

Implementation of a Multiagent System for Competitive Environments on the WWW

Hiroyuki Onjo and Behrouz H. Far

Department of Information and Computer Sciences, Saitama University, Urawa, 338-0825 Japan

SUMMARY

Agents have been receiving a lot of attention as software which behaves autonomously and efficiently on a network and supports human social interactions. Various agent systems have already been proposed. A conventional agent-based system is based on the assumption that all interacting agents are trustworthy. In practice, however, there are many instances in which agents must compete in an environment in which the other agents cannot be trusted. Thus, in this paper the authors discuss the development of an electronic commerce system which uses agents that behave competitively. The agents in this system can determine autonomously the level of uncertainty in a transaction, then make decisions flexibly. In addition, the authors argue for the validity of their system using operational examples in which their system is used in various market scenarios. © 2002 Wiley Periodicals, Inc. Syst Comp Jpn, 33(14): 34–49, 2002; Published online in Wiley InterScience (www.interscience.wiley.com). DOI 10.1002/scj.10032

Key words: multiagent system; competition; uncertainty management; decision support; electronic commerce.

1. Introduction

In recent years the rapid growth of the World Wide Web has led to the creation of an environment which allows

individuals to freely send and receive content across the world. This communications network continues to evolve toward a virtual society. Along with these developments, a new social framework including electronic commerce (e-commerce) and CALS are creating a networked environment. Given this background, in recent years living in a networked society in the midst of what could be called the age of mega-competition demands a very high level of creative and strategic performance in individuals, and requires the ability to process tremendous amounts of information and then find the next strategy. However, in this age which is changing in unprecedentedly complicated ways, it is not easy to respond accurately with only one's own decision-making ability as one is assaulted from various sides. Thus, even in the world of computer software there has been born a movement toward making our networked society function smoothly by creating a flexible, freely distributed system on the network just as exists in human society. Interest in software that supports human social activity through behavior which is autonomous and efficient as is seen in an agent [14] has risen. A multiagent system refers to a system which executes several tasks while having multiple agents work together in various ways.

Fundamentally, a multiagent system was developed as a means to execute tasks *cooperatively* based on the idea that agents can trust each other [14]. However, in recent years agents have taken on the role of smart representatives of people in open environments on the Internet, and so *competition* between smart agents which were developed with different and distinctive behavioral policies has been occurring frequently. In other words, the development of agents as essentially cooperative has become the problem.

© 2002 Wiley Periodicals, Inc.

Therefore, in order to meet the needs of the Internet age, the development of a multiagent system which allows for competitive behavior is essential. However, a multiagent system which allows for competitive behavior has yet to be developed.

Given this situation, the authors proposed [8] a multiagent competitive approach based on the uncertainty principle. This approach was an application of game theory and conceptual networks, and was created in order to have agents make optimally strategic decisions while they predicted the behavior of their competitors autonomously. However, concrete applications of the authors' proposed competitive approach, as well as development methods, have yet to be discussed. Thus, in this paper the authors discuss the development of an agent-based system which supports sales competition in e-commerce as an applied system using this competition approach.

2. Multiagent System Model

In this paper the authors develop an agent system which supports competitive behavior in a multiagent system [3] for e-commerce.

In the e-commerce field, an agent system which supports competitive behavior is desirable for the following reasons: (1) Environments in which agents can trust and cooperate with each other and in which they cannot trust and instead compete with each other are intermixed; (2) because of the WWW's open nature as an environment for social activity, agents using different and distinctive operational policies exist, and competition frequently occurs; (3) because the market is expanding, the creation of various systems which include competitive environments is desirable.

The configuration of an e-commerce agent society, the focus of this paper, is shown in Fig. 1. This system was built for the purpose of adding the flexibility of human society to an e-commerce society. The business of various agents in the system can be seen as similar to that found in human society. In this system, human society is modeled, and seven knowledgeable agents—*Customer*, *Search*, *Catalog*, *Dealer*, *Manufacturer*, *Banker*, and *Delivery*—were created. Each agent executes specialized tasks in e-commerce while maintaining mutual interactions based on relationships with other agents such as cooperation and competition. The specific role of each agent and the relationships among the agents are discussed in Ref. 3.

In this paper the authors will address the competition of Dealer Agents which support customers and retailers of online shops on the Web. In the flow of this system, market competition occurs in the form of handling clients among Dealer Agents when negotiations among multiple Dealer Agents occur via the Customer Agents which support cus-

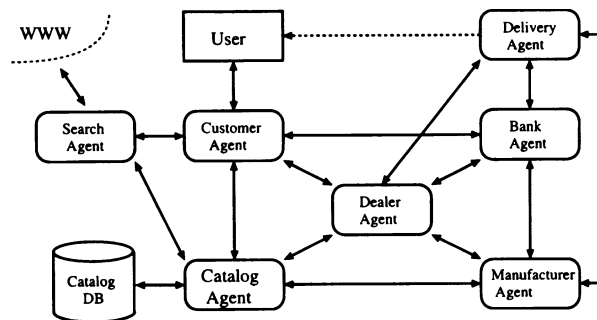


Fig. 1. Society of electronic commerce agents.

tomers when users purchase products. At this point, each Dealer Agent independently selects the sales strategy which will provide the best service for a given customer, then extends this sales competition by providing it to clients. A common form of a sales strategy in e-commerce is to provide services for product pricing which combines consideration of customer demand (e.g., high-quality products are better despite the price) and value added (e.g., discounts, sale items, delivery service, delivery date) [5]. As for the consideration of customer demand, methods such as negotiations using debate between buyer agents and seller agents [9] as well as learning user preferences using conceptual networks [3] have been proposed. In this paper the authors thus focus on a method to determine value added efficiently.

What is important when determining a value added strategy (hereafter simply called a strategy) is strategic analysis of the tendencies of competitors. This is because improving the trade-off between “implementing services which appeal to customers” and “decreasing costs” is important due to the general costs of selection strategies, even though offering services which are more appealing than a competitor's is also important when considering strategies for sales competition. In addition, from the standpoint of economics, in a competitive environment composed of several decision-makers with confronting profit and loss, the magnitude of the utility obtained as a result of a certain decision depends on the behavior of competitors, and as a result, obtaining stable profit requires coming up with a strategy that considers not only one's own situation but also the competitor's actions [16]. In e-commerce, worldwide businesses that exceed national boundaries represent one gigantic market, and so compared to business transactions in a human society in which a dominant position can be easily obtained within a confined community, the importance of strategic actions vis-à-vis competitors is all the greater [1]. Based on the above, Fig. 2 illustrates the process of market competition for a Dealer Agent and two parties.

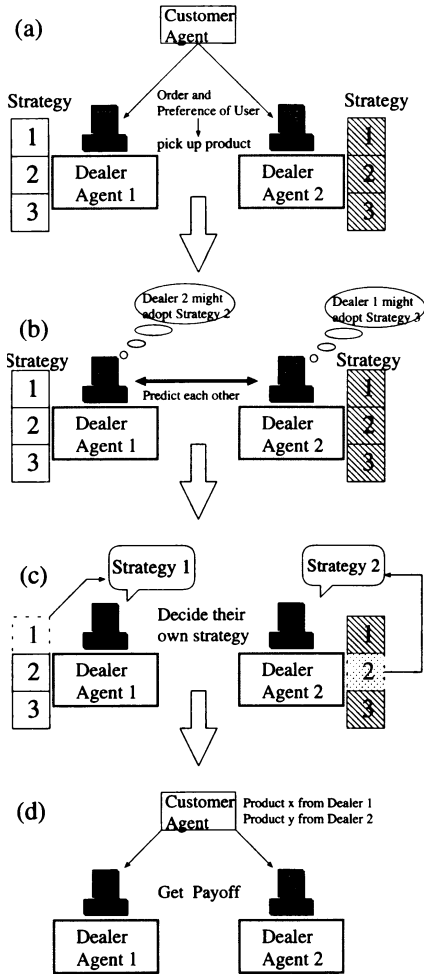


Fig. 2. Process of market competition.

