A cooperative exchanging mechanism among seller agents for group-based sales

Takayuki Ito\textsuperscript{a,*,} Hiromitsu Hattori\textsuperscript{b}, Toramatsu Shintani\textsuperscript{c}

\textsuperscript{a}Center for Knowledge Science, Japan Advanced Institute of Science and Technology, I-1 Asahidai, Tatsunokuchi, Nomi, Ishikawa 923-1292, Japan

\textsuperscript{b}Graduate School of Engineering, Nagoya Institute of Technology, Gokiso, Showa-ku, Nagoya 466-8555, Japan

\textsuperscript{c}Department of Intelligence and Computer Science, Nagoya Institute of Technology, Gokiso, Showa-ku, Nagoya 466-8555, Japan

Abstract

Agent-mediated electronic markets have been a growing area in intelligent agent research and development in recent years. Agents can act autonomously and cooperatively in an electronic market on behalf of their users. In such an electronic market, if a seller agent does not have enough of a particular item, it misses the opportunity to sell the item. Buyers also miss the opportunity to purchase the item. Namely, the overall negotiation utility is decreased. Thus, we propose a new cooperation mechanism among seller agents based on exchanging their goods in our agent-mediated electronic market system, G-Commerce. In G-Commerce, seller agents and buyer agents negotiate with each other. In our model, seller agents cooperatively negotiate in order to sell goods in stock. Buyer agents cooperatively form coalitions in order to buy goods based on discount prices. Seller agents’ negotiations are completed by using an exchanging mechanism for selling goods. Our experiments show that this exchanging mechanism enables seller agents to sell goods in stock effectively. We also demonstrate how our exchanging mechanism satisfies Pareto optimality.

© 2002 Published by Elsevier Science B.V.

Keywords: Agent-mediated electronic commerce; Cooperation among agents; Seller cooperation mechanisms; Exchanging mechanisms

1. Introduction

Over the last few years, the number of electronic commerce (e-commerce) sites has increased tremendously. These e-commerce sites have introduced new business models for effective and efficient commerce. Intelligent agents have been studied extensively in the field of artificial intelligence and multiagent systems [19]. An agent can act autonomously and cooperatively on behalf of its users on the Internet. This is very useful since it is tedious for people to monitor, buy, and sell at multiple e-commerce sites. If we applied agent technologies to e-commerce systems, we could further enhance the intelligence of user support.

There have been several studies on agent-mediated electronic commerce [7]. We can classify this research into two categories. One category covers...
agent-based electronic markets (e.g. Kasbah [3], Tete-A-Tete [8], FishMarket [17], eMediator [18], and AuctionBot [21]). The other covers shopping information gathering agents [1,5,6,10,11]. This related work is described in Section 5. In this paper, we focus on agent-based electronic markets.

There are several agents in an electronic market. Agents try to sell or buy goods on behalf of their user. The main advantage of agent-based electronic markets is that agents can reduce a user’s work load when he is selecting, monitoring, and negotiating. Suppose a user wants to sell an item (e.g. a computer). If the user employs a seller agent in an electronic market, the user does not need to find several buyers, select one buyer, and negotiate with that buyer. Instead, the user’s agent can automatically find several buyer agents, select the best buyer agent, and negotiate with the best buyer agent on the user’s behalf.

Many e-commerce sites have been developed that can handle volume discounts (e.g. Volumebuy.com). On these sites, buyers can negotiate advantageously with sellers and purchase goods at volume discount prices by forming a buyer coalition. In this paper, we propose a new agent-mediated electronic commerce system based on volume discounts. In this system, buyer agents can effectively make a coalition and purchase items at the volume discount price [12]. Yamamoto and Sycara [22] have also proposed a volume discount scheme (the GroupBuyAuction scheme) for coalition formation among buyer agents on agent-based electronic markets.

