An efficient multilateral negotiation system for pervasive computing environments

Sanghyun Park*, Sung-Bong Yang

Department of Computer Science, Yonsei University, Seoul 120-749, Republic of Korea

Received 22 August 2006; received in revised form 3 June 2007; accepted 16 July 2007
Available online 29 August 2007

Abstract

In this paper we propose an automated negotiation system that can efficiently carry out multilateral negotiations with multi-attributes in pervasive computing environments. For the multilateral negotiation system proposed in our study, an e-commerce framework for pervasive computing environments in which the components can dynamically join and disjoin a virtual market is also introduced. This framework overcomes the locality of pure ad hoc networks and supports efficiently multilateral negotiation models in pervasive computing environments.

In order to achieve the applicability toward multilateral negotiations, the concept of a mediator agent and the bilateral negotiation scheme based on linear programming is utilized for the proposed system. The experimental results show that the proposed system produces higher joint profits in negotiations about 5% and is about 96 times faster in reaching agreements on the average under the condition of agreement for reciprocity than a negotiation system based on the trade-off mechanism.

Keywords: Multilateral negotiations; Electronic commerce; Pervasive computing; Linear programming; Jini architecture

1. Introduction

Automated negotiations among multiple participants have been researched as one of the prominent fields in e-commerce (Kraus, 2001; Jennings et al., 2002). Most negotiation systems in e-commerce, however, have not been evolved sufficiently at a level of robust systems applicable to commercial web sites, especially to the field of multilateral negotiations. Such underdevelopment is due to the fact that the attributes, which may influence the behavior of the participants, cannot be defined precisely and that the success of a negotiation is dependent not only on the price but on other factors such as diverse personal inclination of the participants. Moreover, multilateral negotiations are more complicated and more time-consuming than bilateral negotiations due to the consideration of multilateral matching among the participants.

*Corresponding author. Tel.: +82 2 2123 2717; fax: +82 2 365 2579.
E-mail address: psh@cs.yonsei.ac.kr (S. Park).

0952-1976/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.engappai.2007.07.005
negotiation systems in Faratin et al. (1999), however, did not seem to consider the execution time in reaching agreements for negotiations and their applicability toward multilateral negotiations. For adaptive agent-based negotiations, Oliver (1996) showed that agents could learn strategies using a genetic algorithm-based learning technique and Oprea (2002) suggested a negotiation system that uses a feed forward artificial neural network as the learning ability of a negotiation model in the context of agent-based e-commerce. These studies have shown satisfying results on bilateral negotiations under long-term deadlines. But their systems do require longer time intervals to obtain better profits. Kurbel and Louchko (2005) proposed a multi-lateral negotiation model of agents with fuzzy constraints on an electronic marketplace for personnel acquisition. However, multi-issue negotiation with fuzzy constraints on both parties requires more information from a participant and makes it more difficult to explain the strategies to the participant. That is, the more intelligent the agents are, the more complex user interface is required. Lopes et al. (2004) presented a generic agent negotiation model for multi-party and multi-issue. Their model paid attention to the problem of integrating an individual behavior with a negotiation model and focused on the negotiation strategies that are motivated by human negotiation procedures. However, their negotiation model does not consider the environments of negotiations such as ad hoc networks or ubiquitous computing networks. Moreover, they plan to perform the experimental validation of the model which handles two-party multi-issue negotiation as future work. As the technologies advance, the e-commerce environments have been diversified from the traditional wired and/or wireless networks to short range ad hoc networks, and they are shifting towards the pervasive computing environments (Ratsimor et al., 2004). An ad hoc network can be defined as a network without central controls and connections to the outside world. By pervasive computing we mean that computing power is available everywhere at any time. In e-commerce, for pervasive computing environment, dynamic networking such as numerous and unpredictable connections and/or disconnections of participants should be allowed, and the limitation of the physical area in an ad hoc network should be overcome. Therefore, a compound network consisting of wired and/or wireless networks and ad hoc networks is appropriate for a pervasive computing environment in e-commerce. In a pervasive computing environment, an automated negotiation system requires efficient negotiation processes and the facilities of discovering services (Shenoi et al., 2006). With regard to service discovery in a pervasive environment, Ratsimor et al. (2004) proposed a policy-driven service discovery framework suitable for ad hoc networks. Their framework can avoid common problems, such as leader selection, associated with structured compound formation. However, the emerging e-commerce, including mobile commerce, cannot be based only on pure mobile networks, especially on ad hoc networks. In the recent work of Kurkovsky and Harihar (2006), they built a fully functional prototype of the framework implementing the interactive mobile marketing. They proposed an adaptive and personalized m-commerce application designed to deliver targeted promotional information to the participants. They also presented a study demonstrating end-user usability of the framework. However, their framework is based and focused on interacting between a buyer with a mobile device and a server installed at every participating retail location providing information about the products relevant to buyer’s shopping preferences. It is close to a context-aware recommendation system for mobile users rather than an automated trading system in the pervasive computing environments.