First, pick-up is performed for a product whose customer demand can be considered by obtaining information on product orders and customer preferences via the Customer Agent. This function has already been implemented in prior research [3] [Fig. 2(a)]. Next, in Step (b) each competitor predicts which strategy the other will use. Here, Strategies 1 to 3 exist as possible choices. In the example in Fig. 2, Dealer Agent 1 predicts that its competitor will pick Strategy 2, and Dealer Agent 2 predicts that its competitor will pick Strategy 3. Then, in Step (c), the Dealer Agent selects the best opposing strategy based on its prediction. Here, Dealer Agent 1 selects Strategy 1, and Dealer Agent 2 selects Strategy 2. In this example, in Step (b) Dealer Agent 1 was able to make an accurate prediction for its competitor's behavior, but Dealer Agent 2 made a mistake in its prediction for whatever reason. In the final Step (d), each Dealer Agent gets its payoff in the form of creating sales. For instance, if the customer gets Product x from Dealer Agent 1 and Product y from Dealer Agent 2, then both get a payoff.

The main point in this paper is the development of an agent which has a function to determine an optimal strategy while predicting the behavior of its competitors independently, as described above. Here, the problem is that because the competitor hides knowledge of what its own strategy might be, predicting its behavior is uncertain. In other words, handling uncertainty becomes a technical issue. In this paper, the authors take two approaches: (1) implementation of a rational decision-making function in an uncertain world and (2) implementation of a function for a Dealer Agent to ease uncertainty while experiencing competition.

3. The Multiagent Competition Model [8]

3.1. Modeling a multiagent competitive environment

As described in the previous section, when competing with the other Dealer Agents, analysis of the behavior of competitors is important. In this research the authors separate Dealer Agents for which these preferences are known from those that are not, then view the former using game theory, and the latter as "natural" in that they behave randomly, assuming their strategy to be a "natural state." The authors then formalize this by bringing it together and addressing uncertainty. A competitive environment Γ which includes uncertainty will be based on game theory, and will be defined as follows.

$$\Gamma = (D, N, X, \{A_i, \succeq_i, P_i\}_{i \in D}) \quad (1)$$

Here the elements are as follows. D : the set of Dealer Agents; N : the set of natural states (competitor behavior); X : the set of results ($A_1 \times \dots \times A_n \times N \rightarrow X$); A_i : the set of behaviors (strategies) for Dealer Agent i ; \succeq_i : preferences for X for Dealer Agent i [for this, the base utility function $u(x)$ exists for the order of preferences for the base transitivity]; P_i : the breakdown of information for Dealer Agent i with respect to the set N . Figure 3 shows an example of the

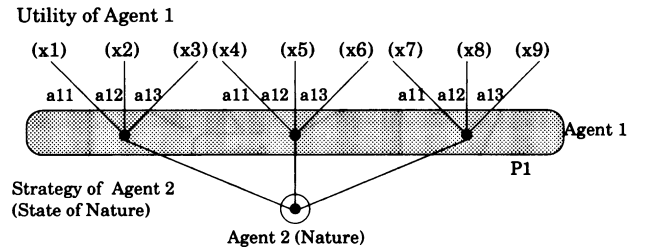


Fig. 3. Competitive environment represented by extensive form.

extensive form for game theory for competition among Dealer Agents based on the above form. In Fig. 3, in one-to-one competition, each Dealer Agent has three strategies, and Dealer Agent 1 is assumed to be in a state in which the preferences of Dealer Agent 2 are unknown. As a result, Dealer Agent 1 assumes Dealer Agent 2 to be natural. Here, P_1 is the information breakdown for Dealer Agent 1. What strategy the competitor will take is unknown. In other words, which of the left, center, or right paths in P_1 will be taken is unknown, and so one of the three may be selected. At this point, Dealer Agent 1 allocates a belief (Belief; probability estimate) which does not oppose probability laws for the behavior of Dealer Agent 2, then selects a strategy to optimize expected utility. All Dealer Agents engage in this behavior, and so a social equilibrium point is reached [16]. The problem here is that the Dealer Agent somehow allocates belief on its own. Thus, the authors propose a method in which information uncertainty is divided into three levels, then the most rational belief for each case is assigned.

3.2. A method for decision-making for competitive strategies for dealer agents

3.2.1. Information uncertainty and decision-making policy

When making decisions, the extent of information uncertainty has a great impact. In this paper the authors divide information uncertainty into three levels, then assign belief using the following policy.

- *Level 1:* Decision-making based on certainty

When the behavior of the competitor can be specified 100%. In this instance, conventional game theory analysis can be used as is.

- *Level 2:* Decision-making involving risk

Although the behavior of the competitor cannot be specified, the probability of it happening is known. In this instance, a known probability distribution is assigned as a belief, then the alternative which optimizes the expected utility is selected as a strategy. In addition, the use of the risk management method proposed in Section 3.2.2 allows for flexible and rational decision-making based on a conventional system.

- *Level 3:* Decision-making involving uncertainty

When the probability of the competitor's behavior cannot even be specified. In this instance, because there is no information to make a judgment with, belief must be

assigned using a standard other than probability. In this paper, the authors propose a decision-making method in Section 3.2.3 based on the degree of optimism of the user.

3.2.2. A decision-making method based on risk management

In general, in a decision-making problem which accompanies risk as in Level 2, each Dealer Agent takes a known probability as a belief, then acts to maximize the expected utility. A utility function is needed to determine the action of each Dealer Agent. In general, the utility function for a decision problem which accompanies risk is determined by taking the expected value of the net profit “(sale price less purchase price of a product) \times sales volume — necessary costs for adopting the strategy” when using a particular strategy. If this theory is followed, two strategies with the same expected value cannot be distinguished from each other. However, this decision-making method cannot necessarily be considered rational. This is because when the distribution is large even if the expected values are equal, the risk of loss grows, and impacts effective selection [15]. Therefore, the user's attitude toward risk with respect to uncertainty must be considered when maintaining good selection for the Dealer Agent. The attitude toward risk can be classified as follows [15].

- risk prone: oriented toward high risk, high return
- risk aversion: oriented toward low risk, low return
- risk neutral: cannot distinguish the two above

The creation of a utility function is essential in order to have preferences which reflect the attitude of the Dealer Agent toward risk. A utility function which considers the attitude toward risk can be conceived by using the precise definition of optimism in the field of economics. However, this approach is inappropriate as it is too complex to handle a Dealer Agent flexibly. Thus, the authors address this issue using a heuristic function. This paper focuses on the fact that the magnitude of a risk depends on the magnitude of the distribution. If the expected value is $E(x, p)$, and the magnitude of the distribution is $V(x, p)$, then the utility function can be defined as

$$u(x) = E(x, p) - \eta V(x, p) \quad (2)$$

Here, η is an integer, and is the weighted coefficient for comfort or aversion of risk. When η is positive, the utility value falls as the magnitude grows larger and larger (as the risk becomes greater), and so risk is avoided; if η is negative, then the utility value falls as the distribution becomes larger, and so risk is acceptable; when η is zero, the expected value is the utility function, and so risk is neither embraced nor avoided. The advantage of this approach is

that preferences which reflect the attitude toward risk can be created flexibly and readily merely by passing to the Dealer Agent a number for the attitude a user has toward risk. In addition, when several selections can be made from a set of strategies, risk management can be achieved by combining them. For instance, if Strategies 1 and 2 are combined, if \tilde{x} is the probability variable for Strategy 1 in accordance with probability p , \tilde{y} is that for Strategy 2 in accordance with probability q , the expected values are $E(\tilde{x})$ and $E(\tilde{y})$, and the distributions are $V(\tilde{x})$ and $V(\tilde{y})$, then the covariance for \tilde{x} and \tilde{y} is $\text{Cov}(\tilde{x}, \tilde{y})$. If the allocation for a usable resource is α for Strategy 1 and $(1 - \alpha)$ for Strategy 2, then the allocation α^* for the resource which optimizes Eq. (2) is

$$\alpha^* = \frac{E(\tilde{x}) - E(\tilde{y}) + 2\eta(V(\tilde{y}) - \text{Cov}(\tilde{x}, \tilde{y}))}{2\eta(V(\tilde{x}) + V(\tilde{y}) - 2\text{Cov}(\tilde{x}, \tilde{y}))} \quad (3)$$

and the allocation of α^* to the resource for Strategy 1 and of $(1 - \alpha^*)$ to the resource for Strategy 2 can be adopted as a strategy. This method of managing risk by combining strategies with different distributions as above is known as the Portfolio Method in economics [16]. The use of the approach in this paper allows for risk management using a simple representation, and as a result, can also be easily implemented in an artificial Dealer Agent.