If a seller agent does not have enough of a particular item, he misses the opportunity to sell the item. Buyers also miss the opportunity to purchase the item. Namely, if a seller agent does not have enough items, the overall negotiation utility is decreased. In the real world, a seller can try to find another seller agent who has extra items in stock. Then, they can try to make an exchange so that they can sell enough items to meet buyer demand. In our electronic market, G-Commerce (a group-based e-commerce support system), we propose a cooperation mechanism among seller agents based on exchanging items in their stock. We previously proposed an exchanging protocol among agents for a meeting scheduling problem [9]. In this paper, we modify the exchanging protocol and apply it as a cooperation mechanism among seller agents on G-Commerce.

The situation we assume is in fact typical and practical. Our aim is to realize a free market support method for C2C (consumer to consumer) trading on the Internet. For example, we assume a free market where used books or used furniture are traded. We do not assume B2C or B2B commerce. Instead, we assume a P2P-based free market on the Internet. Sellers and buyers can make a small community for trading based on the P2P network style. In such a free market, it is natural to assume that sellers do not have enough goods to handle a lot of buyers’ demands. Furthermore, in such a free market, volume discounts can be frequently made since in general sellers hope to sell their goods as soon as possible. In such practical situations, our proposed method is effective.

This paper consists of the following five sections. In Section 2, we present an outline of our agent-based e-commerce system, G-Commerce. Also, we formalize an agent-mediated market system based on volume discounts. In Section 3, we propose a new exchanging mechanism among seller agents. In Section 4, we demonstrate the Pareto optimality of our exchanging mechanism, and discuss the results of our current experiments. In Section 5, we overview the work of others in this area of research. We give our concluding remarks in Section 6.

2. G-Commerce: An agent-mediated electronic market

2.1. Outline of G-Commerce

Fig. 1 shows an outline of our electronic market system, G-Commerce. There is one electronic market in the system, on which agents act both cooperative-ly and autonomously. A seller (human) can create a seller agent that can automatically find and negotiate with buyer agents. A buyer (human) can create a buyer agent that can automatically find seller agents and negotiate with seller agents. These agents can also access World Wide Web sites and sell/purchase goods on the Internet. Agents are connected through
a P2P (Peer-to-Peer) network [13]. While sellers can advertise their items on their own computers, buyers can also present their demands on their own computers. In this paper, a market means a group of agents that are connected with each other via a P2P network.

In G-Commerce, a seller agent can sell multiple items. Each item has a price table, a deadline, and the number of that item in stock. A price table represents the item’s discount rate according to the number of buyers. Fig. 2 shows an example of a price table. According to this example, if buyer agents purchase five items, the unit price becomes $85.00. If buyer agents purchase twenty items, the unit price becomes $35.00.

2.2. A negotiation scheme on G-Commerce

G-Commerce is a kind of reverse commerce system (e.g. Priceline.com), where buyer agents pool their demands and seller agents offer discount prices to sell large volumes of products at once. We define terms and notations as follows.

Buyer agents: \( B = \{b_1, b_2, \ldots, b_j\} \) denotes a set of buyer agents.

Seller agents: \( S = \{s_1, s_2, \ldots, s_m\} \) denotes a set of seller agents.

Items: \( G = \{g_1, g_2, \ldots, g_n\} \) is a set of items.

Seller’s items: \( G(s_j) = \{g(s_j)_1, g(s_j)_2, \ldots, g(s_j)_n\} \) is a set of seller \( s_j \)’s items. A seller can sell several kinds of item.

![Fig. 1. Outline of G-Commerce.](image1)

![Fig. 2. Example of a seller’s price table.](image2)
Buyer’s desired item: is a buyer $b_s$’s desired items.

Price table: $s_j$’s price table of $g_i$ is represented as the function $p_{s_j,g_i}: N \rightarrow R$. and are the set of natural numbers and real numbers, respectively. $p_{s_j,g_i}(n)$ is a unit price when $n$ of $g_i$ is sold together. $n$ denotes the number of $g_i$.

