

REFLECTING THE PERSPECTIVES OF MULTIPLE AGENTS IN DISTRIBUTED REASONING FOR CONTEXT-AWARE SERVICE

SEUNGWOK HAN AND HEE YONG YOUN

*College of Information and Communication Engineering, Sungkyunkwan University,
Suwon, 440-746, Korea*

E-mail: lovedonny@skku.edu, youn@ece.skku.ac.kr

Received 1 April 2011

Accepted 21 February 2013

Abstract

Effective manipulation of context is very important in providing the context-aware services. In recent years, a variety of context models have been proposed to properly handle the key aspects of the context, while focusing on scenario-based acquisition, management, and representation of context. However, they are difficult to be employed for the agent-based system requiring distributed reasoning. In this paper we propose a context modeling approach for distributed reasoning and merge operator reflecting the perspective of constituent agents for rational reasoning. In addition, an agent-based context-aware system is developed implementing the proposed scheme. Performance evaluation by computer simulation on a use case of smart classroom shows that the proposed approach enables the agents to rationally reason and thereby provide intelligent context-aware services to the users.

Keywords: agent; context modeling; distributed reasoning; merge operator; ubiquitous computing

1. Introduction

Since the emergence of ubiquitous computing paradigm, context-awareness has been considered as one of the key ingredients realizing the paradigm. The objective of context-awareness is to provide optimized service without overburdening the user with excessive information and decision-making tasks. For supporting context-awareness, not only the user requirements but also the contexts of the environment are required to be efficiently perceived. Context-aware system has the ability to recognize the user context and operate based on that¹. It typically handles personal and subjective information, and thus the primary requirement is the rationality of the service³¹⁻³³. This largely depends on the correlation among the contexts which dynamically changes according to the situation⁷. Among various research topics on context-awareness, efficient

utilization of context information, especially context modeling, has been receiving a great deal of attentions.

The context model representing the characteristics of the situation is the cornerstone enabling context-aware service. It is used to reason the occurrence of a certain situation, and a specific event is triggered according to the result of the reasoning. In recent years, numerous context models have been proposed for specifying the acquisition, management, and relationship of contexts². The contexts are usually handled by a centralized reasoning server. However, centralized reasoning limits the performance of the system providing the services in distributed environment, and adversely affects the agility of the system. In order to effectively provide context-aware service, thus, a novel context model based on distributed reasoning is needed.

In the literature a context modeling approach supporting distributed reasoning for covering the

perspectives of different entities and resolving the conflicts in the context was proposed³. It enables the agents to individually keep the situation model and carry out the reasoning task. It also proposes a merge operator rendering a richer situation model from the basic model formed with the fragmentary information on the situation. Here the agents independently reason about the context and interact with each other to raise the accuracy of the reasoning. The merge operator allows that the perspectives of respective agents participating in the merge process are not lost but preserved in the merged situation model. However, the model uses a same trust level for each model, and simply doubles the weight of importance during the merge process. It may thus generate an unclear situation model when the difference of the merged perspectives is large for the same situation. Also, it is difficult to support different individual perspectives and solve the conflict among the perspectives of the cooperative processes. Consequently, retaining the diversity of the perspectives of the original agents and taking advantages of agent-based collaboration via distributed reasoning are difficult.

In this paper, thus, we propose a new modeling methodology for distributed reasoning based on the perspectives of constituent agents. It enables the agents to rationally reason using the consented situation model representing the agreed perspective of them, which solves the conflict among different perspectives. In the process of making a consented situation model, reflection rate is determined by the importance of the agents employed for rational reasoning. A weight is assigned to each context in proportion to the characteristics of the user. Indicative weight is assigned to the sub-range of the values of a context model to differentiate the importance of the sub-ranges. Also, the notion of confidence weight is employed to decide the trust level of the context information since the context value is not always trustworthy. In order to facilitate distributed reasoning, a merge operation is proposed which integrates the perspectives of the agents to build a consented situation model. In the merge process the reflection rate is allocated to the agents for making a rational consent model. Merging of different perspectives of the context model is carried out using the context weight and indicative weight. It permits reflecting the perspective of constituent agents for

rational reasoning. The proposed scheme is applied to the agent-based context-aware system¹³⁻¹⁸ developed by the authors for providing context-aware service. It applies reasoning to the situation and requests a service using the proposed model. Computer simulation with a use case of nursing home service shows that the proposed approach enables the agents to rationally reason and thereby provide intelligent context-aware service more efficiently than the earlier approaches.

The rest of the paper is organized as follows. Section 2 describes the related work, and Section 3 presents the proposed scheme. The performance of the proposed approach is evaluated in Section 4, and Section 5 concludes the paper with some remarks.

2. The Related Works

The goal of context-aware service is to provide intelligent service to the user without explicit user request. It can be achieved by designing a context model effectively supporting the collaboration of the agents, while properly reflecting the perspective of each agent.

2.1. Context-aware Computing

Context-aware computing enables the system to be aware of the user's context and adapt its behavior accordingly⁵. The state-of-the-art of context-aware research centers around the development of efficient context models and context reasoning approaches.