3.2.3. Decision-making involving uncertainty

When decision-making is accompanied by uncertainty, belief must be assigned using a mathematical model which does not involve probability. Here, according to behaviorists, when humans do not know the probabilities involving uncertainty, it is known that they will act based on their degree of optimism. As a result, in this paper the authors propose a method to quantify a user's degree of optimism, and then apply it to the decision-making function of the Dealer Agent.

Ordered Weighted Averaging (OWA) operator described in Ref. 12 can be applied to the quantification of degree of optimism. The OWA operator averages the set of natural states as the function F , then uses an n -dimensional vector which does not oppose probability rules:

$$\mathbf{W} = [w_1, w_2, \dots, w_n] \quad (4)$$

and for the values a_1, \dots, a_n ,

$$F(a_1, \dots, a_n) = \sum_j w_j b_j \quad (5)$$

with the characteristics which satisfy (b_j being the j -th largest element among the set a_1, \dots, a_n). The weighting parameter \mathbf{W} in the OWA operator can be interpreted in various ways. The probability distribution for a similar event, in a natural state, in other words the probability that

the optimal solution for j will occur, is known mathematically [13]. In this instance, the pseudo-probability is not assigned to each ω , but to the order of preferences for the results.

Here, the problem becomes how to determine the weighted vector \mathbf{W} . One method is to have the human counterpart of a Dealer Agent make an appropriate decision. In this method, the Dealer Agent cannot make a decision on its own. Thus, in this paper \mathbf{W} is found automatically using the degree of optimism of the Dealer Agent. If $\text{Opt}(W)$ is the degree of optimism for the Dealer Agent,

$$\text{Opt}(W) = \sum_j w_j (n - j) / (n - 1) \quad (6)$$

is known to be the definition [12]. Thus, \mathbf{W} can be found using the following linear planning equation applying the entropy maximization principle:

$$\max - \sum_j w_j \log_2 w_j \quad (7)$$

Subject to

$$\text{Opt}(W) = \sum_j w_j (n - j) / (n - 1)$$

$$\text{Opt}(W) \in [0, 1]$$

$$\sum_j w_j = 1$$

$$w_j \geq 0 \quad \text{for } j = 1, \dots, n$$

Using this method has the advantage of allowing for a decision-making function which reflects the psychological state of the user as represented by the Dealer Agent.

By considering the value of the OWA operator obtained using the proposed method as the pseudo-probability distribution for the behavior of the competitor, the belief of the Dealer Agent (not violating probability law) can be derived even when the probability distribution for the behavior of the competitor is unknown. In this case, the Dealer Agent can make a decision by calculating the expected utility using its belief as based on the derived degree of optimism.

3.3. Updating belief based on probability estimates

When the method in the previous section is used, rational decision-making can be performed while managing the risk related to the competitor's behavior. However, the quality of decision-making may fall because the behav-

ior of the competitor remains uncertain. Therefore, in this paper the authors attempted to address uncertainty by analyzing the behavior of the competitor. Specifically, the authors observe the changes in the state of the environment, and then created a function to respond to the state changes by updating belief based on the changes.

Strategic analysis much deeper than that for the degree of optimism in the previous section is needed in order to update belief with respect to the behavior of the competitor. In this paper the authors create an analysis function for the Dealer Agent by using a dynamic belief network (DBN) to describe a game theory analysis method which can respond dynamically to strategic analysis. A DBN can represent dynamic state changes compactly, and so is compatible with game theory.

Figure 4 shows the general structure of a DBN. A DBN is composed of a knowledge node which is aware of the external environment and a state node which describes the state. The agent predicts the current state by monitoring signals which are an information source in order to know the current state based on a sensor model. Then, based on a state transition model, belief is updated using estimates of the probability distribution for the next state when a particular strategy is adopted. In this paper, the game theory analysis method is mapped to the DBN, then the knowledge of the Dealer Agent is modeled by structuring a sensor model and a state change model. Specifically, the following state variable sets necessary for strategic analysis are considered.

- Competitor Dealer Agent Type (T): The preferences of the competitor must be known in order to predict the behavior of the competitor Dealer Agent. This type is a variable which determines the form of the competitor’s preferences, in other words, the utility function. Within the framework of the authors’ research, this is determined by the risk avoidance variable η in Section 3.2.2. For instance, if there are three levels to this type, then $T = \{-1, 0, 1\}$.
- Knowledge Hierarchy Structure (K): Merely knowing the competitor’s type is insufficient for

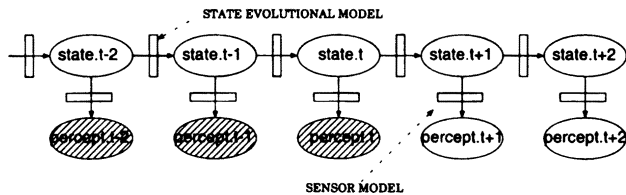


Fig. 4. General structure of a dynamic belief network.

strategic decision-making. A knowledge hierarchy structure in which whether or not the type is mutually known, and whether or not the competitors know that they know is important [16]. This is because in the same way that you predict the behavior of your competitor your competitor also predicts your behavior, and so your competitor’s behavior depends not only on its type, but also on its prediction of your type. When there are two competing agents, the knowledge hierarchy structure includes four possibilities:

- (1) Whether or not Dealer Agent 1 knows the type of Dealer Agent 2
- (2) Whether or not Dealer Agent 2 knows the type of Dealer Agent 1
- (3) Whether or not Dealer Agent 1 knows that “Dealer Agent 2 knows its own type”
- (4) Whether or not Dealer Agent 2 knows that “Dealer Agent 1 knows its own type”

This is modeled as the four states θ_1 to θ_4 in Table 1. Note that although the knowledge hierarchy structure includes four states, the information breakdown P_1 and P_2 for each Dealer Agent is

$$P_1 = \{\{\theta_1, \theta_2\}, \{\theta_3, \theta_4\}\},$$

$$P_2 = \{\{\theta_1, \theta_4\}, \{\theta_2, \theta_3\}\}$$

As a result, for the purpose of calculation only the two states in the information breakdown need to be predicted.

- Belief of Belief (B): A hierarchy structure in which whether or not the knowledge hierarchy structure is mutually known is modeled. In order to further deepen the strategic analysis, if the competitor does not know you, then “what type the competitor mistakenly thinks you are” must be analyzed. Here, Belief of Belief refers to the belief regarding how “the competitor has predicted your type (or

Table 1. Knowledge hierarchy structure

	θ_1	θ_2	θ_3	θ_4
Dealer 2 knows Dealer 1	No	Yes	Yes	No
Dealer 1 knows Dealer 2	Yes	Yes	No	No
State	θ_1	θ_2	θ_3	θ_4

mistaken it)” for each Dealer Agent. If your type consists of three categories, then $B = \{-1, 0, 1\}$.

- Competitor Strategy (N): The competitor’s strategy. If the competitor has three strategies, then $N = \{\omega_1, \omega_2, \omega_3\}$.
- Signal Presence (I): Whether or not an information signal about the competitor’s behavior is present. This is expressed as a binary value. In other words, $I = \{T, F\}$.