In this paper, we propose a negotiation system that can carry out multilateral negotiations efficiently with multi-attributes for pervasive computing environments. The proposed system adopts the concept of confidential mediators for guaranteeing fairness in negotiations. A mediator receives offers from buyers and sellers, while buyers and sellers do not expose their offers to their counterparts. The proposed multilateral negotiation system is based on a bilateral negotiation scheme in which a linear programming model is exploited for finding a buyer–seller pair with better profits. For the proposed system, we also introduce a framework for a pervasive computing environment in which the components can dynamically join and disjoin a community network. This framework overcomes the locality of pure ad hoc networks and supports efficiently multilateral negotiation models in pervasive computing environments. In order to evaluate the performance of the proposed system, the bilateral negotiation system presented by Faratin et al. (2000) has been implemented. In addition, we have extended Faratin’s system so that it can perform multilateral negotiations. The experimental results show that our system produces higher joint profits in negotiations about 5% and carries out about 96 times faster on the average than Faratin’s system.

The remainder of the paper is organized as follows. In Section 2, we introduce an e-commerce framework for a pervasive computing environment. Section 3 presents a multilateral negotiation system for a virtual market, which is based on the framework proposed in Section 2. Section 4 describes the simulation environment of a trading model and the experimental data sets for the participants in a negotiation. Section 5 provides the experimental results. Finally, the conclusions and future work are given in Section 6.

2. An e-commerce framework for a pervasive computing environment

2.1. An overview of a system structure

In this section we introduce an e-commerce framework for a pervasive computing environment. Fig. 1 illustrates an e-commerce structure with dynamic federations of
lookup services in a pervasive computing environment. By *dynamic federation* we mean that the members of a federation are dynamically determined according to the properties of the requested services such as the item of trading. In order to find other lookup services, a peer-to-peer (P2P) network is adopted as a communication backbone because of its scalability of networking. We named our framework ‘P2P-based lookup server’, a PLUS for short. In the system structure of Fig. 1, each PLUS can federate dynamically with other PLUSes. A PLUS provides its own services for other PLUSes within the same federation, and requests services to them as well. A PLUS may become a member of other federations simultaneously. A PLUS has a local service community consisting of mobile and/or static devices. Such devices may include PDAs, smart phones, laptop computers, desktop computers, etc.

In this structure, a task agent on a device plays as a surrogate for its corresponding negotiation participant such as a buyer/seller or a mediator. The scope of the structure as shown in Fig. 1 may be composed of a sub-network of a LAN, a LAN itself, or a WAN. In this paper, we apply PLUS to a multilateral e-commerce system although it can be applicable to various pervasive computing systems such as u-hospital (Carrillo-Ramos et al., 2005).

### 2.2. The framework of a PLUS

A PLUS consists of three layers—the utility agent layer, the service repository layer, and the infrastructure layer as shown in Fig. 2. The utility agent layer of a PLUS assists the task agents on the devices to connect with the PLUS and to register and/or receive the services. For these purposes, in the utility agent layer there are two utility agents, the service connection agent and the federation agent. The service connection agent registers the services of some task agents (as the service providers) into the service repository layer of the same PLUS and provides the services for some task agents (who have requested services) of the devices. The service connection agent maintains a list of services available in the PLUS, called the service list, and searches its service list to provide the task agents with their requested services. When the service connection agent fails to find a requested service, it asks the federation agent to find it. The federation agent keeps its own location list which has the location information such as the URLs of
other PLUSs and searches other PLUSs listed in its location list for the service. When a requested service is not available in any PLUS listed in the location list, the federation agent can request a service to neighboring PLUSs through a P2P protocol. The location information of newly found PLUS is added into the location list by the federation agent. Another main role of the federation agent is to make a federation with other PLUSs. The services in a federation are available to all the task agents on the devices in the same federation.

The service repository layer provides the lookup services for the service connection agent in the utility agent layer. Any of the lookup services such as Jini, universal plug and play (uPnP), and service location protocol (SLP) can be adopted for this framework. The service repository layer finds and registers the services that the service connection agent requests. The infrastructure layer is based on the platforms such as virtual machines, hardware for networking, or the hardware resources of the PLUS framework.