Cost of items: cos $r(s_j, g_i)$ denotes $s_j$’s cost for purchasing or producing item $g_i$.

Reservation price: $r(b_s, g_i)$ represents buyer agent $b_s$’s reservation price for $g_i$.

Seller’s number of items: $num(s_j, g_i)$ denotes the number of $g_i$ items sold by $s_j$.

Buyer’s desired number of items: $num(s_j, g_i)$ denotes buyer agent $b_s$’s desired number of $g_i$.

Coalition: $C(g_i) \subseteq B$ denotes a buyer coalition to purchase $g_i$.

Coalition’s desired number of items: $num(C(g_i)) = \sum_{b_k \in C(g_i)} num(b_k, g_i)$ denotes coalition $C(g_i)$’s desired number of item $g_i$.

Seller’s utility: When seller agent $s_k$ buys $g_i$ from $s_j$, his utility can be defined as $u(s_k, g_i) = r(s_k, g_i) \cdot num(s_k, g_i) - p_{s_k, g_i}(num(s_k, g_i)) \cdot num(b_s, g_i)$.

Buyer coalition’s utility: Coalition $C(g_i)$’s utility can be defined as $u_{C(g_i)}(g_i) = \sum_{b_s \in C(g_i)} r(b_s, g_i) \cdot num(b_s, g_i) - p_{s_j, g_i}(num(C(g_i))) \cdot num(C(g_i))$.

In G-Commerce, the basic idea is that buyer agents first try to buy items directly from seller agents. If they fail, the seller agents try to exchange their items so that they have enough items in their stock to sell. Agents negotiate with each other according to the following steps.

Step 1. Buyer agents arrive at a market. Each buyer agent declares its demand. The demand of buyer agent $b_k$ consists of the reservation price $r(b_s, g_i)$ and the desired number of items $num(b_k, g_i)$. Buyer agents trying to purchase the same item $g_i$ form coalition $C_j$.

Step 2. When seller agent $s_j$ arrives at the market, if there are coalitions $C(g_i)$ and $g_i \in G(s_j)$, it tries to sell item $g_i$. If not, it waits for a buyer who wants to purchase its item $G(s_j)$ until the deadline given by the user is reached.

Step 3. If $num(s_j, g_i) \geq \sum_{b_k \in C(g_i)} num(b_k, g_i)$, then seller $s_j$ succeeds in selling item $g_i$ to coalition $C(g_i)$ and this negotiation is concluded. If $num(s_j, g_i) < \sum_{b_k \in C(g_i)} num(b_k, g_i)$, then go to Step 4.

If there are two or more sellers qualified at the same time, the seller who arrives at the market first is selected for trading. Sellers are ordered by their arrival time. Suppose there are two seller agents, A and B, and B arrives at the market after A. In this case, if A and B are qualified for trading, A is selected since A arrives before B.

Step 4. In this step, in order to satisfy buyer demand, seller agent $s_j$ tries to increase the number of items $g_i$ in its stock by using our exchanging mechanism. For example, seller agent $s_j$ can get more of item $g_j$ from another seller agent $s_k$. In exchange for item $g_j$, seller agent $s_j$ gives another item $g_i$ to seller agent $s_k$. The details of the exchanging mechanism are explained in Section 3, but can be outlined as follows:

First, seller agent $s_j$ tries to find a seller agent who is waiting to exchange item $g_j$. If there is such an agent, seller agent $s_j$ tries to exchange items. When there are a lot of agents who can exchange items, it tries to find the best deal. However, if there is no agent waiting to exchange items, seller agent $s_j$ tries to wait for another seller agent to arrive at this market until the deadline given by the user is reached.

3. An exchanging mechanism among seller agents

In Step 4 in Section 2.2, if a seller agent does not have enough items, he tries to wait for another seller agent who can exchange items. Then, if they succeed in exchanging their items, both seller agents can increase their utility. Even the buyer agents can benefit because of the greater discounts offered for purchasing a large number of items. Here, we propose a cooperation mechanism among seller agents based on exchanging their items.