Figure 5 shows an example of the sensor model which models these elements. In the sensor model, each Dealer Agent monitors information source signals related to the behavior of competitors from the world of the Internet. The signals cannot always be monitored, however. Thus, the phenomenon O is added to the model. O is a phenomenon for which the Dealer Agent obtains a signal. The sensor model can be created by adding to the model the conditional probability table (CPT) which indicates the probability of the phenomenon O occurring. In the example in Fig. 5, the signal is found to have a probability of 0.5. Then, when the phenomenon O occurs, the state I is certain to be true. In general, a probability $\Pr(O^i|S^{t-1})$ for which the Dealer Agent monitors the signal in state S^{t-1} (state S is determined by a combination of each state variable) is obtained in the sensor model. Next, Fig. 6 shows an example* of the state change evolution model for competition between two agents. In this paper, it is assumed as shared knowledge that the set of mutual competition and the respective types do not change. The direction of the arrows represents the effect relationship for state evolution when a Dealer Agent adopts a particular strategy. The CPT is added in at each node. A competitor’s behavior is affected by the preferences (type) of the competitor, and how the competitor has predicted one’s own type (Belief of Belief). In other words, N^t is affected by T^{t-1} and B^t (because it is assumed that the type does not change). In addition, B^t depends on whether or not the type is mutually known, and as a result the knowledge hierarchy structure K^t is affected. The knowledge hierarchy structure depends on I^{t-1} based on the idea that the degree of understanding about mutual types is deepened by having a signal. Because the dependence of each state includes uncertainty, it is expressed using a conditional probability. Here, the point of market competition between Dealer Agents is the probability distribution for the strategy N^t for the competitor in the current state. If the combination of state variables is the state S , and Bayes’s Theorem is applied while considering the correlation described above, then the probability distribution (belief) about a competitor’s strategy can be updated as follows:

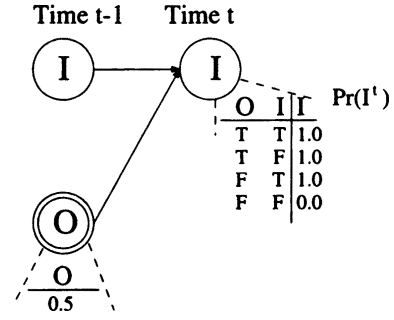


Fig. 5. Example of a sensor model.

$$\Pr(N^t|B^t, S^{t-1}) \cdot \Pr(B^t|K^t, S^{t-1}) \cdot \Pr(K^t|S^{t-1})$$

Finally, the belief of a Dealer Agent is updated by combining the probability obtained from the signal in the sensor model and the probability distribution for the competitor’s strategy obtained from the state evolution model. The Dealer Agent makes competitive strategic decisions in accordance with the framework in Section 3.2.2 using the newly updated belief as a probability distribution for the competitor’s behavior.

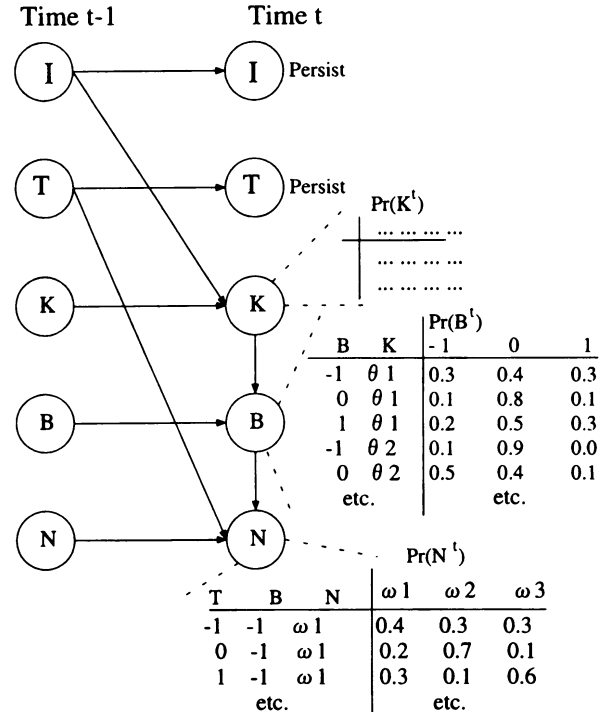


Fig. 6. Example of a state evolution model.

*The authors’ model is an example using two agents, but the dependence between states is common for all agents.

4. Design of the Agent System

Figure 7 shows the agent's internal architecture of the authors' system. As is the case in a conventional expert system, it has a reasoning engine and a knowledge base (KB). In the authors' system, a communication engine and a documentation engine are also added to deal with distributed environments. The communication engine connects to a network and acquires information from communication with other agents and from the Internet. The documentation engine compiles the obtained information in an appropriate format for the agents to use with ease. The communication and documentation engines make communication and information acquisition with other agents on a network simpler. In addition, the unified architecture in Fig. 7 for all of the agents is characteristic of this project. The communication engine and the documentation engine have many shared components for all the agents,* and as a result, development which shares the agent components is possible. This has the advantage of increasing extensibility and making design easier. In this system, a user interface (UI) is added, and agent design proceeds using the three engines, UI, and knowledge base (KB). At present a decision-making function for the reasoning engine and UI has been implemented, and the communication and documentation engines, and the function for updating belief using the DBN are under way.

4.1. Design of the user interface

An online shop manager helps the user decide on a strategy while having a dialog with the Dealer Agent via the UI shown in Fig. 8. The UI consists of two screens, a main screen and a control screen. The user can tell the Dealer Agent his/her degree of risk aversion and degree of optimism using the sliders marked with "1" and "2" in Fig. 8. In addition, in order to reduce the burden on users, in this system subjective analysis of a psychological state which includes fuzzy aspects is possible by assigning a verbal expression using words to the numbers for the degree of risk aversion and degree of optimism as shown in the example in Fig. 9. In "3," the user can select whether or not to use a strategy using Portfolio combinations. The strategy decided on by the Dealer Agent is shown to the user by displaying in text format "Customer ID," "Product(s) ordered," "Competitor and prediction of behavior," "Optimal strategy," "Expected payoff," and "Utility." Finally, the user clicks "OK" as shown in "5" if she/he is pleased with the strategy that is shown. If the strategy the Dealer Agent has decided on is not appealing, the user clicks "Change"

*For instance, in the communications engine, all agents have the shared structure shown in Fig. 10 later in this paper.

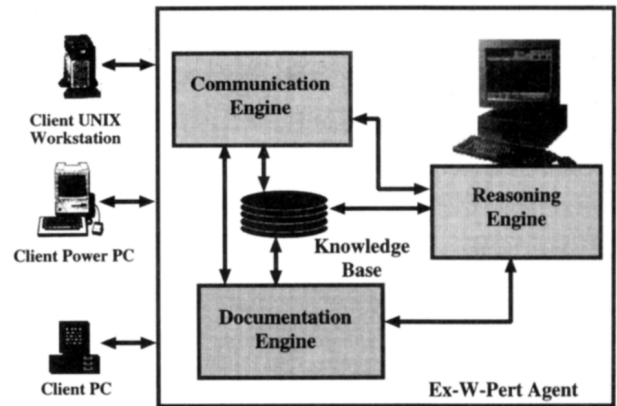


Fig. 7. Agent architecture.

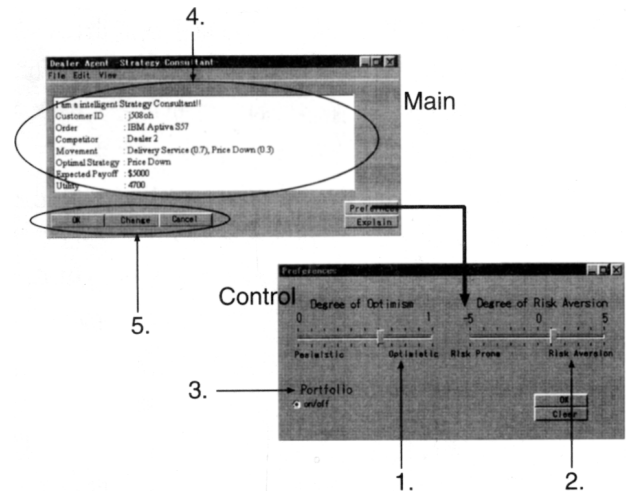


Fig. 8. User interface.