3. A negotiation system based on PLUS

3.1. Evaluating the profits

The profits of the participants, buyers or sellers, are quantified with numerical values in order to measure the degree of satisfaction of the participants. In this paper, the multi-attribute utility theory (MAUT) (Barbuceanu and Lo, 2000) is applied to evaluate the profits of the participants, considering multiple attributes of the merchandise. Each attribute of the merchandise has a weight indicating a relative preference to each of other attributes. The utility function in MAUT is expressed in terms of the weights of the attributes and the evaluation function at the offer values of the attributes, where the offer value of an attribute is defined as the value that a participant offers for the attribute in a contract. Therefore, a contract of a participant that consists of the offer values of all the attributes can be regarded as a ‘proposal’ in the real-world negotiations. The value of the utility function can be regarded as the profit of either a buyer or a seller. The utility function Profits($x_i$) of a participant can be expressed as follows:

$$\text{Profits}(x_i) = \sum_{i=1}^{n} w_i \cdot E(x_i), \quad \sum_{i=1}^{n} w_i = 1,$$

where $n$ is the number of attributes, $x_i$ is a variable representing the offer value of the $i$th attribute, $w_i$ is the weight of the $i$th attribute, and finally the evaluation function $E(x_i)$ of the $i$th attribute is expressed in terms of the request values (request_value) and the allowable values (allowable_value) as follows:

$$E(x_i) = \frac{x_i - \text{allowable_value}_i}{\text{request_value}_i - \text{allowable_value}_i},$$

where the allowable value means the maximum value to which the participant concedes for negotiation, and the request value means the maximum value the participant wants in negotiation. For example, for the price attribute of a product, if a buyer wants to pay $800 for the product, yet the buyer may pay up to $1000 for it, then the request and the allowable values for the buyer are $8000 and $1000, respectively. In Eq. (2), allowable_value, and request_value, are the allowable and the request values of the $i$th attribute, respectively. If $x_i = \text{request_value}_i$, the degree of satisfaction of the $i$th attribute is the highest, and $E(x_i)$ representing the degree of satisfaction of the $i$th attribute becomes 1. On the contrary if $x_i = \text{allowable_value}_i$, the degree of satisfaction of the $i$th attribute is the lowest, and $E(x_i)$ is set to 0. Therefore, when $x_i$ is within the range between request_value and allowable_value, $E(x_i)$ ranges between 0 and 1. If $x_i$ is out of the range between request_value and allowable_value, $E(x_i)$ is set to either 0 or 1 depending on the value of $E(x_i)$. That is, if $E(x_i)<0$, it is set to 0, and if $E(x_i)>1$, it is set to 1.

3.2. A simple formulation of a bilateral negotiation

We propose a model of a bilateral negotiation considering the efficiency of a negotiation and the extensibility toward multilateral negotiations. In the proposed bilateral negotiation model, the concept of confidential mediators is adopted for guaranteeing fairness in negotiations. A mediator agent receives the negotiation information from the participants, a buyer and a seller, in a bilateral negotiation and carries out a negotiation between a buyer and a seller who have the common negotiation ranges (CNRs) of all the attributes of the product. The negotiation range of the $i$th attribute means the range between request_value and allowable_value, for a participant and the CNR of a seller and a buyer for an attribute denotes the range overlapped between the negotiation ranges of both parties for the attribute. A mediator agent can finally find an agreement for reciprocity between both parties. Reciprocity means the reciprocal profit between a buyer and a seller, and an agreement for reciprocity means an agreement that is not partial to one participant over the other within a small allowable range.

In this paper, we exploit linear programming to solve the problem of finding an agreement for reciprocity in a bilateral negotiation. The objective function for a linear programming is established by maximizing the joint profit which is the summation of the profits of both a buyer and a seller, because each bilateral negotiation is to be carried out under the conditions of agreement for reciprocity, and because the value of the profit of a participant is a nonnegative real number between 0 and 1. Note that other operators like the multiplication of both profits may not return proper values for evaluating an agreement. The constraint condition is that the difference between the profits of the buyer and the seller should be smaller than a
threshold value, denoted as $\delta$. For the sake of fairness for both participants, the value of $\delta$ would better be as small as possible. However, as the value of $\delta$ gets smaller, the possibility of existence of the agreements which satisfy the possibly tight conditions on negotiations gets lower and lower. The performance of solving a linear programming model in the proposed system is not affected theoretically by any variations on the threshold value, and it is a matter of choice for the user of the negotiation system to select a proper threshold value according to the characteristics of negotiations. In this paper, considering the meaning of reciprocity and the agreement ratio of negotiations simultaneously, the threshold value is set to 0.01 for the negotiation systems, which means that the partiality of the profits between a buyer and a seller is not greater than 1%. The boundary conditions in a negotiation can be decided by the CNRs of all the attributes of the product. The following describes a linear programming model of the problem of finding an agreement for reciprocity:

**Procedure:** The linear programming formulation of a bilateral negotiation.

**The objective function:**

Maximize \[ z = \text{Profits}^{\text{buyer}}(x_i) + \text{Profits}^{\text{seller}}(x_i). \]  

(3)

**The constraint conditions:**

\[ |\text{Profits}^{\text{buyer}}(x_i) - \text{Profits}^{\text{seller}}(x_i)| \leq \delta. \]  

(4)

**The boundary conditions:**

The lower bound of \( \text{CNR}_i \leq x_i \leq \) the upper bound of \( \text{CNR}_i \) 

\[ (i = 1, 2, \ldots, n), \]  

(5)

where $z$ is the optimal objective value, $\delta$ indicates the threshold for reciprocity, $n$ is the number of attributes of the product, $\text{Profits}^{\text{buyer}}(x_i)$ and $\text{Profits}^{\text{seller}}(x_i)$ represent the profits of the buyer and the seller, respectively, and $\text{CNR}_i$, the CNR of the $i$th attribute, can be expressed as follows:

\[ \text{CNR}_i = \{\text{Negotiation\_range}_i \text{ for buyer} \} \cap \{\text{Negotiation\_range}_i \text{ for seller} \}. \]  

(6)

Recall that the negotiation range of the $i$th attribute, Negotiation\_range$_i$, means the range between request_value and allowable_value for either a buyer or a seller.

From the above linear programming model, a solution, the set of variables $x_i$, satisfying the above conditions is an agreement for reciprocity in a bilateral negotiation. If an offer value is expected to be an integer (e.g. the price of a product), the offer value can be rounded off the fraction.

### 3.3. The proposed multilateral negotiation system

#### 3.3.1. The components of the proposed multilateral negotiation system

The negotiation system for a pervasive computing environment consists of a virtual market and task agents on the devices of the participants. In the system, the role of a task agent is a surrogate of a participant such as a buyer, a seller, or a mediator. A virtual market in our study is a place where the client agents—buyer and seller agents—can find appropriate mediator agents, and where the mediator agents register their proxies. A virtual market can be implemented with a PLUS as shown in Fig. 3. In the utility agent layer of a PLUS, the utility agents are implemented on Java Agent DEvelopment framework (JADE), which is one of the foundation for intelligent physical agents (FIPA) compliant platforms and is a convenient tool for agent application development. The service repository layer of a virtual market has been implemented to the Jini architecture (Waldo, 2001) although other service discovery architectures such as uPnP and SLP may be adopted in the framework. The Jini architecture is a simple infrastructure for providing arbitrary services in a network and for creating spontaneous interactions between the objects that use these services. In the Jini protocol, both a service provider and a service requester can find dynamically the location of the lookup service (Jini LUS) through a discovery protocol. In addition, Jini can support the development of a protocol-independent distributed system. Service proxies can communicate with a service provider with Internet inter-ORB protocol (IIOP), object remote procedure call (ORPC), remote method invocation (RMI), or any other distributed protocols. Therefore, Jini has advantages of location- and protocol-independent distributed architectures and of dynamic join and exits of services with no impact on the network, in contrast to other distributed technologies such as common object request broker architecture (CORBA), RMI, and distributed component object model (DCOM) (Bettstetter and Renner, 2000).

A virtual market with which a client agent initially contacts can federate with other virtual markets in case that it does not own any requested services. In the Jini architecture, the federation of Jini LUSs can be easily implemented (Waldo, 2001), which is one of the advantages of the PLUS.
of the Jini architecture. The federation can be formed in various topologies in the Jini architecture, and we utilize a simple and robust method of forming the bilateral connection between two lookup services for the proposed negotiation system. Two lookup services can be linked by registering the proxy of each lookup service. An arbitrary client agent on a Jini LUS can contact with the services of other Jini LUSs in the same federation. The client agent need not know where the service is. The structure of the proposed multilateral negotiation system aforementioned is shown in Fig. 3.

