta^{s_n}}}{\alpha^{s_n}(\beta^{s_n})^{\beta^{s_n}}\ p_j^{\beta^{s_n}}}\right)^{\frac{1}{\beta^{s_n}-1}} \tag{8}$$

4. ANALYTIC APPROACH

We demonstrate the proposed algorithm successfully calculates Pareto optimal solutions by comparing VM solutions with an analytic approaches, named fixed-point algorithm. We will explain this analytic approach briefly below.

Scarf showed that how to compute an approximate Walras equilibrium and proposed a general algorithm for the calculation of a fixed point of a correspondence (Scarf, 1973). This algorithm, named Scarf's algorithm, has been surprisingly efficient to find a general equilibrium that was guaranteed to converge, though does not permit a gradual improvement in the degree of approximation of the solution.

The Scarf's algorithm applies a procedure to the problem of computing a fixed point of mapping of the unit simples into itself, a mapping whose existence is established by fixed point theorem. In using Scarf's algorithm to find such a fixed point, the unit simplex is divided into a finite number of smaller simplices, each defined by *I* vertices that are each associated with a label. Each vertex is labelled with the index number of goods, which has the maximum value in market excess demand. The market excess demand of goods *i* at the vertex is defined as the next equation:

$$E_i = \left(\sum_{m=1}^{M} x_i^{c_m} + \sum_{n=1}^{N} x_i^{s_n}\right) - \left(\sum_{m=1}^{M} y_i^{c_m} + \sum_{n=1}^{N} y_i^{s_n}\right) \tag{9}$$

The algorithm operates by moving through adjacent simplices, deleting and adding vertices. The starting point and each continuation step in the algorithm are such that, prior to finding an approximate equilibria, the algorithm always considers simplices whose verticles have all but one label present, with one label appearing as a duplicate. The procedure involves deleting a vertex with a duplicate label and replacing it with a new vertex, and it then moves to an adjacent simplex. Because of the finiteness of the number of simplices, and the inability to terminate the procedure other than at a close approximation to an equilibrium, the procedure is guaranteed to find an approximation to an equilibrium.

5. EXPERIMENTAL RESULTS

Experimental 2-producer-2-consumer market is illustrated in Figure 2. In this market 2 kinds of goods, goods 1 and goods 2, are also traded.

Experimental values of each parameter (i.e. initial value set) in the model are shown in Table 1 (consumer agent) and Table 2 (producer agent). Consumer agents send their bids as supply & demand functions to both the goods homogeneously. Producer agents send their demand & supply function to goods 1 & goods 2, respectively. That means they both produce goods 2 from goods 1 in this market.

Table 1 Parameter of consumer agent

Agent	Utility function(u^{c_m})	Initial endowment $(e_1^{c_m}, e_2^{c_m})$	Initial utility
1	$1.0(x_1^{c_1})^{0.3}(x_2^{c_1})^{0.7}$	(30.0, 10.0)	13.90
2	$1.0(x_1^{c_2})^{0.5}(x_2^{c_2})^{0.5}$	(20.0, 40.0)	28.28

Table 2 Parameter of producer agent

Agent	Input	Output	Production function
1	goods 1	goods 2	$y_2^{s_1} = 2.0(x_1^{s_1})^{0.7}$
2	goods 1	goods 2	$y_2^{s_2} = 3.0(x_1^{s_2})^{0.5}$

Simulation results on price and trade changes of each goods in the Walrasian VM are shown in Figure 3, and 4, respectively. In this model we obtained equilibrium price vector $\hat{p} = (\hat{p}_1, \hat{p}_2)$ as follows:

Equilibrium price vector $\hat{p} = (0.95673, 0.98543)$

Normalised equilibrium price vector $\hat{p} = (0.49261, 0.50739)$

And final resource allocation in consumer agent and producer agent are shown in table 3 and 4, respectively.

Table 3 Resource allocation in model 3

Agent	Consumption $(x_1^{c_m}, x_2^{c_m})$	Utility
c_1	(12.67, 28.69)	22.45
c_2	(31.56, 30.64)	31.10

Table 4 Resource allocation in model 3

Agent	Consumption	Production	Profit
s_1	3.39	4.70	1.39
s_2	2.39	4.63	2.28

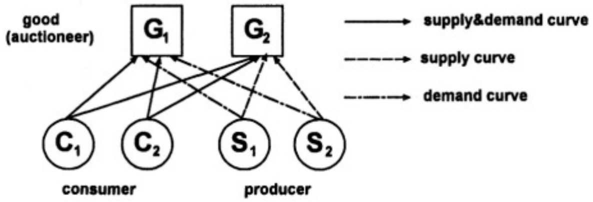
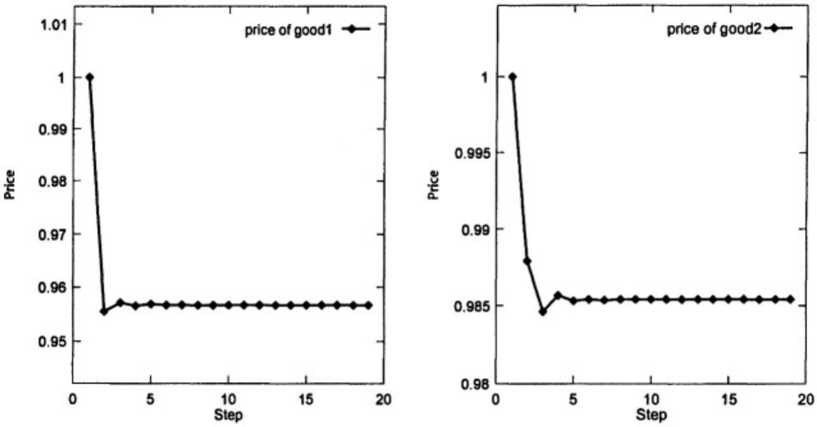