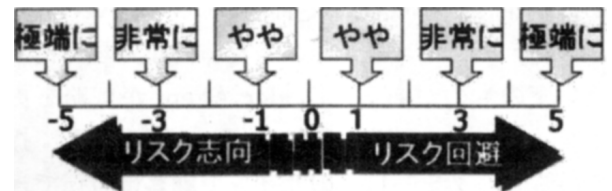


Fig. 9. Degree of risk aversion by verbal representation.

and can view other possible strategies. When an appealing strategy is found, the user selects “OK.” The reason the Dealer Agent decides on a strategy after receiving approval from the user is that although this system can be seen as a decision-making support system from the standpoint of supporting strategic decision-making, what is important in decision-making support is supporting the decision-making of the user, and not having the system make its own decisions as it likes [11].

An explanatory function will be implemented as a new function in the UI. The explanatory function is a function which provides explanations of the competitive process via the GUI to users during and immediately before the end of a competition. During competition, winners and losers frequently can be clearly separated using the characteristics of the competition, but in the authors’ system, because the Dealer Agent makes decisions using a back end, the user may find himself/herself dissatisfied when his/her own agent loses a competition. Thus, a function which displays the flow of strategic decision-making in a competition using graphics is necessary in the GUI. In addition, from the standpoint of HCI (Human–Computer Interaction), providing the sense that the user is controlling the agent is important [7].

4.2. Design of the reasoning engine and the knowledge base

The KB in the e-commerce agents described in Section 2 is designed using a three-layer hierarchical structure (S-R-K Layers)—Skill-based level, Rule-based level, and Knowledge-based level—in order to model real-world human knowledge and adapt flexibly to complex problems [3]. At the Skill-based level, responses which are stipulated based on judgment standards indicated by the user are implemented. At the Rule-based level, responses based on rules and on examples are used. At the Knowledge-based level, the agent adapts creatively through its own trial and error with respect to problems that are not present in the judgment standards or the rules. With respect to support for market competition, flexible behavior is needed when knowledge about a competitor is uncertain. As a result, responses at the Skill-based level or the Rule-based level cannot be used. Thus, the Dealer Agent has a Knowledge-based level KB based on the DBN described in Section 3.3 built in. A topic for the future is implementing a decision-making function which effectively uses past examples using the Rule-based level and then making it possible to use it in concert with the Knowledge-based level.

The reasoning engine has a function to use the knowledge accumulated in the KB. For the Dealer Agent which represents the topic of this paper, probability analysis is

performed for decision-making and on the behavior of a competitor, as described in Section 3.

4.3. Design of the communication engine and the documentation engine

The Dealer Agent must be able to act on the Internet in order to engage in market competition. Thus, it is designed to be connected to a network, and a communications engine is created to fulfill the role of managing communications and messaging with other agents, and to acquire information from the Internet. COBRA and Java were used as the development protocol and language because of considerations regarding large-scale commercial transactions via the Internet for e-commerce, and superior compatibility with networks, platform independence, and interoperability. The flow of message management in the communications engine is shown in Fig. 10. First, messages from other agents or signals regarding the behavior of a competitor are received as input and stored in the input queue. Next, how the input data are to be processed is determined through discussion with the message manager and the reasoning engine, and the decision is sent to the appropriate engine. Output is similar; the message splitter talks with the reasoning engine, then determines whether to send the data to another engine or return them to the UI, and stores the data in the output queue.

The specific role of the communications engine for the Dealer Agent is to collect information on the behavior

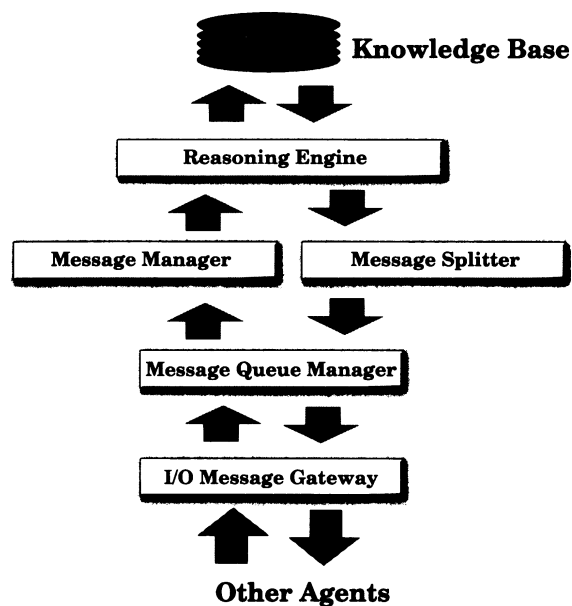


Fig. 10. Message management.

of a competitor in a symbolic fashion, to receive orders for products from the Customer Agent, and to talk with the user via the UI. The order receiving function in the Customer Agent has already been implemented using a KQML-COBRA-based dialog protocol [3]. With respect to dialog with the user via the UI, implementation was performed using normal Java socket connections. The competitor signals are acquired by collecting information present in electronic shopping malls, sites where online shopping is advertised on the Internet. An electronic shopping mall describes on its own site (even partially) what services it has implemented in order to have each online shop advertise the content of its services to customers. Therefore, hints about the behavior of a competitor can be obtained by having the communications agent gather this content as signals. Promotional sites for e-commerce will, it is said, focus on XML descriptions in the future [6]. In descriptions using XML, the information is structured based on its meaning. As a result, structural information based on the content can be readily acquired in a symbolic format that can be handled by the probability estimation framework in Section 3.3.

The documentation engine performs editing and restructuring of data obtained from the reasoning engine and the communications engine into an appropriate format. Below a concrete example of the processing in the Dealer Agent documentation engine is given. When the Dealer Agent is updating a strategy, inside the agent data which are handled using a symbolic format are edited in XML format so that they can be advertised to customers on the Dealer Agent's site, and edited into text format when passing the results of decision-making to a user via the UI.

5. Implementation and Evaluation Example

5.1. Product purchase flow

This section gives an operational example of the system when engaging in e-commerce (product purchases) using the authors' system. The appearance of the Customer Agent UI is shown in Fig. 11. The customer registers as a customer with several Dealer Agents via the Customer Agent prior to purchasing a product. When the customer selects "Purchase Product" from the main screen, an estimate request screen like the one shown in Fig. 12 appears. The customer indicates the product code* for the desired product and the number in the estimate request screen. The customer selects "Estimate," and an estimate request can then be sent to all Dealer Agents the customer has previously registered with. Because the estimate request is sent to several Dealer Agents, competition occurs among the Dealer Agents. At this point, each Dealer Agent reflects the

*Product codes are provided by the Catalog Agent.

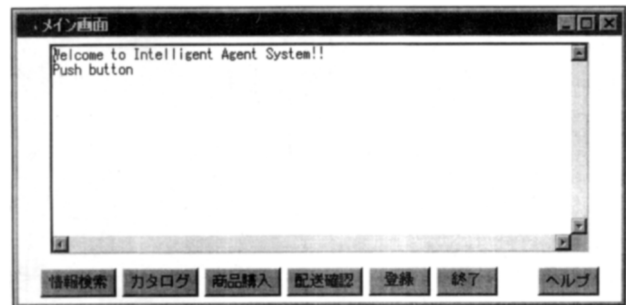


Fig. 11. Main interface of the Customer Agent.

customer's estimate request, and must return an estimate which is more appealing than its competitors. Thus, each Dealer Agent decides on a market strategy while talking with retailers and in accordance with the decision-making method described in Section 3, then returns the process to the Customer Agent.