Fig. 3 shows an example of exchanging items. There are two seller agents, $s_1$ and $s_2$. Seller agent $s_1$ has four $g_1$s and two $g_2$s. Seller agent $s_2$ has two $g_1$s and three $g_2$s. Suppose that a buyer coalition $C(g_i)$
tries to purchase six $g_1$s from seller agent $s_1$. Also, another buyer coalition $C(g_2)$ tries to purchase five $g_2$s from the seller agent $s_2$. Since seller agent $s_1$ does not have enough $g_1$s, it tries to negotiate with seller agent $s_2$.

First, seller agent $s_1$ offers to give two $g_2$s to seller agent $s_2$ in exchange for which $s_2$ gives $s_1$ two $g_1$s. Here, if $p_{s_1,g_1}(6) \times 6 - \cos t(s_1, g_1) - p_{s_1,g_2}(2) \times 2 > 0$ for $s_1$, and $p_{s_2,g_2}(5) \times 5 - \cos t(s_2, g_2) - p_{s_2,g_1}(2) \times 2 > 0$ for $s_2$, we define that both of the seller agents, $s_1$ and $s_2$, can agree to this proposal. Since both of the seller agents can make a profit, they can exchange items. We define a criterion for exchanging items as follows:

**Definition 1.** (A criterion for exchanging items)
Suppose there are two seller agents, $s_1$ and $s_2$, and two kinds of items, $g_1$ and $g_2$.

**(Case 1)** In the case that $st(s_1, g_1) = num(C(g_1)) - num(s_2, g_1) > 0$ and $st(s_2, g_2) = num(C(g_2)) - num(s_1, g_2) > 0$, if agent $s_1$’s utility $u(s_1) = p_{s_1,g_1}(num(C(g_1)) \times num(C(g_1)) - \cos t(s_1, g_1)) - p_{s_1,g_2}(st(s_2, g_2)) \times st(s_2, g_2) > 0$, and agent $s_2$’s utility $u(s_2) = p_{s_2,g_2}(num(C(g_2)) \times num(C(g_2)) - \cos t(s_2, g_2)) - p_{s_2,g_1}(st(s_1, g_1)) \times st(s_1, g_1) > 0$, these agents can reach an agreement since both of them can get positive utility.

**(Case 2)** In the case that $st(s_1, g_2) = num(C(g_2)) - num(s_1, g_2) > 0$ and $st(s_2, g_1) = num(C(g_1)) - num(s_2, g_1) > 0$, if agent $s_1$’s utility $u(s_1) = p_{s_1,g_2}(num(C(g_2)) \times num(C(g_2)) - \cos t(s_2, g_2)) - p_{s_1,g_1}(st(s_1, g_1)) \times st(s_1, g_1) > 0$, and agent $s_2$’s utility $u(s_2) = p_{s_2,g_1}(num(C(g_1)) \times num(C(g_1)) - \cos t(s_1, g_1)) - p_{s_2,g_2}(st(s_2, g_2)) \times st(s_2, g_2) > 0$, these agents can also reach an agreement since both of them can get positive utility.

As an exceptional case, if two seller agents are qualified for both conditions (Definitions 1 and 2), then there are two feasible exchanges. One exchange might be that $s_1$ gives $g_1$ while receiving $g_2$ from $s_2$. The other possible exchange is a reverse of that exchange. In this case, agents calculate their social surplus (i.e., total utility among two agents). Then, agents select the exchange that maximizes their social surplus. If the social surpluses are the same, agents select the exchange randomly.