3.3.2. Procedures of participation in the multilateral negotiation system

There are two different participation procedures in the negotiation system proposed in this paper. One is for a mediator agent and the other for a client agent. Fig. 4 shows the registration procedure of a mediator agent in the negotiation system. In this figure, a mediator agent, who treats an item of trading and contacts PLUS $P_A$, registers its proxy through the service connection agent on $P_A$. The proxy of the mediator agent has the name of the item (product $K$) as a proxy’s attribute and includes the connection information such as the location (URL) and the contact point (a port number) of the mediator agent. The service connection agent on $P_A$ appends the service properties (such as the name of the item) of the mediator agent to the service list and registers the proxy of the mediator agent in the lookup service on $P_A$.

The buyer agent, who is in the vicinity of $P_A$ and wants to purchase $K$, requests the service connection agent on $P_A$ to find a mediator agent of trading $K$. The service connection agent requests the lookup service to provide the proxy of the mediator agent with the client agent in case that the service is on $P_A$ as shown in Fig. 5(a). If there is no service in $P_A$, the service connection agent on $P_A$ recognizes the absent of the service in $P_A$, and requests the federation agent on $P_A$ to find it (Fig. 5(b)). In the listed order in the location list, the federation agent requests the service connection agents on other PLUSs to find the service. When the service is found in $P_B$, the service connection agent on $P_B$ informs the federation agent on $P_A$ of the existence of the service, and then the federation agent on $P_A$ federates with $P_B$. The buyer agent, thereafter, can download the proxy of the mediator agent from $P_B$. Note that all the services on a PLUS in a federation are available to all the client agents in the same federation. The buyer agent finally can connect with the mediator agent using the connection information including the downloaded proxy. Seller agents also follow the same procedures.

3.3.3. Procedure of matching couples in the multilateral negotiation system

A client group consists of the client agents who have connected with a mediator agent. In each round of negotiations, a client group is reorganized with the clients who entered newly for the current round and the clients who have failed to reach agreements in the previous round. In multilateral trading a mediator agent must determine the final couple in the client group according to a given matching criterion. By the final couple we mean a pair of a buyer and a seller finally matched in a round. The period of one round is dependent on the properties of the product in

![Fig. 4. The service registration of a mediator agent in the multilateral negotiation system.](Image)

![Fig. 5. The schematic diagrams for a client agent’s participation in the multilateral negotiation system. (a) In the case that the service is in the current PLUS and (b) in the case that the service is in other PLUS.](Image)
a negotiation. The process of determining the final couples in a round is described below:

**Step 1**: Constructing the negotiation partners. A negotiation partner is a pair of a buyer agent and a seller agent who have the CNRs of all the attributes. Since a mediator agent finds all possible negotiation partners in a client group, a buyer agent can be related with more than one seller agent and vice versa. Therefore, a buyer or a seller agent may belong to more than one negotiation partner. The mediator agent evaluates the joint profits of the negotiation partners with the formulation aforementioned in Section 3.2.

**Step 2**: Determining the final couples in the current round. A mediator agent is designed to find negotiation partners and determines the final couples among the negotiation partners according to the matching criterions. In the case of the maximum profit criterion, the mediator agent evaluates the profits of all the negotiation partners and sorts them in the non-increasing order of joint profit. Thereafter, the final couples are determined from the sorted list, while removing buyer agents or seller agents who have already been included in the final couples. The time complexity of determining the final couples with the maximum profit criterion is \(O(N \log N)\), where \(N\) is the number of negotiation partners, since sorting takes longer time than any other operations. In the case of the maximum cardinality criterion, a mediator agent gets as many final couples as possible without concerning the profits. A mediator agent in this criterion is designed to match the couples with the Ford–Fulkerson algorithm that solves the maximum cardinality matching in a bipartite graph. It finds the final couples in \(O(NM)\) time, where \(M\) is the number of client agents who participated in the negotiation (Evans and Minieka, 1992). One of the matching criterions should be predetermined for each mediator agent according to the characteristics of negotiations such as dealing products, prices, and other attributes of a product. The mediator agents prepare the next round with the client agents who have failed to get matched along with newly entered client agents. Fig. 6 illustrates an example of a round in a multilateral negotiation.

### 4. Experimental environments

#### 4.1. A trading scenario for simulation

In this paper we have chosen used cars for trading because of their appropriateness for representing the various propensities to consume, although any commodities can be traded in the proposed multilateral negotiation system. In the trading model, each buyer wants to purchase a specific model of a used car in a virtual market where sellers wish to trade various models of used cars. There may be more than one seller with the same model in each round; however, a negotiation can still be carried on because each car may have different values for its attributes in general.