On the Customer Agent side, the Customer Agent decides which Dealer Agent to contract with while talking to the user. Figure 13 shows the contract order screen. In this example, the most appropriate Dealer Agent is selected and displayed. In the authors' system, when estimates are provided for three products, for example, all three can be purchased from one Dealer Agent, or each product can be specified for purchase. Finally, when the customer clicks "Purchase," a message indicating this is sent to the Dealer Agent which has the contract, and a message denying the other Dealer Agents is sent.

5.2. Operational flow of a dealer agent

This section provides an example of Dealer Agent decision-making. Fundamentally, an experiment using the authors' system for practical business is desirable, but be-

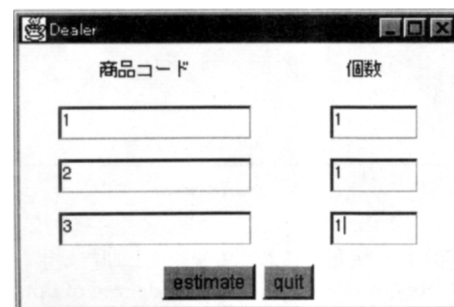


Fig. 12. Interface for call for estimation.



Fig. 13. Interface for purchasing goods.

cause obtaining actual data is problematic, analysis using micro-economics [16] is performed, and an example of decision-making is illustrated by applying the system to virtual market data which reflect a real case. In the field of economics, the structure of a market is broadly classified as monopolistic, oligopolistic, perfectly competitive, or monopolistically competitive [10]. Because competition does not occur in a monopoly or oligopoly, a system which assumes perfect competition or monopolistic competition was used.

(1) Price competition under perfect competition: Perfect competition means that the market structure has absolutely no barriers to entry or product differentiation present in various stores. In this market structure, because products cannot be differentiated, price competition is severe, and pricing competition occurs without limits on prices. Let us consider an unrestricted price competition between two Dealer Agents. Because the two Dealer Agents cannot distinguish between the virtually identical products that are handled, prices are reduced to the utmost possible, and competition occurs at pricing limits. For the sake of simplicity, Table 2 shows a payoff matrix for conditions in which a ¥40,000 product is sold to a customer at a high price (¥40,000) or a low price (¥30,000) as a market strategy. When the competitor is at the high price, if your Dealer Agent shows the low price, then purchase is made, and your Dealer Agent gains ¥30,000 in profit. If both are at the high

Table 2. Payoff matrix of price competition

	High Price	Low Price
High Price	(2, 2)	(0, 3)
Low Price	(3, 0)	(-1, -1)

price, then because there is no difference in the products, the probability of a sale is fifty-fifty, and both gain ¥20,000. If both show the low price, and if the price limit is ¥30,000, then there is no expectation of a sale at the low price (because customers compete), and so a deficit (-¥10,000) is expected because costs are not met. The preferences for this game environment are (low price, high price) > (high price, high price) > (high price, low price) > (low price, low price), and in economics this pattern has been analyzed as the chicken game [16].

(2) Product differentiation in monopolistic competition: Monopolistic competition is a market structure in which barriers to entry are low and there is product differentiation for several stores. Because there is product differentiation, if a competitor's behavior can be predicted, then price competition which can easily be eliminated can be avoided. Here as well, let us consider competition between two Dealer Agents. Table 3 shows an example of a payoff matrix for market competition with product differentiation. Because each Dealer Agent wants to avoid price competition, let us consider trying to gain customers by changing the content of service with the competitor. For instance, two Dealer Agents that handle computers consider whether to focus on PCs or on workstations. If both handle the same items, then the price competition in the previous section falls, and there is a risk of being wiped out. At present, although the demand for PCs is higher than that of workstations, if the competitor focuses on PCs, and you adopt a strategy of focusing on workstations, then you can dominate the demand for workstations. If both Dealer Agents adopt a strategy of focusing on PCs with their high demand and harsh price competition, then the payoff drops. The preferences for this game environment then are (PC, WS) > (WS, PC) > (WS, WS) > (PC, PC). In economics, this has been used for the analysis of product differentiation in monopolistic competition as a leader game.

5.2.1. An example of dealer agent decision-making

In the two market structures above (business environments), there was no dominant strategy, and the optimal reactive strategy was to adopt a strategy different from the competitor's. If the competitor's behavior is known (Level 1 conditions in Section 3.2.1), then a strategy opposite to

Table 3. Payoff matrix of product differentiation

	WS	PC
WS	(2, 2)	(3, 5)
PC	(5, 3)	(1, 1)

the competitor's should be adopted. Because each Dealer Agent hides its strategy from the other, however, prediction of the competitor's behavior is uncertain. In this section, examples of Dealer Agent decision-making based on prediction of the behavior of a competitor are given.

(a) Example of decision-making based on degree of optimism

When there is no information about the behavior of a competitor, belief is derived from Eq. (7) based on the degree of optimism of the user of the Dealer Agent in accordance with the framework in Section 3.2.3, and a strategy to optimize the expected utility is adopted in accordance with the OWA operator. For instance, in example 1 above (price competition), when the user's degree of optimism is $\text{Opt}(W) = 0.7$, $[w_1, w_2] = [0.7, 0.3]$, and so the expected value for a "high price" is $0.7 \times 2 + 0.3 \times 0 = 1.4$, and for "low price" it is $0.7 \times 3 + 0.3 \times (-1) = 1.8$. As a result, "low price" is chosen. Figure 14 illustrates the changes in the expected utility of the two strategies based on the magnitude of the degree of optimism for the business activity in example 1 using this method.* As can be seen in the results, up to a degree of optimism of 0.5, "high price" has a higher expected utility, and so "high price" is chosen. When 0.5 is exceeded, "low price" has a higher expected utility, and so "low price" is chosen. As can be seen in the payoff matrix for example 1, the "low price" strategy is more beneficial when the competitor uses the "high price" strategy, although when the competitor uses the "low price" strategy, the loss is greater. In other words, when accurate, the payoff is high, but when wrong, the loss is high. In contrast, for the "high price" strategy, the payoff is low when accurate, but the loss is low when wrong. These results can also be seen in the magnitude of the tendency of the expected utility. Therefore, when the degree of optimism is high, the high-payoff "low price" strategy is chosen, but when the degree of optimism is high there is a fear of loss, and so the "high price" strategy is adopted in order to profit more securely.

In the same fashion, Fig. 15 shows the expected utility for example 2 (competition with product differentiation). Workstations have the greater expected utility up to a degree of optimism of 0.3, and so workstation sales are used. When 0.3 is exceeded, PCs have the greater expected utility, and so PC sales are used. As was the case previously, this is also because as the degree of optimism rises, the Dealer Agent is set to favor PCs due to the larger profit and loss with PCs.

Given the above, interactive decision-making based on the user's degree of optimism can be made under the

* Although the effective value is negative, in this case the user decides whether or not to adopt the strategy. The same holds below.

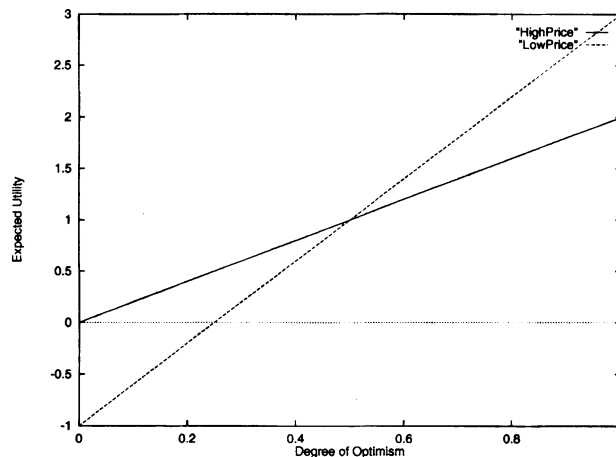


Fig. 14. Result of example 1 with degree of optimism.

authors' system even when the probability of the competitor's behavior is completely unknown.