Also, in our method, if two seller agents compete to exchange their goods, a specific buyer agent who arrives at the market before the other agent has the first chance to exchange the goods. If the seller agent fails to reach an agreement, then the other agent has a chance to try to exchange its goods.
4. Discussion

4.1. Pareto optimality of exchanging between two agents

Our exchanging mechanism is one of the rules applied to bargaining problems in game theory [16]. In a bargaining problem, if an outcome \((u(s_1, s_2))\) satisfies the following three conditions, we can say that the outcome is Pareto optimal.

1. \(u(s_1) \geq u(s_1)^*\) and \(u(s_2) \geq u(s_2)^*\). Here, \(u(s_1)^*\) and \(u(s_2)^*\) mean the current status. Essentially, this means \(u(s_1)\) and \(u(s_2)\) maximize the utilities.
2. \((u(s_1), u(s_2))\) is a point of \(R\). \(R\) denotes the set of all possible points.
3. There is no \((u(s_1)', u(s_2)')\) in \(R\), different from \((u(s_1), u(s_2))\) such that \((u(s_1)' \geq u(s_2)')\) and \((u(s_2) \geq u(s_2))\).

In our exchanging mechanism, agents have two strategies. One is to try to exchange. The other is to do nothing. The two strategies are called \(\text{str.} 1\) and \(\text{str.} 2\). Obviously, when agents can reach an agreement by exchanging, the utilities \(u(s_1)\) and \(u(s_2)\) maximize the agents’ utilities. This satisfies condition (1). Also, in our exchanging mechanism, the set of all possible strategies is \{\(\text{str.} 1, \text{str.} 1\), \(\text{str.} 1, \text{str.} 2\), \(\text{str.} 2, \text{str.} 1\), \(\text{str.} 2, \text{str.} 2\)\}. \(\text{str.} 1, \text{str.} 2\) means \(s_1\)’s strategy and \(s_2\)’s strategy are \(\text{str.} 1\) and \(\text{str.} 2\), respectively. The set of all possible utilities is \{\(u(s_1), u(s_2)\), \((0, 0)\), \((0, 0)\), \((0, 0)\)\}. Furthermore, \((u(s_1), u(s_2))\) is the utilities that result from \((\text{str.} 1, \text{str.} 2)\). Obviously \((u(s_1), u(s_2))\) is a point of \(R\). This satisfies condition (2). In terms of condition (3), agents can get positive utilities only when they use \((\text{str.} 1, \text{str.} 2)\). When they use other strategies their utilities are \((0, 0)\). This means condition 3 is satisfied. Therefore, we can say an outcome using our exchanging mechanism between two agents satisfies Pareto optimality.

4.2. Experimental results

In order to show how effectively our proposed exchanging mechanism can work, we conducted several experiments in the following setting.

- Number of sellers: 5
- Number of buyers: 500
- Prices of items: 200 to 300
- Kinds of items: 10
- Seller’s maximum number of any one kind of item: 10
- Distribution for buyer’s desired item: Normal distribution
- Distribution for seller’s utility for an item: Uniform distribution
- Distribution for seller’s number of items: Uniform distribution

The traditional method consists of Step 1 to Step 3. Our method consists of Step 1 to Step 4. Essentially, the traditional method does not use the exchanging mechanism we have proposed. We tried varying the parameters in order to make comparisons between several settings. One setting consists of 1000 trials. One trial consists of 1000 intervals. One interval consists of one cycle of the traditional method or our method. Buyer agents and seller agents arrive at this electronic market every \(n\) intervals. \(n\) is an inversely proportional number to the number of agents.

Figs. 4–7 show the results of our experiments. Fig. 4 shows a comparison between the seller’s utility and the number of sellers. Fig. 5 shows a comparison between the rate of items sold and the number of sellers. The vertical axis means the average of the seller’s utility in Figs. 4 and 6, and the average of the rate of items sold in Figs. 5 and 7.

As these graphs indicate, our method can gain higher utilities and a greater rate of items sold than the traditional method. In particular, if the number of sellers is between 10 and 25, our method can gain about twice the utilities that the traditional method can. In terms of the rate of items sold, our method can gain about three to five times as much as the traditional method. The reason why the seller’s utility and the rate of items sold is low when the number of sellers is low is that, since there are not enough sellers and buyers, there are fewer deals.