The scenario for simulations is as follows. A person wants to buy a used car for CAR_A model. The buyer now launches a buyer agent who requests its service connection agent on a nearby PLUS to find the mediator for CAR_A. The buyer agent need not know the locations of the mediator agent. The service connection agent now provides the client agent with the proxy of the mediator agent for trading CAR_A, who has in advance registered on the same PLUS. Similarly, a seller agent can find the mediator agent for CAR_A. Now the mediator agent gathers the participants for trading of CAR_A model during a fixed time interval (called the round time), and then begins to make a negotiation for the current round. The mediator agent finally matches the final couples in the current round by the negotiation procedure aforementioned in Section 3.

The typical attributes of a used car are its price, the year when it was manufactured, the mileage on its odometer, and the warranty of the car. The sellers and the buyers have their own values on each attribute such as the request values, the allowable values, and the weights of the attributes. The agent for one party does not know any negotiation information about others. Only the mediator agent receives the negotiation information from the members of a client group and carries out a multilateral negotiation in each round. In this paper the negotiation information is prepared so that each attribute could represent all possible variations. Table 1 shows a sample of negotiation information of a buyer.

![Fig. 6. An example of a round in a multilateral negotiation.](image-url)

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Request values</th>
<th>Allowable values</th>
<th>Weights</th>
</tr>
</thead>
<tbody>
<tr>
<td>Price (dollars)</td>
<td>8000</td>
<td>12,500</td>
<td>0.5</td>
</tr>
<tr>
<td>Year (year)</td>
<td>2003</td>
<td>1999</td>
<td>0.2</td>
</tr>
<tr>
<td>Mileage (miles)</td>
<td>30,000</td>
<td>50,000</td>
<td>0.1</td>
</tr>
<tr>
<td>Warranty (months)</td>
<td>24</td>
<td>12</td>
<td>0.2</td>
</tr>
</tbody>
</table>

Table 1

Sample of negotiation information of a buyer
In Table 1 a buyer wishes to purchase a car with $8000. The buyer may make a concession to a seller up to $12,500 if the seller proposes a contract that is more profitable to the buyer in the other attributes (mileage, year, or warranty) than the previous contract. Furthermore, the buyer wants a car manufactured in 2003, and may also make a concession to the seller up to the year 1999 if better offers in other attributes are proposed. Similarly, the buyer wants a car with 30,000 mile and may purchase a car with up to 50,000 mile if there are better offers in other attributes. The warranty is requested for 24 months from the buyer and is also allowed at least 12 months. The weights of the attributes shown in Table 1 indicate that the buyer conceives the price as the most important factor in purchasing a car, the manufacturing year and the warranty as the next, and the mileage as the least significant factor. A sample of negotiation information for a seller can be given similarly.

### 4.2. Generating simulation data sets

The experimental data sets of multilateral negotiation systems are randomly generated in order to simulate the real-world trading model. In the experiments, the negotiation information of sellers and buyers is randomly created from Tables 2 and 3, respectively. In Table 2, \( P_{\text{req}} \) and \( M_{\text{req}} \) are the request values of price and mileage, respectively, and they are predetermined prior to the allowable values of price and mileage. Similarly, in Table 3, \( P_{\text{aw}} \) and \( M_{\text{aw}} \) are the allowable values of price and mileage, respectively, and these values are chosen prior to the request values of price and mileage.

### 4.3. Equipments for evaluating efficiency of a negotiation system

In our experiments we simplified the structure of networks and devices as shown in Fig. 7 since we focus only on the efficiency of the multilateral negotiation systems. Note that buyer agents and seller agents may or may not belong to the same PLUS with a mediator agent because the services on PLUSs in a federation are available to all the clients in the federation.

Client agents and mediator agents can be distributed over various networks, and they may perform their duties on various devices. For the convenience of experiments, however, client and the mediator agents are simulated on desktop computers. Buyer and seller agents are created in the thread unit from the buyer agent server (BASv) and the seller agent server (SASv), respectively. Therefore, they have their own negotiation information and accomplish their duties independently without having knowledge of the negotiation information of other client agents. Each mediator agent created from the mediator agent server (MASv) can trade a single product and carries out a multilateral negotiation on multi-rounds independently.