(b) Example of decision-making based on risk management

An operational example and the operating characteristics for a Dealer Agent in a Level 2 situation as in Section 3.2.2 are shown here. In this instance, the probability distribution is already known, and as a result, the system adopts the strategy which maximizes the utility function depicted in Eq. (2).

In example 1, the belief of the Dealer Agent is assumed to be a state in which the Dealer Agent believes that

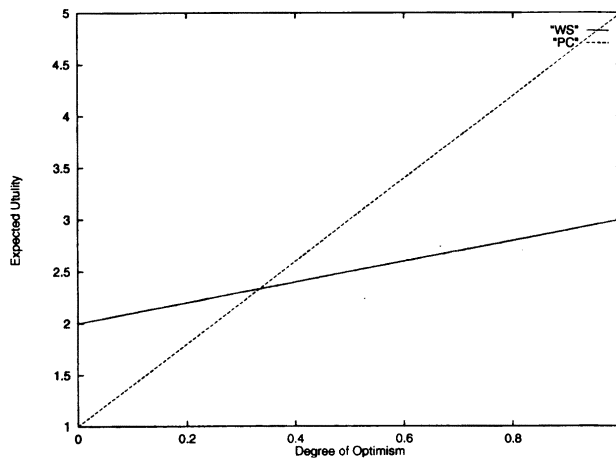


Fig. 15. Result of example 2 with degree of optimism.

there is a 0.6 probability that the competitor will take the “high price” strategy, and a 0.4 probability that it will take the “low price” strategy. At this point, if the expected value is simply taken, then the expected value for when the “high price” strategy is used is $0.6 \times 2 + 0.4 \times 0 = 1.2$, and the expected value for the “low price” strategy is $0.6 \times 3 + 0.4 \times (-1) = 1.4$. As a result, the “low price” strategy has a greater utility, although more flexible decision-making is possible with the proposed risk management function. Figure 16 shows the changes in the utility of the two strategies depending on the magnitude of the degree of risk aversion for when the risk management function is used in example 1. The “low price” strategy is more beneficial up to a degree of risk aversion of 0, and as a result, the “low price” strategy is adopted. When 0 is exceeded, the preferences reverse, and the “high price” strategy is adopted. This is because the Dealer Agent prefers the low risk and low return of the “high price” strategy as the degree of risk aversion rises, because the “low price” strategy is high risk and high return with large profits and losses, as was described earlier (this can also be seen in the tendency of the utility function).

Figure 17 shows the changes in the utility that accompany the degree of risk aversion when the probability that the competitor will take the “workstation” strategy is 0.6, and the probability that it will take the “PC” strategy is 0.4 in example 2. When the degree of risk aversion exceeds 0, the utility value inverts as the preferences of the Dealer Agent shift to risk aversion, and the Dealer Agent will select “workstations.” This is also because the Dealer Agent prefers the low risk and low return of “workstations” when the user’s preferences shift to risk aversion.

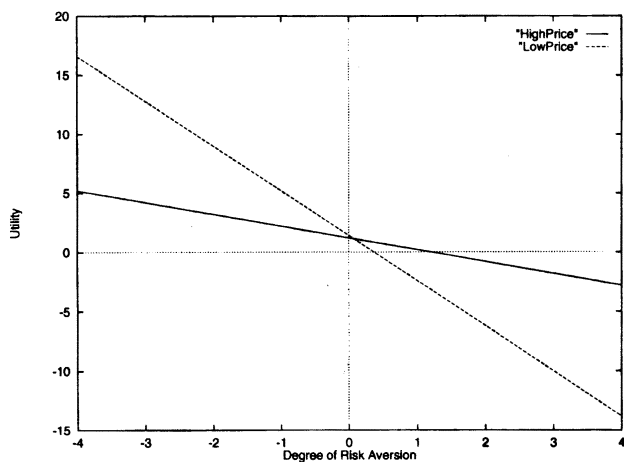


Fig. 16. Result of example 1 with degree of risk aversion.

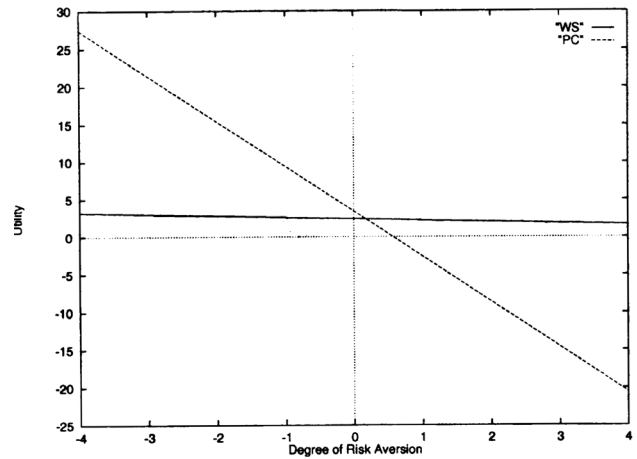


Fig. 17. Result of example 2 with degree of risk aversion.

Given the above, it is clear that decision-making which reflects the desire for user risk management can be performed in the authors’ system.

5.2.2. Example of updating belief

With respect to updating belief, the fundamental characteristics of any market structure are the same. As a result, here an example using example 1 will be given. For the sake of simplicity, the type for each Dealer Agent will be $T = \{-1, 0, 1\}$ (risk comfortable, neutral, averse, respectively) for both agents (Belief of Belief also has three states).

In order to predict the behavior of the competitor (for example 1, the “high price” strategy or the “low price” strategy), the type of competitor and the Belief of Belief (how the competitor views my type) must both be considered. Here, an example is given of updating belief as prior knowledge when information about the Belief of Belief becomes available as a signal in a situation in which the probability distribution of the competitor’s type is known as a belief. First, the belief regarding the competitor’s type is assumed to have a probability of 0.4 for risk comfortable ($T = -1$), a probability of 0.3 for risk neutral ($T = 0$), and a probability of 0.3 for risk aversion ($T = 1$) for prior knowledge. As this point, based on the results in the previous section, the “low price” strategy is used for when risk is comfortable, and the “high price” strategy when risk averse. When risk is neutral, the expected values are almost the same, and so if the “high price” and “low price” strategies are equally probable, then the prior knowledge (belief) about the behavior of the competitor is $0.3 \times 1.0 + 0.3 \times 0.5 = 0.45$ for the “high price” strategy, and $0.4 \times 1.0 + 0.3 \times 0.5 = 0.55$ for the “low price” strategy to be adopted.

However, this belief does not reflect what the competitor will predict for its own type, and as a result, the behavior of the competitor cannot necessarily be predicted accurately. Thus, in a situation in which the Dealer Agent knows that the knowledge hierarchy structure is in state θ_4 (see Section 3.3) ($P(\theta_4) = 1$), as for information on the Belief of Belief, the Dealer Agent finds information that the probability that the competitor thinks its own type is risk oriented ($B = -1$) is 0.5, the probability that it is risk neutral ($B = 0$) is 0.2, and the probability that it is risk averse ($B = 1$) is 0.3.* If only the belief regarding Belief of Belief is considered, the competitor will adopt a strategy opposite to its own, and so when “ $B = -1$,” it will use the “high price” strategy, when “ $B = 0$ ” it will with 50–50 odds use either strategy, and when “ $B = 1$ ” it will adopt the “low price” strategy. In other words, the signal obtained can be interpreted as information that the probability that the competitor will use the “high price” strategy is $0.5 \times 1.0 + 0.2 \times 0.5 = 0.6$, and the probability of the “low price” strategy is $0.3 \times 1.0 + 0.2 \times 0.5 = 0.4$. Thus, by joining together the prior knowledge regarding the competitor’s behavior that the Dealer Agent already has (derived from belief about the type) with the information on the behavior of the competitor resulting from the obtained signal, the belief about the competitor’s behavior can be updated using the DBN in Fig. 6. For instance, the probability that the state $T = -1$ and the state $B = -1$ will occur at the same time is $0.4 \times 0.5 = 0.2$, and the belief regarding the competitor’s behavior for when these two occur at the same time is [high price, low price] = [0.5, 0.5] on average. In other words, when Fig. 6 overall is considered, the probability that the competitor will use the “high price” strategy when the state is $T = -1$ or $B = -1$ is $0.2 \times 0.5 = 0.1$, and the probability for the “low price” strategy is $0.2 \times 0.5 = 0.1$. The results for all combinations of the variables for the states T and B modeled in Fig. 6 using this calculation show that the probability that the competitor will adopt the “high price” strategy is 0.525, and the probability for the “low price” strategy is 0.475. In other words, belief is updated to “P (high price) = 0.525, and P (low price) = 0.475” by obtaining information about the Belief of Belief, one more element for determining the behavior of the competitor, based on the belief “P (high price) = 0.45, and P (low price) = 0.55” when the only prior knowledge is the type of competitor. The Dealer Agent estimates the probability that the competitor will adopt the “low price” strategy a bit on the high side at the stage when only prior belief is available, but then revises the “high price” strategy estimate a bit upward when belief is updated. This is because it estimates that the probability that the competitor will adopt the “high price” strategy, opposite of its own, is high due to having information that the