All figures except Fig. 6 show that the seller’s utility and the rate of items sold eventually decrease to the level of the traditional method. Since the seller’s maximum number of items was fixed, if the number of sellers increases, it is natural for a single
seller that his utility and his rate of items sold decrease.

Thus, the average of the seller’s utility and the rate of items sold eventually decreased. In Fig. 6, the seller’s utility continues to increase. The reason for this is that if seller’s maximum number of items increases, the seller’s opportunities to exchange and sell its goods increase. In all figures, the proposed method and the traditional method converge eventually. The following gives the reasons for this. In terms of Figs. 4 and 5, the number of buyers and the seller’s maximum number of any one kind of item are fixed at 500 and 10, respectively. Thus, in Figs. 4 and 5, the proposed method and the traditional method converged since the number of successful trades between a seller and a buyers’ coalition depended on the above fixed numbers. When the number of sellers increases, the number of buyers (buyers’ coalitions) is not enough to inherently make successful trades. Similarly, in Figs. 6 and 7, the number of buyers and the number of sellers are fixed at 500 and 5, respectively. Thus, in Figs. 6 and 7, the proposed method and the traditional method also converged since the number of successful trades again depended on the number of the above fixed numbers. Even if the seller’s maximum number of

![Comparison between seller’s utility and number of sellers](image1)

Fig. 4. Comparison between seller’s utility and number of sellers.

![Comparison between rate of items sold and number of sellers](image2)

Fig. 5. Comparison between rate of items sold and number of sellers.
any one kind of item increases, the amount of buyers' inherent demand is the same. For this reason, both methods converged.

We assume a P2P-based free market on the Internet. Sellers and buyers can make a small community for trading based on the P2P network style. In contrast, the current market like Priceline.com uses a central server and many sellers registered at the server. In this case, the market can be a big community. Thus, 50 is an appropriate number of sellers for experiments to show the effectiveness of our P2P-based method. When the number of sellers is smaller than 50, we can still clearly see the difference between the proposed method and the traditional method in Figs. 4–7. Thus, our proposed method is more effective than

Fig. 6. Comparison between seller’s utility and seller’s maximum number of items.

Fig. 7. Comparison between rate of items sold and seller’s maximum number of items.
the traditional method in the situation of a free market on the Internet.

4.3. Examples of user interfaces

Figs. 8–10 show examples of the user interfaces of G-Commerce. The system has been implemented by using MiLog (Mobile intelligent agents using Logic programming) [15]. MiLog has been implemented by using Pure Java. To realize an efficient intelligent agent development environment, Milog provides a hybrid programming environment in which an agent can be designed by logic programming and Java programming. In MiLog, we can create agents that have web-service and access functions. A MiLog agent can behave as a CGI program and can access other web servers via HTTP.

The window in Fig. 8 shows a user interface for inputting a user’s preferences. A user can access his agent via a web browser. A MiLog agent has a function that enables it to behave as a web server. We utilize this function to allow users to access their agent. A user can input his desired item and a reservation price in the text box in the middle of the interface.

The window in Fig. 9 shows a user interface for displaying the current status of purchasing. Via this interface, users can see which item is being monitored, which item is on sale, and which item is being purchased. Also, a user who wants to sell some items can input information about them via a web browser. If an agent is able to make an agreement, it provides the details of the agreement to the user in order to ask the user whether it should actually make a contract.

Fig. 10 shows a user interface for presenting connections among agents. In the left window in Fig. 10, a user can monitor other agents who are known

![Image](image_url)
by the user’s agent. Once agent A connects to another agent B via a P2P connection, we say that agent A knows agent B. Here, “galant,” “summer,” “terai,” “spring,” and “nexia” are negotiating. In the right window in Fig. 10, a user can check user’s names and addresses of agents who are known by the user’s agent.