Figure 2 - 2-producer-2-consumer market

It is observed that the total consumer's utility is increased in table 7 compared with table 2, because the producer's profit is returned to consumer agents in this market under the zero-profit conditions in the general equilibrium theory. The utility of agent c_1 is increased by 17.2 % especially. That is because of the agent's preference to goods 2 as well as the zero-profit conditions. In this market all the producer agents supply goods 2, and that increases the utility of agent c_1. On the other hand, the utility of agent c_2 is slightly decreased, because it becomes slightly difficult to get both goods due to the stronger supply flow from producer agents to agent c_1.

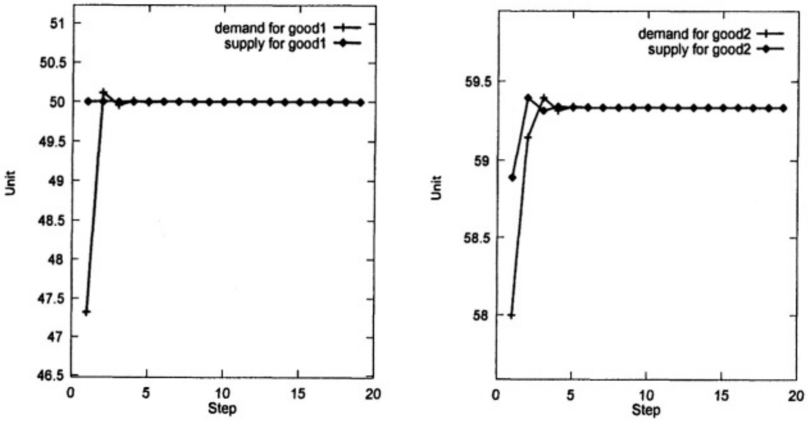
We applied fixed point algorithm in this market model to confirm Pareto optimality of the VM solutions. We calculated a fixed point of mapping of the unit simples into itself, a mapping whose existence is established by fixed point theorem. In using Scarf's algorithm to find such a fixed point, the unit simplex is divided into a finite number of smaller simplices (i.e. grid size). In this paper the grid size is set to 100,000 for the precise comparison.



(a) Price of goods 1

(b) Price of goods 2

Figure 3 Price convergence



(a) Trade in goods 1

(b) Trade in goods 2

Figure 4 Trade convergence

We obtained equilibrium price vector $\hat{p} = (\hat{p}_1, \hat{p}_2)$ by the fixed point algorithm as follows:

Equilibrium price vector $\hat{p} = (0.49262, 0.50739)$

It is obvious that the equilibrium price set obtained by Walrasian VM is almost equivalent to the one from the fixed point algorithm, and that means VM solutions have been confirmed to be converged into Pareto optimal. The small difference is caused by the grid size of the fixed point algorithm.

We compared the calculation time between VM approach and fixed point algorithm. CPU consumption time (second) for each approach in this market model is as follows:

VM : 0.0017 Fixed point algorithm: 12.025

VM approach is obviously more than 7,000 times as fast as the analytic approach in this model. It has also been proved that the proposed VM based approach is much more practical in terms of calculation time to obtain Pareto optimal solutions in resource allocation problems.

By the computer simulation, we have confirmed that Walrasian VM takes advantage of the market analogy into resource allocation problem, and that leads to effective search of Pareto optimal solution for supply chain management.

5. CONCLUSIONS

We newly proposed a Walrasian Virtual Market (VM) approach for VE in this paper. Firstly we mentioned our general concept to apply VM into negotiation malgorithm for VE, and explained general idea of Walrasian market model in economics. Then one of multi-agent programming, named market-oriented programming, was focused and explained its negotiation process in the Walrasian market. After a brief explanation of several analytical approaches, we defined agent behaviour based on Walrasian market-oriented programming. As a basic study, we analysed its Pareto optimality by computer simulation experiments, and it has been confirmed that our approach is efficient both in Pareto optimality and calculation performance.

6. 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. Kaihara, T., Supply chain management based on market Mechanism in virtual enterprise, Infrastructures for Virtual Enterprises, Kluwer Academic Publishers, Boston, pp.399-408, 1999.
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, 2002.
4. Kaihara T., Supply Chain Management with Market Economics, International Journal of Production Economics, Elsevier Science, Vol. 73, Issue 1, pp.5-14., 2001a.
5. Kaihara T., Multi-Agent based Supply Chain Management with Dynamic Environment, Proceedings of International Conference on Production Research-16 (CD-ROM), 2001b.
6. Kreps, D. M., A Course in Microeconomic Theory. Harvester Wheatsheaf, New York, 1990.
7. Scarf, H. E., The computing of Economics Equilibria, New Heaven, Yale University Press, 1973.
8. Shoven J. B. and Whalley J., Applying General Equilibrium, Cambridge University Press, 1992.
9. Wellman M. P., A Market-Oriented Program-ming Environment and its Application to Distributed Multi-commodity Flow Problems, ICMAS-96, pp385-392, 1996.