Context is characterized as the information that can be used to characterize the situation of an entity¹. An entity is regarded as a person, place, or object relevant to the interaction between the user and computer. Different types of models of context information have been proposed so far such as key-value, markup scheme, graphic, object-oriented, logic-based, and ontology-based model. Each of them is effective for some specific domain². Numerous existing context modeling approaches synthesize the information on the location, object, physiological readings of people, etc., for real-world applications. These approaches focus on the scenario-based acquisition, management, and representation of context. One shortcoming of them is that the measure reflecting the relative importance, accuracy, and reliability of the facts has not yet been adequately incorporated³.

Manipulation of context data in an intelligent way is one of the most challenging contemporary research tasks, and is often referred to as context reasoning⁶. A more precise definition of context-reasoning is deducing new and relevant information from various sources of context-data for specific applications or users. Context-reasoning, in the hierarchy of context view, makes it possible to map raw context data called low-level context to higher level contexts. Context reasoning can be applied in many ways. One way is to apply ontology to the process and utilize logical reasoning for the mapping⁸. The other is probabilistic reasoning, which employs Bayesian network to produce extensible probabilistic models used in the mapping phase²⁷.

The process of context-awareness consists of three main tasks; perception, interpretation, and execution². In the first task of perception, the low-level context obtained by sensors, RF-ID, or user input is taken as raw data. The second task of interpretation derives higher-level context using the inference and learning engine. Finally, the inferred context is delivered to the service to allow intelligent operation. The second task is performed like a human recognition process, and various approaches have been proposed to effectively interpret the user context. Unfortunately, there is no generic algorithm capable of robustly recognizing the situation using the information supplied from various sources.

2.2. Agent Technology

Agents can be defined as autonomous, problem-solving computational entities capable of effective operation in dynamic and open environments. The typical characteristics of an agent are autonomy, intelligence, mobility, and social ability. Additional characteristics are reactivity for responding to the change in the environment, veracity for prohibiting wrong information, and rationality for supporting rational method. The agent system can be classified into multi-agent system and mobile-agent system. The multi-agent system handles complicated operations requiring the collaboration of the agents. In mobile-agent system the agents move through the network to process the tasks, and it is widely used for mobile computing applications⁴. An agent system could be both multi-agent system and mobile agent system.

An agent in the multi-agent system is a kind of computer program cooperating with other agents in the distributed computing environment. By collaborating with other agents, it can provide the users with sophisticated services, which is impossible with only a single program. The agents provide the designers and developers with a way of developing the applications using autonomous communicative components, and lead to the construction of software tools and infrastructure supporting the design^{9,30}.

The agent-based technology allows the services to show 'human like' autonomic behavior using the context or situation information. Recently, social agent-based planning and cooperative work have been proposed for sharing common goals along with individual perspectives. They can share ontology, knowledge, and preferences of the constituent members. Here the agents are employed for developing the context-aware system, in which they cooperate with each other for providing customized services.

2.3. Distributed Reasoning

Many researches on context-awareness have adopted the agent paradigm to implement the context-aware system. Each agent deployed in this system manages a specific task and collaborates with each other to achieve a common goal. Most existing approaches employ a centralized reasoning server and the client-server architecture. It is thus limited in capitalizing the essential features of agent, and suffers the problem of bottleneck and single point of failure. The more the number of agents increases, the more such limitation and problem become serious. In this aspect, enabling the agents to independently reason about their context without relying on a centralized reasoning server can be an effective solution.

With distributed reasoning the tasks performed by a centralized reasoning server are distributed to the agents so that parallel reasoning can be possible^{23-26,28}. It effectively supports dynamic environment and lets the agents capitalize the characteristics of independence and autonomy. Amir et al.³ proposed a formalized approach enabling the agents to perform distributed reasoning about the current situation. Also, they demonstrated an approach considering additional information available from other peer agents at run time in context-aware

pervasive computing system. The approaches focus on a novel unified model describing the context and situation using geometrical metaphors and the concepts derived from the state-space model. They also include a context algebra based on the model enabling distributed reasoning by merging and partitioning context models representing different perspectives of the agents over the object of reasoning. A new modeling approach and algorithm for the collaboration and distributed reasoning about the situation is still needed, however, which further aid the agents in a pervasive system for efficiently providing customized services to the users²².

3. The Proposed Scheme

3.1. Motivation

We demonstrate the motivation of the proposed approach with a use case of nursing home service. Here the agents individually manage the personal contexts and cooperate with each other based on a model. For modeling the patient’s situation, the contexts of acceptable range of various conditions are specified by domain experts. The contexts are able to be aggregated from bionic sensors, and an agent is deployed as an aggregator in the sink node of the bionic sensor with the model developed for reasoning low-level contexts²⁹.

Suppose that a number of patients are taking a group therapy in a living room. Each agent is deployed in the user’s personal devices such as PDA and smart phone. A situation model about a pleasant room condition is developed based on the perspectives of the users. The agents apply reasoning about the situation based on the user’s perspective, and request a service such as activating air-conditioner or humidifier for the user. Assume that some patients caught a cold, and thus they might feel chillier than healthy people. In this situation the perspectives of different users need to be conspicuously reflected in the service. In other words, the results of reasoning should be different from them derived with other situations, and the agents request different services accordingly.