5. Overview of related works

In this section, we present an overview of the work of others related to our study. AuctionBot, eMediator, Kasbah, FishMarket, Tete-A-Tete, and GroupBuyAuction can be classified as agent-based electronic markets. Our system is also in this category. AuctionBot [21] is an auction server. Users can create auctions to sell their items. In auctions, agents can bid according to a pre-defined protocol. AuctionBot provides an API for users to create agents. Kasbah [3] provides a marketplace on the Web. Users can create agents that buy and sell in the marketplace. On Kasbah, deals between agents are conducted based on a simple protocol. FishMarket [17] provides an electronic auction site. Users can encode several bidding strategies to their agents. Although FishMarket is an experimental system, virtual tournaments have been conducted on it several times. Tete-A-Tete [8] provides an electronic market. On Tete-A-Tete, agents cooperatively negotiate with each other based on arguments. eMediator [18] is an electronic commerce server and consists mainly of eAuctionHouse and eCommitter. eAuctionHouse is a configurable auction place that supports many auction types. eCommitter is a leveled commitment contract optimizer that can solve the Nash equilibrium thresholds. Group-BuyAuction [22] is an electronic market on which agents automatically negotiate with each other on behalf of their users. In particular, in the Group-
BuyAuction, buyer agents can form coalitions in the difference between our approach and that of Group-BuyAuction is the exchanging mechanism among seller agents.

Sherlock 2, AuctionWatch, BargainFinder, ShopBot, Jango, BiddingBot, Preist’s work, and Anthony’s work can be classified as shopping information gathering agents. Apple’s Sherlock 2 is a meta-search engine which can access several search engines on the Internet. Additionally, Sherlock 2 can search for a desired item at online auction sites. AuctionWatch [2] is a search engine for items in several auction sites. Users can search for their desired items by providing keywords. BargainFinder [4] is a shopping agent for on-line price comparisons. Given a specific item, BargainFinder requests prices for it from pre-specified merchant Web sites. ShopBot [5] evolved from BargainFinder. ShopBot can automatically determine how to represent information and queries on the online merchant sites. Jango [6] is an advanced ShopBot that helps a user decide what to buy and where to buy it. BiddingBot [10] is one of the shopping support agents that can actually attend, monitor, and bid in real auction sites.

For BiddingBot we have proposed several cooperation [14] and Anthony [1] each proposed a single autonomous agent that can participate in simultaneous multiple auctions.

6. Conclusions

In this paper, we proposed an exchanging mechanism among seller agents for the agent-mediated electronic market, G-Commerce. On G-Commerce, seller agents and buyer agents negotiate with each other. In our model, seller agents cooperatively negotiate in order to effectively sell goods in stock. Buyer agents cooperatively form coalitions to buy goods based on discount prices. By employing an exchanging mechanism, seller agents can effectively exchange their items in stock. Consequently, they can sell more items as a group. We implemented an agent-mediated electronic market based on the negotiation scheme we proposed. The results of our experiment demonstrated that our exchanging mechanism can effectively facilitate agents’ deals. Furthermore, we demonstrate the Pareto optimality of
our exchanging mechanism. Since our exchanging mechanism can satisfy Pareto optimality, our agents can rationally reach an agreement in our electronic market. In the current exchanging mechanism, we can guarantee Pareto optimality between two agents. Our method only finds solutions locally. In future work, we need to develop a mechanism that can reach a global Pareto optimality. If we can use a central agent or server, it is easy to compute a global Pareto optimality. However, in a P2P network, a central agent or server should not be employed. Thus, a novel negotiation mechanism is needed to solve this problem. Furthermore, we assume that agents are software programs and can trade on behalf of their users. Accordingly, at this stage, we show a method for such agents to automatically trade on the Internet. As a future work, other factors for negotiation, e.g., price negotiation, will be needed.

References