competitor strongly believes its own type to be “risk oriented.” Based on the above, the authors’ system can predict the behavior of a competitor from many different perspectives by unifying various uncertain information when predicting the behavior of a competitor.

Note that although information about the Belief of Belief was addressed in this example, if the strategy of the competitor in the prior state can be predicted, then belief can be updated using the same process. In addition, the situation for updating belief is identical for example 2 as well.

5.3. Advantages of the system

The advantages of using the authors’ system for e-commerce can be viewed as follows. The first advantage is that a user does not need to have knowledge of strategic decision-making for sales. In order to have stable profits in a wide-ranging business like e-commerce, strategic interactions with competitors are important. As a result, specialized knowledge is needed. However, at present the services of consultants who provide strategic decisions for retail, the object of this paper, are quite expensive. When the authors’ system is used, the system can engage in strategic decision-making as a representative of the user merely by having the user provide information about her/his degree of optimism and degree of risk aversion. As a result, the system is very useful for retailers when they enter a network market.

The second advantage is that interactive decision-making using the degree of optimism is possible even in situations in which the behavior of the competitor is completely unknown. As competition becomes more fierce in e-commerce, the number of businesses that join the market is expected to rise. Under such circumstances situations in which adequate information cannot be gathered may occur. In the authors’ system, a decision-making function based on the degree of optimism which provides behavioral information that can address such circumstances has been implemented.

The third advantage is that the user’s desires can be reflected. In a decision-making support system, it is important for the system to reflect the user’s desires in order to have the user trust it [7, 11]. In the authors’ system, decision-making is made while reflecting the degree of optimism and the degree of risk aversion. As a result, the users can trust the system.

In this paper the authors confirmed these advantages by using examples applied to market structures. Because game theory, the basis of economic analysis, was used in the system, the system can be applied to many other market structures by changing the payoff parameters. In addition, although the examples used competition between two

* A method to derive knowledge as a probability from an information source such as the Web described qualitatively has been proposed in Ref. 2.

agents, the flow of decision-making is fundamentally the same for any number of competitors.

6. Related Research

Reference 5 presents research on supporting e-commerce dealers using an agent system. In this research sales strategies are decided upon by having users and dealers talk to each other. What is particularly interesting here is the discussion of combining strategies. Sales competition in conventional e-commerce involves simple price discounts, resulting in competing elements with a very limited breadth. In contrast, the system proposed in Ref. 5 implements negotiations on various elements using product pricing that has value added to the discounts. However, the system in Ref. 5 cannot predict the behavior of a competitor, and so the impact from the competitor on whether or not a product can be sold is significant. As a result, payoff stability is a problem. In the authors' system, the agents have a function to combine strategies which consider the behavior of the competitor using a Portfolio, and so higher stability can be expected.

The Shopbot agent proposed in Ref. 4 has a function to support setting prices. This research proposes a simple economic model. In this model, dynamic behavior that occurs among sets of buyers and sellers is simulated. Game theory equilibrium has been confirmed to occur dynamically in this model. The system proposed in the authors' paper cannot set prices, and so in this sense the Shopbot is superior to the authors' system. However, Shopbot cannot make sales decisions which consider value added outside of pricing, and as a result the authors' proposed system can be said to be better for various sales strategies. A future direction for research might be to combine the price setting method in Ref. 4 and the method for strategically determining value added based on managing uncertainty as proposed in this paper to create one system.

7. Conclusion

In this paper the authors described the development of an agent which supports sales competition in e-commerce for the purpose of developing a multiagent system which supports competition in social activities on the World Wide Web. In the proposed system a three-level decision-making model for addressing uncertainty in competitive states was built into the system, thus enabling strategic sales strategy decisions which consider the behavior of sales competitors. In addition, the authors gave examples of decision-making for agents based on transactions based on economic analysis. In the future, completing the agents and

demonstrating the validity of their use for a wider range of cases is important.

Acknowledgment. The authors thank the anonymous referees for valuable opinions.

REFERENCES

1. Adams N, Yesha Y. Strategic direction in electronic commerce and digital libraries: Towards a digital agora. *ACM Computer Surveys* 1996;28:818–835.
2. Druzdzel MJ, van Gaag LC. Elicitation of probabilities for belief networks: Combining qualitative and quantitative information. *Proc 11th Int Conf of Uncertainty in Artificial Intelligence*, Quebec, 1995, p 141–148.
3. Far BH, Souenia SO, Hassan H, Saniepour S, Hashimoto AH. An integrated reasoning and learning environment for WWW-based software agents for electronic commerce. *IEICE Trans Inf Syst* 1998;E81-D:1374–1386.
4. Greenwald AR, Kephart JO. Shopbots and pricebots. *Proc 16th Int Conference on Artificial Intelligence*, 1999, p 506–511.
5. Guttman R, Maes P. Agent-mediated integrative negotiations for retail electronic commerce. *AMET '98*, p 1–13.
6. McGraph S. XML by example: Building e-commerce applications. Prentice-Hall; 1998.
7. Norman DA. How people might interact with agents. In Brandshaw JM (editor). *Software agents*. MIT Press; 1997. p 49–55.
8. Onjo H, Far BH. A unified view of heterogeneous agents' interaction. *Proc Workshop on Software Agent and its Applications*, 2000, p 133–140.
9. Parsons S, Sierra C, Jennings NR. Agents that reason and negotiate by arguing. *J Logic Computation* 1998;8:261–292.
10. Samuelson PA. *Economics*. McGraw-Hill; 1948.
11. Turman E, Aronson JE. *Decision support systems and intelligent systems*, 5th ed. Prentice-Hall; 1998.
12. Yager RR. On ordered weighted averaging aggregation operators in multi-criteria decision making. *IEEE Trans SMC* 1988;18:183–190.
13. Yager RR. Decision making under Dempster-Shafer uncertainties. *Int J Gen Syst* 1990;20:233–255.
14. Ishida J. Considerations about agents. *Artif Intell Assoc* 1995;10:663–667.
15. Ishikawa A. *Decision-making theory*. Kyoritsu Publishing; 1983.
16. Kajii A, Matsui A. *A strategic approach to micro-economics*. Nippon Hyoronsha; 2000.

AUTHORS (from left to right)



Hiroyuki Onjo graduated from the Department of Information and Computer Sciences of Saitama University in 1999 and completed his master's studies in 2001. He then joined Casio Computing Ltd., and is pursuing research related to multiagent system and decision-making theory.

Behrouz H. Far (member) graduated from Tehran University in 1983 and completed his doctorate at Chiba University in 1990. At present he is an associate professor on the Faculty of Engineering of the University of Calgary. He is pursuing research related to distributed artificial intelligence and quantitative estimation. He was the manager of the KBSE Research Group from 1995 to 1997. He is a member of ACM, IEEE, the Japanese Society for Artificial Intelligence, and the Information Processing Society of Japan.