The scenario above can be used to evaluate the effectiveness of distributed reasoning because the agents have different perspectives while the service is shared among the agents. Due to this conflicting situation, collision of service requests may eventually occur.

Various services provided in the smart space like the nursing home might often encounter such situation. Therefore, in the agent-based distributed reasoning system, a merge operator is needed to mediate the perspectives of the agents for rational reasoning. To realistically handle such circumstance, we define a situation model of living room. It handles a context model of indoor temperature, which is connected to the service activating air-conditioner. The agent reasons about the situation, and requests a service based on that. The context model represents the range of the temperature decided based on the perspective of its user.

3.2. The Modeling

A context is information that can be used to characterize the situation of the interested entity. The context-aware engine employs the situation model to infer the situation using a specified model. The agent-based context-aware service efficiently capitalizes the collaborative characteristics of the agents. Since the context is assessed from the current situation in which the users are involved, customized services can be provided to them. The situation model provides a general approach for modeling the contexts, and enables effective reasoning with individual perspective. The proposed approach lets the context model be included in the situation model as follows.

- *Situation model*: A situation model represents a real-life situation used to infer the occurrence of the situation.

$$sm_k = \{cm_1^k, cm_2^k, \dots, cm_n^k\} \tag{1}$$

where sm_k represents k -th situation model consisting of n context models. For example, assume “*Emergency situation*” consisting of three context models; $sm_1 = \{pulse\ variation, temperature\ variation, amount\ of\ activity\}$. The three context models are listed in Table 1.

Table 1. The context models of an emergency situation.

Context model	Context
Pulse variation	Amount of activity/ skin temperature/ location / average pulse rate
Temperature variation	Acceptable pulse variation/ amount of activity/ location / skin temperature
Amount of activity	The time duration of activity/ Average pulse rate/ location

- **Context:** A context, c_i , is an entity consisting of context name, cn_i , and value, cv_i , respectively.

$$C = \{c_1, c_2, c_3, \dots, c_n\},$$

$$c_i = \{ \langle cn_i, cv_i \rangle \mid \forall cn, \forall cv \} \quad (2)$$

where C is a set of a contexts. The name of a context is used as its identifier within the service domain, and the set of them is $CN = \{cn_1, cn_2, \dots, cn_n\}$. cv_i is a sensed value, and the set of them is denoted by $CV = \{cv_1, cv_2, \dots, cv_n\}$. With the definitions above, an example of C containing four contexts used for identifying the state of pulse variation is represented as follows.

$$C = \{ \langle \text{amount of activity}, 83 \rangle, \langle \text{skin temperature}, 23 \rangle, \langle \text{location information}, \text{room} \rangle, \langle \text{average pulse rate}, 83.3 \rangle \}$$

- **Context model:** The structure of context model is

$$cm_i^k = \{ \langle c_i, v_i, w_i \rangle \mid \forall c, \forall v, w \in [0, 1] \} \quad (3)$$

$$v_i = \begin{cases} v_i \subseteq CV & \text{, if } v_i \text{ is AV} \\ l_i \leq v_i \leq m_i & \text{, if } v_i \text{ is AR} \end{cases} \quad (4)$$

$$\sum_{i=1}^n w_i = 1 \quad (1 \leq i \leq n) \quad (5)$$

$$V = \{v_1, v_2, \dots, v_n\} \quad (6)$$

where v_i is defined as generally acceptable range of the value (AR) or a set of attribute value (AV). For example, the ‘skin temperature’ of healthy condition is represented as $22.5^\circ\text{C} \leq \text{‘skin temperature’} \leq 25.5^\circ\text{C}$. The context value of ‘location’ is either *room*, *dining room*, or *living room*. The AR of the smallest value and largest value of the acceptable range is represented as l and m , respectively. The weight function, w , defines the importance of context. The AR varies with the situation of user. It is obtained by

$$l'_i = l_i + l_i ap, \quad m'_i = m_i + m_i ap \quad (6)$$

$$v_i = l'_i \leq v_i \leq m'_i$$

where ap is a factor used as additional information for the AR. Table 2 lists the contexts defined for the nursing home service and AR. The AR identifies healthy condition in the situation of sleeping, normal, and exercise. The minimum pulse rate is fixed as 47.5, while the maximum pulse rate is obtained by $(220 - \text{Age}) * h$. Here h is the variable representing the health condition of the person, which is listed in Table 3. For example, the maximum pulse rate for a 50 year old user in the situation of exercise is calculated as $(220 - 50) * 0.7 = 119$. If the user has a cardiovascular disease, the maximum pulse rate is decreased further by 10%, and it becomes 107.1 ($= 119 * 0.9$).

Table 2. The contexts of the medicare service and acceptable range of the values.

Context	Range of the values
Amount of activity	100 ~ 500
Skin temperature	22.5°C ~ 25.5°C
Pulse rate	47.5 ~ (220 - Age)* h
Average pulse rate	54 ~ 99 per minute
Variation of skin temperature	Average temperature $\pm 8^\circ\text{C}$ (in 10 minutes)
Location	Room, garden, dining room, living room, toilet, church

Table 3. The health condition deciding the maximum pulse rate.