In each round new client agents participate in the trading with the client agents who failed to be matched in the previous round. Note that a buyer agent may be related with more than one seller agent and vice versa as mentioned in Section 3.3.3.

### 5. Experimental results

In this paper, we compare the bilateral negotiation system presented by Faratin et al. (2000) with the proposed system for bilateral negotiations. In addition, we have also extended Faratin’s system so that it can perform multilateral negotiations and call it the multilateral trade-off system (MTOS) because it is based on the trade-off system. The proposed system is called MPS, because its matching criterion is maximum profits. Although the agent systems based on the trade-off mechanism are not limited to the linear evaluation function, \( E(x_i) \), we compare the negotiation results under the same experimental conditions of linear evaluation function used in Faratin et al. (2000) for fair comparisons. In the comparison of the negotiation systems, we focus on the efficiency of a system with respect to joint profit, execution time, and the capability of extending toward multilateral negotiations in a virtual market.

#### 5.1. The negotiation procedure in the previous negotiation system

A trade-off can be defined informally as the mechanism in which one party lowers its values on some attributes and demands more on other attributes at the same time. Therefore, a trade-off helps in searching for a contract that is equally valuable to the current contract, and such contract may benefit the other party as well. In a bilateral negotiation, let \( X \) be an offer of one party (agent \( q \)) to the other party (agent \( p \)), and \( Y \) be a subsequent offer from agent \( p \). A trade-off for agent \( q \) with respect to \( Y \) can be defined as follows:

\[
\text{tradeoff}_q(X, Y) = Z,
\]

#### Table 2

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Request values</th>
<th>Allowable values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Price (dollars)</td>
<td>( 22000 \leq P_{\text{req}} \leq 32000 )</td>
<td>( [P_{\text{req}} - 6000, P_{\text{req}} - 3000] )</td>
</tr>
<tr>
<td>Mileage (1000 mile)</td>
<td>( 100 \leq M_{\text{req}} \leq 150 )</td>
<td>( [M_{\text{req}} - 70, M_{\text{req}} - 10] )</td>
</tr>
<tr>
<td>Warranty (months)</td>
<td>( [2, 6] )</td>
<td>( [12, 36] )</td>
</tr>
</tbody>
</table>

#### Table 3

The data template for generating buyers

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Request values</th>
<th>Allowable values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Price (dollars)</td>
<td>( P_{\text{aw}} - 6000, P_{\text{aw}} - 3000 )</td>
<td>( 22000 \leq P_{\text{aw}} \leq 32000 )</td>
</tr>
<tr>
<td>Mileage (1000 mile)</td>
<td>( 100 \leq M_{\text{aw}} \leq 150 )</td>
<td>( 100 \leq M_{\text{aw}} \leq 150 )</td>
</tr>
<tr>
<td>Warranty (months)</td>
<td>( [2, 6] )</td>
<td>( [2, 6] )</td>
</tr>
</tbody>
</table>
where \( q \) means agent \( q \), which is either a buyer agent or a seller agent, \( Z \) is the offer that satisfies \( \text{Profits}^q(Z) = \text{Profits}^q(X) \) and is assumed to be the offer that is the most similar to \( Y \). Several negotiation systems have been implemented with the various strategies of the trade-off mechanism. Among these systems, the strategies using the personal negotiation information of other party could guarantee higher profits for both parties in a negotiation (Faratin et al., 2000). Therefore, when the similarity is evaluated with the information of the other party such as the weights of attributes, an agreement of the two can achieve higher profits.

In a trade-off system, although the weights of the attributes for one party need not be open to the other party, it must be exposed to the other party during negotiation process for much higher joint profits. In MPS, however, such information is exposed only to the mediator agent. In the online environment, the concept of negotiations with a trusted mediator is often more useful than that of direct negotiations with unspecific persons (Limthanmaphon et al., 2000).

### 5.2. The results of comparison in bilateral negotiations

In this section the experiments for MPS and MTOS have been carried out in order to compare the performances of bilateral negotiations. Table 4 shows the results of bilateral negotiations with 50 negotiation partners randomly generated from Tables 2 and 3. In MTOS the hill climbing search has been utilized for finding possible contracts. Hence, its execution time and joint profits vary according to the number of sampling points. For MTOS 10, 100, and 200 sampling points have been used.