Health condition	State	Value
Current disease	Cardiovascular	-10%
Count of emergency	< 1	-5%
Timing of last emergency	< 10 days ago	-20%
	< 30 days "	-10%
	< 1 year "	-5%
Activity	Sleeping	50%
	Normal	60%
	Exercise	70%

An example of context models of emergency situation is represented in Table 4. The amount of activity is used in inferring the variation in pulse and temperature for different weights. The variation in pulse is closely related to the amount of activity and average pulse rate. Meanwhile, the variation in temperature depends on the location and the variation of skin temperature since the amount of activity does not contribute much to the change of skin temperature. The time duration in which the activity has been extended has a larger weight of 0.4 than average pulse rate of 0.3.

This is because if the user does not move for a while, an emergency situation could be inferred even though the average pulse rate indicates normal state. The data listed in Table 2, 3, and 4 are predetermined by domain experts.

Table 4. An example of context models of emergency situation.

Context model (confidence)	Context name	Context value	Weight
Pulse variation (0.89)	Activity	83	0.4
	Skin temperature	23	0.2
	Location	Indoor	0.1
	Pulse rate	83.3	0.3
Temperature variation (0.78)	Acceptable pulse variation	8.5	0.2
	Activity	83	0.2
	Location	Indoor	0.3
	Variation of skin temperature	-2	0.3
Amount of activity (0.83)	Time duration	> 5 min	0.4
	Pulse rate	83.3	0.3
	Location	bathroom	0.3

Assume that a user does not move in the bathroom for more than 5 minutes and thus the amount of activity is smaller than 100, and the skin temperature is rapidly decreasing. Then the model assesses an emergency situation (i.e., falling over in the bathroom by accident, etc.). According to the proposed approach, the confidence of the emergency situation is 0.825 with the same weight of 0.33 for each of the three contexts models.

• *Indicative weight, I*: In the context model some sub-range of the range of a context model is more indicative than other ranges. For example, for the context of “skin temperature”, the values between 22.5°C and 23.5°C is more indicative than other values in inferring the situation.

$$\tau_i(x) = I(x \in [l_i, m_i]) , I \in [0,1] \tag{7}$$

where the indicative function, τ_i , assigns a weight, I , to each range of the context model to reflect how much the context value influences the situation. l_i and m_i denote the minimum and maximum value of cv_i , respectively.

3.3. Confidence of Context Model

We have described the situation and context model employing context weight and indicative weight. With the proposed modeling approach, one context model in a situation model indicates one condition required to be satisfied for the occurrence of the situation. If the context value lies within the valid range of the context model, one condition needed to infer the occurrence of the situation can be said to be satisfied.

During the reasoning process, each context is separately evaluated. The context value is not always trustworthy, and thus the confidence on the expected performance of the sensor is determined by the knowledge already accumulated²⁰. For example, according to the performance of the sensors, the confidence about an individual user context obtained from the sensor is able to be decided as confidence interval of [0.3, 0.5] for user-A and [0.4, 0.8] for user-B, respectively. Knowledge on technical specifications such as general performance of the sensor, current working status, statistical data on the evolution of its behavior, and the performance records can be used to estimate the performance of the sensor. The expected performance should be used to decide the confidence of each sensor²¹. The confidence is applied to each sensor's probability mass. Given a probability mass function of a sensor, m_i , corresponding confidence weight, $conw_i$, the weight adjustment process is expressed as

$$m_i'(c_i) = conw_i m_i(c_i), c_i \in cm_j, conw_i \in [0, 1] \tag{8}$$

$$\mu(cm_i^k) = \sum_{i=1}^n m_i'(c_i) \tag{9}$$

where $m_i'(c_i)$ is the adjusted probability mass function for the context. μ is the confidence function applied to a context model. The confidence of a situation model in the context awareness service is obtained by summation of $\mu(cm_j)$.

Each agent might have a distinct perspective on any specific situation, which means that the agents may have different situation models for the same situation. The confidences computed by the agents also vary according to the agents' perspective. The agents recognize the occurrence of a situation when the confidence is larger than the confidence threshold,

which is predetermined by the system designer or administrator.

3.4. Merging Different Perspectives

Merging several situation models makes respective situation model richer by integrating the perspectives and outcomes of reasoning of various agents⁸. The merge operation proposed in this paper generates a consented situation model so that the agents can rationally reason about a specific situation by sharing the service provided by the system. Note that a situation model is associated with the service. The agents provide the service defined in the situation model if the confidence is larger than the confidence threshold. If the agents have different perspectives for the same situation, they will provide distinct service according to their perspectives. Also, if the agents have a situation model associated with a service shared among the agents, collision of service may occur due to different confidences.

In order to facilitate distributed reasoning, merge operation is employed which builds a consented situation model integrating the perspectives of the agents. The consented situation model avoids the conflict of service requests. Recall that the proposed modeling approach is characterized by context model, context weight, and indicative weight. Since they represent the perspectives of a situation, merging different perspectives requires merging these factors. With the merge process, a more rational situation model could be built using the reflection rate (the beliefs and capabilities of agent). For example, if the reflection rate of Agent-A is higher than Agent-B, the perspective of Agent-A is reflected more in the resultant consented situation model. In the proposed agent-based system, the service agent assigns a reflection rate to individual agents requesting the service according to the rules designed by the domain expert. For example, the air-conditioning service should reflect the perspective of weak people suffering cold more significantly than healthy people. The reflection rate of a situation is the difference between the reflection rate and the average reflection rate computed by

$$r_s = \frac{rs_s - \frac{1}{n} \sum_{i=0}^n rs_i}{\frac{1}{n} \sum_{i=0}^n rs_i} = \frac{rs_s}{\frac{1}{n} \sum_{i=0}^n rs_i} - 1, rs_{i,s} \in [0,1] \quad (10)$$

$$w'_i = r_s w_i \quad (11)$$

where r_s denotes the reflection rate of situation, S , rs_i is reflection rate of merged situations obtained by domain experts, and n is the number of context models. Using the reflection rate, the new adjusted context weight, w_i , is computed.

The merge process consists of three tasks. The first task is to compose a situation model by applying the reflection rate. The second and the last one adjust the context weight and indicative weight, respectively.

3.4.1. Composition of Situation Model

Given two context models for c_i , cm_i^k and cm_j^k , the composition of context models is processed in two phases. The first phase is to update the context models by applying the reflection rate, and the second phase is to merge the ranges of the updated context models.

In the first phase, the smallest and largest value of the merged context model, cm_m^k , is obtained by

$$cm_m^k = \left\{ \left\langle c_{i,j}, v_m, w(c_m) \right\rangle \mid \forall c, \forall v, w \in [0,1] \right\} \quad (12)$$

$$\overline{l}_{(cm_m^k)} = \overline{l}_{(cm_{i,j}^k)} + r_s \left(\overline{l}_{(cm_{i,j}^k)} - \frac{1}{n} \sum_{i=0}^n \overline{l}_{(cm_{i,j}^k)} \right) \quad (13)$$

$$\overline{m}_{(cm_m^k)} = \overline{m}_{(cm_{i,j}^k)} + r_s \left(\overline{m}_{(cm_{i,j}^k)} - \frac{1}{n} \sum_{i=0}^n \overline{m}_{(cm_{i,j}^k)} \right)$$

$$v_m = \begin{cases} v_i \cup v_j & , \text{if } v \text{ is AV} \\ \overline{l}_{(cm_m^k)} \leq v_m \leq \overline{m}_{(cm_m^k)} & , \text{if } v \text{ is AR} \end{cases} \quad (14)$$

where $\overline{l}_{(cm_i^k)}$ and $\overline{m}_{(cm_i^k)}$ denotes the smallest and largest value of the context model for c_i , respectively.

3.4.2. Update of Context Weight

The merged weight of w_m is obtained by convolution:

$$w_m = \begin{cases} \sum_{k=\max(l_i, n-m_j)}^{\min(m_i, n-l_j)} f(k)g(n-k) & , \text{ if } v \text{ is AV} \\ \int_{\max(l_i, n-m_j)}^{\min(m_i, n-l_j)} f(\lambda)g(n-\lambda)d\lambda & , \text{ if } v \text{ is AR} \end{cases} \quad (15)$$

Where $f(k)$, $g(n - k)$ and $f(\lambda)$ and $g(n - \lambda)$ are probability density functions of two discrete and continuous context weights, respectively. Using the reflection rate, the new adjusted context weight, w_m' , is obtained by

$$w_i' = w_i + r_s w_i \quad (16)$$

$$w_m' = \frac{1}{n} \sum_{i=1}^n w_i' \quad (17)$$

where w_m' is the adjusted merged weight. The new context weight is the relative weight of a context with respect to the overall weight of all the existing contexts.

3.4.3. Update of Indicative Weight

In order to assign indicative weight to the merged context, the domains of the indicative functions assigned to the context in the merge process have to be covered. The following function decides the position of the context value of the merged context model in the original context model.

$$v_{(cm_m^k)}(x) = \frac{(x - l_{(cm_m^k)}) (m_{(cm_m^k)} - l_{(cm_m^k)})}{m_{(cm_m^k)} - l_{(cm_m^k)}} + l_{(cm_m^k)}, \quad (18)$$

$(l_{(cm_m^k)} \leq x < m_{(cm_m^k)})$

For the merged context model, the new indicative weight using the reflection rate is obtained as

$$\tau_i'(x) = \tau_i(c_m) + r_s \tau_i(c_m) \quad (19)$$

The time complexity of the proposed approach is $O(n^2)$, where n is the number of contexts in the context model. This yields a new indicative weight such that the updated context model of high context weight and indicative weight has a high indicative weight.

4. Performance Evaluation

The agent-based context-aware system has emerged as a desired platform for developing a cooperative service. It effectively supports the collaboration of the agents and design of intelligent service. In order to evaluate the proposed approach, a new agent-based context-aware system is implemented which supports the collaboration of the agents with distributed reasoning. The proposed modeling approach is deployed in the system for efficient inference. The agent platform is implemented in C++ complying with the FIPA specification¹⁰. The context-aware system is implemented in Java to be integrated with the existing inference engine^{12,19}.

4.1. Agent-based Context-aware System

The proposed system adopts the holarcies¹¹ methodology, which constructs the system in the multi-level group hierarchy. It enables the goal to be recursively decomposed into multiple subtasks so that they can be assigned to each agent group. For representing the context information, the XML-based context definition is employed.