<table>
<thead>
<tr>
<th></th>
<th>MTOS (10)</th>
<th>MTOS (100)</th>
<th>MTOS (200)</th>
<th>MPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average joint profit</td>
<td>1.197</td>
<td>1.206</td>
<td>1.209</td>
<td>1.268</td>
</tr>
<tr>
<td>Average execution time for a bilateral negotiation (ms)</td>
<td>38</td>
<td>366</td>
<td>753</td>
<td>4</td>
</tr>
</tbody>
</table>

In Table 4 the execution time and the joint profits of MTOS increase as the number of sampling points increases. MPS achieves more profits than MTOS with 5% on average, and the average execution time of MPS is about nine times faster than MTOS with 10 sampling points. The overall performance of MPS is much better than MTOS with any number of sampling points.

### 5.3. The results of comparison in multilateral negotiations

The process of multilateral negotiations and the matching criteria of MTOS are established as in MPS. In MTOS, the joint profit of each negotiation partner is determined by bilateral negotiations of the trade-off mechanism. Note that the network overheads such as network latency, the cost of federations among PLUSs, and the communication cost between agents and PLUSs are the same for both MPS and MTOS. Therefore, the comparison of the dynamic capabilities of the negotiation systems was not made in this paper. In the simulation the clients who failed to be matched in the previous round and newly generated clients participated in multilateral negotiations for 10 rounds. Note that the numbers of clients who fail to be matched in each round are different for MPS and MTOS, although the numbers of newly generated clients in a round are the same for both systems. The buyers and sellers are newly generated with half of the number of clients in Table 5, respectively, except for 30 clients. In case of 30 clients, the numbers of buyers and sellers are 20 and 10 (or 10 and 20), respectively, as shown in superscripts in Table 5. The elapsed rounds mean the number of rounds needed for a client to reach an agreement. For example, average elapsed
rounds 1.30 means that the final couples have reached their agreements after 1.30 rounds on average. It is natural that in some cases the average elapsed rounds of MPS may be slightly higher than those of MTOS, since each negotiation system may have different matched couples in each round. The results of joint profits show that MPS achieves more profits (with 2.4% on average) than MTOS regardless of the number of clients.

The curves in Fig. 8 show the traces of the average execution time every 10 rounds for MPS and MTOS. In this simulation 10 clients (five buyers and five sellers) who newly entered in each round and the clients who failed to be matched in the previous round participated in negotiations during 100 rounds. We considered the execution time needed for only multilateral negotiations without considering the network latency. As the rounds are going on, the execution time of MTOS becomes slow and differs greatly from that of MPS. In MTOS, moreover, the execution time varies severely in each round due to the nature of the heuristic approach in determining the joint profit of negotiation partners. MPS performs a round about 10 times faster on average than MTOS (10) over 100 rounds.

6. Conclusions and future work

In this paper, we have proposed an automated negotiation system that can efficiently carry out multilateral negotiations with multi-attributes in pervasive computing environments. In order to achieve the applicability toward multilateral negotiations, the concept of a mediator agent and the bilateral negotiation scheme based on linear programming has been proposed. For the proposed multilateral negotiation system, we have also introduced a framework of a pervasive environment. In this framework the components can dynamically join/disjoin a virtual market which may make a federation with other virtual markets. The framework supports efficiently a multilateral negotiation model in pervasive computing environments.

For experiments the trade-off negotiation system has been implemented and compared with our system for the efficiencies of negotiations under the same experimental condition of linear evaluation function. The system proposed in this paper produces higher profits in negotiations and is much faster in execution time than the trade-off system under the condition of agreement for
reciprocity. Although our system had been implemented by assuming the linearity of evaluation function, our system improved the efficiency of the joint profits and the execution time with the extension toward multilateral negotiations in the pervasive computing environments.

The issues in relation to the confidence of mediator agents and the development of delicate protocols for agent interoperability will be included in the future work. In addition, we plan to proceed with more elaborate modifications of the framework for application to practical systems of the pervasive computing environments, such as u-hospital.

References


Sanghyun Park received the B.S. and M.S. degrees in Mechanical Engineering from Yonsei University, Seoul, Republic of Korea, in 1996 and 2001, respectively. During 1996–1999, he stayed in Aerospace Division, Korean Air, and worked as network and system administrator for Computer Aided Design/Engineering. He is currently a Ph.D. candidate at the Department of Computer Science, Yonsei University, and his research interests include e-commerce, intelligent mobile agent, and ubiquitous network.

Sung-Bong Yang serves as a faculty member at the Department of Computer Science, Yonsei University, Seoul, Republic of Korea, and his research interests include e-commerce, mobile system, peer-to-peer computing, and 3D computer graphics. He received the Ph.D. degree in Computer Science from the University of Oklahoma, USA, in 1992.