The proposed system shown in Fig. 1 consists of context-aware agent, aggregator, and service agent. The agent platform is a middleware employing the HTTP-based MTP (Message Transport Protocol) which transfers ACL (Agent Communication Language) messages between the agents. It transfers data representing the entire message including the envelope in an HTTP request³⁴⁻³⁶.

The context-aware agent obtains the context

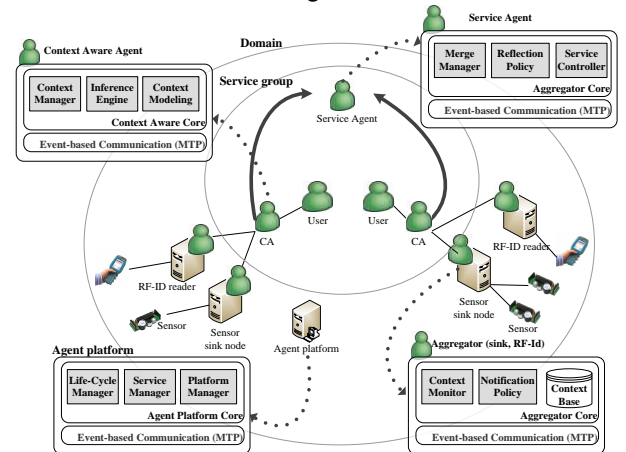


Fig. 1. The structure of the proposed context-aware system.

information obtained by the aggregator, and then deduces a high level context using Jess¹² rule engine. It also employs the proposed modeling approach. The agent-based aggregator deployed in the sensor node gathers and manages the context information for the context-aware agent. The service agent acquires the context information through the aggregator or context-aware agent, and allocates a reflection rate to the context-aware agent by the rules. It provides the service decided using the proposed merge scheme.

4.2. Case Study

We demonstrate the effectiveness of the proposed approach using a case study of nursing home service. In the proposed agent-based context-aware system, each context model is regarded as a collaborating agent. We create 100 agents and divide them into two groups; Group-A represents healthy people, while Group-B represents the cold patients. Therefore, Group-B wishes to maintain higher temperature than Group-A as shown in Table 5.

Table 5. The context models of two groups.

Parameter	Group-A	Group-B
Range of context	20~24	22~28
Context weight	1.0	1.0
Reflection rate	-0.5~0.0	0.0~0.5
Number of agents	70	30

We randomly generate 100 situation models according to the model described in Table 5. The indicative weight function of the generated context model is a quadratic function which assigns 1 to the middle value of the range and 0 to the smallest and largest value of the context model, respectively. The generated situation model is assigned to each agent. Three merge schemes are implemented for comparison. The first one is simple combination merge which merges the situation model by averaging the outcomes of reasoning processed by individual agents based on its perspective. The second and last scheme employ the consent merge scheme proposed in this paper, while reflection rate is applied only to the last one. We first evaluate the effectiveness of the three merge schemes with varying temperatures.

Fig. 2 shows the confidences of the three schemes. The proposed consent merge scheme allows the confidence to be between 0 and 1, and enables the agents to correctly identify the occurrence of the situation. However, the combination merge scheme cannot do that because it simply averages the confidences computed by each agent. Observe that the consent merge scheme implementing the reflection rate puts high confidence for relatively high temperature than the merge without it. In other words, its curve is shifted to the right compared to the other curve. This indicates that the perspectives of the Group-B patients

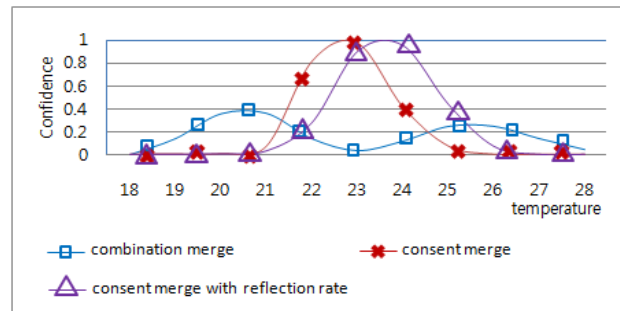


Fig. 2. The range of the confidences in merged context models.

are accounted more significantly than the Group-A during the merge process. This clearly demonstrates that the reflection rate plays an important role in merging the models.

Using the merged context models, we let the agents periodically compute the confidence weight and request a service to maximize the accuracy of the outcome of the reasoning. In the simulation the agent requests turning on or off the air-conditioner, while the initial temperature is 18 °C.

Fig. 3 depicts the change of temperature caused by the agents requesting the air-conditioning services, and Fig. 4 does the confidence weight of the agents of the merged context model. Observe from Fig. 3 that the proposed scheme maintains higher temperature than the combination merge scheme. This is expected by the confidence of the merged context model shown in Fig. 2. If the initial temperature was 28°C, the combination merge scheme would maintain the range of 25°C~ 26°C. It is somewhat ambiguous with respect to providing customized service. Fig. 3 also shows that it is possible to reflect the perspectives of specific agents using the reflection rate.

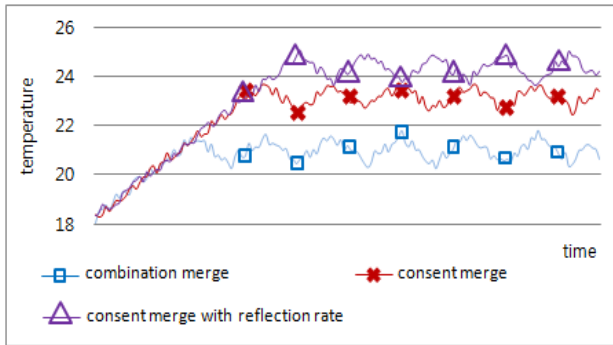


Fig. 3. The change of temperatures in the room.

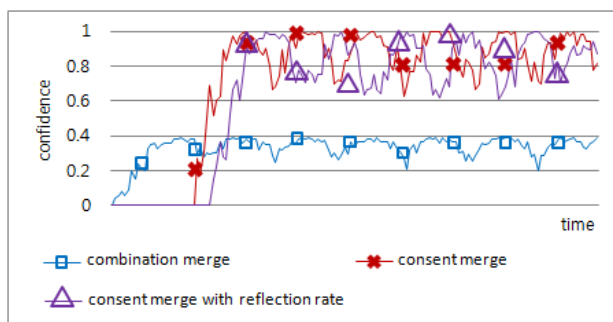


Fig. 4. The confidence weight computed by the agents.

Fig. 5 represents the standard deviation of the confidences computed by the agents. Notice that the proposed schemes show relatively smaller standard deviation than combination merge, which implies that the service is provided to all the agents as fair as possible. The standard deviation of the confidence with reflection rate is higher than that without it. This indicates that reflection rate increases the confidence of specific agents, which provides exclusive rational

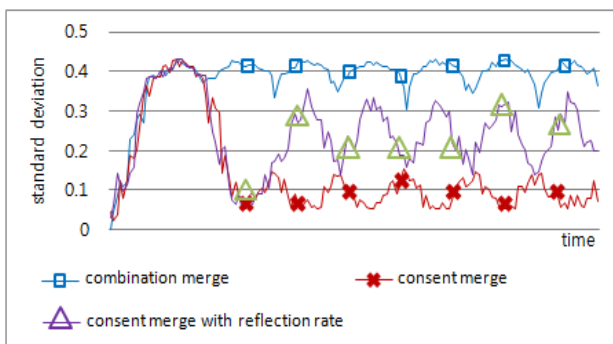


Fig. 5. The standard deviation of the confidence weights.

service to the users.

5. Conclusion and Future Work

In this paper we have introduced an approach for modeling of context-aware service, which allows efficient agent-based distributed reasoning about the situation. The proposed context modeling is able to reflect the perspective of the agents and enables the agent to independently reason about the current situation. The context model explicitly has the name, value, and weight of the context, and indicative weight decided based on the analysis of domain. The weight is applied to each context to reflect the characteristics of the user. Moreover, the concept of confidence is employed since the context value is not always trustworthy. These parameters allow effective expression of the features of the contexts.

In order to facilitate distributed reasoning, the merge operation has also been proposed which effectively integrates the perspectives of the agents to build a consented situation model. In the merge process the reflection rate is assigned to the agents, which creates a rational consented model. The merge process includes context model, context weight, and indicative weight for properly merging different perspectives of the context model. Computer simulation on a use case implemented on the context-aware system developed by the authors displayed the effectiveness of the proposed scheme. We have found that the perspectives of the agents might change according to the change of the environment or specific rules. In case of the proposed scheme, the agents change their perspectives during the merge process.

We aim to carry out the following work in the future to improve the proposed modeling approach. First, the quality and uncertainty of context information need to be elaborately handled in the model. To further increase the applicability and support fault tolerance, second, the proposed approach needs to be enhanced by explicitly employing some redundancy scheme and analyzing the satisfaction level of the service in various domains obtained from the feedback of the users. Moreover, the agent coordination mechanism is an important and interesting problem. In the proposed scheme, the conflicts are solved via merge operator using a linear model for the merging. However, for the case where the

underlying constraint satisfaction problem is extremely hard or the attributes involved nonlinearly change, the proposed model needs to be tuned. This will be investigated in the future.

We believe that context model takes a crucial role in various application domains including health care, emergency service, and prevention of catastrophe. We will investigate the effectiveness of the proposed approach for these practical but critical applications³⁷.

Acknowledgment

This work was supported in part by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology (2012R1A1A2040257), Korea Association of Industry, Academy and Research Institute (C0017380), and DAPA and ADD (UD10070MD). Corresponding author: Hee Yong Youn.

References

1. Dey, A. K., "Understanding and Using Context," Personal and Ubiquitous Computing archive, 5 (1), 2001.
2. Henriksen, K., and Indulska, J., "Developing context-aware pervasive computing applications: Models and approach," *Pervasive and Mobile Computing*, 2 (1), 2006, 37-64.
3. Padovitz, A., Loke, S. W., and zaslavsky, A., "Multiple-Agent Perspectives in Reasoning About Situations for Context-Aware Pervasive Computing Systems," *IEEE Trans. Systems, Man and Cybernetics, Part A*, 38 (4), 2008, 729-742.
4. Luck, M., McBurney, P., and Preist, C., "Agent Technology: Enabling Next Generation Computing," AgentLink community, 2003.
5. Satyanarayana, M., "Challenges in Implementing a Context Aware System," *IEEE Transactions on Pervasive computing*, 1 (3), 2002, 2-4.
6. Nurmi, P. and Floreen, P., "Reasoning in Context-Aware Systems." *International Conference on Artificial Intelligence*, 2005, 650-655.
7. Chang, C. K., Jiang, H., Ming, H., et al., "Situ: A Situation-Theoretic Approach to context-Aware Service Evolution," *IEEE Services Computing*, 2 (3), 2009.
8. Anagnostopoulos, C., and Hadjiefthymiades, S., "Enhancing Situation-Aware Systems through Imprecise Reasoning," *IEEE Transactions on Mobile Computing*, 7 (10), 2008, 1153-1168.
9. Jennings, N. R., "An agent-based approach for building complex software systems," *Communications of the ACM*, 44 (4), 2001, 35-41.
10. FIPA-Foundation for Intelligent Physical Agents, <http://www.fipa.org>
11. Horling, B., and Lesser, V., "A survey of multi-agent organizational paradigms," *The Knowledge Engineering Review*, 19 (4), 2005, 281-316.
12. The Rule Engine for the JavaTM Platform. <http://herzberg.ca.sandia.gov/jess/>
13. Park, A. H., Park, S. H., and Youn, H. Y., "A Flexible and Scalable Agent Platform for Multi-agent Systems," *Transactions on engineering, computing and technology, Enformatika*, 19, 2007, 1-6.
14. Han, S., Song, S. K., and Youn, H. Y., "Modeling and Verification of Context-awareness Service for Time Critical Applications Using Colored Petri-Net," *International Conference on Intelligent Agent Technology*, 2008, 71-74.
15. Kim, J. H., Han, S., Youn, H. Y., et al., "Prediction-based Dynamic Thread Pool Management of Agent Platform for Ubiquitous Computing," *International Conference on Ubiquitous Intelligence and Computing*, 2007, 1098-1107.
16. Kim, J. H., Chu, J. H., Youn, H. Y., et al., "Hierarchical P2P Networking Two-Level Compression Scheme for Multi-agent System Supporting Context-Aware Applications," *International Conference on Intelligent Systems Design and Applications*, 2008, 519-524.
17. Song, S. K., Han, S., and Youn, H. Y., "A New Agent Platform Architecture Supporting the Agent Group Paradigm for Multi-Agent Systems," *International Conference on Intelligent Agent Technology*, 2007, 399-402.
18. You, Y. K., Han, S., Youn, H. Y., et al., "Context-based Dynamic Channel Management for Efficient Event Service in Pervasive Computing," *International Conference Computational Science and its Applications*, 2007, 24-32.
19. Verstichel, S., Strobbe, M., et al. "Distributed reasoning for context-aware services through design of an OWL meta-model," *International Conference on Autonomic and Autonomous Systems*, 2008, 70-75.
20. Wu, H., "Sensor Data Fusion for Context-Aware Computing Using Dempster-Shafer Theory," Ph. D. thesis, Carnegie Mellon University, 2003
21. Populaire, S., Blanc, J., et al. "Fusion of Expert Knowledge with Data using Belief Functions: a case study in wastewater treatment," *International Conference on Information Fusion*, 2002, 7-11.
22. Zacarias, M., Pinto, H. S., et al, "A context-aware and agent-centric perspective for the alignment between individuals and organizations," *Information Systems*, 35 (4), 2010, 441-466
23. Roy, N., Gu, T., and Das, S., "Supporting pervasive computing applications with active context fusion and semantic context delivery," *Pervasive and mobile computing*, 2010, 21-42.

24. Bikakis, A., and Antoniou, G., "Distributed Defeasible Contextual Reasoning in Ambient Computing," *Ambient Intelligence*, 5355, 2008, 308-325.
25. Lu, J., Li, Y., et al. "Distributed reasoning with fuzzy description logics," *International Conference on Computational Science*, 2007, 196-203.
26. Zhang, D., Guo, M., et al, "Context reasoning using extended evidence theory in pervasive computing environments," *Future Generation Computer Systems*, 26, 2010, 207-216.
27. Pavlin, G., Oude, P. D., et al, "A multi-agent systems approach to distributed Bayesian information fusion," *Information Fusion*, 11, 2010, 267-282.
28. Rizou, S., Haussermann, K., et al, "A System for Distributed Context Reasoning," *International Conference on Autonomic and Autonomous Systems*, 2010, 84-89.
29. Kanda, T., Yanagisawa, Y., et al, "A Distributed Inference System on Sensor Nodes using Neighbors' Context Data," *International Workshop on Databases for Next Generation Researchers*, 2007, 116-121.
30. Wei, Z., Wang, N., et al, "An agent-based context-aware middleware for pervasive computing," *International Symposium on Information Science and Engineering*, 2008, 116-119.
31. Hong, J., Suh, E., et al, "Context-aware system for proactive personalized service based on context history," *Expert Systems with Applications*, 36 (4), 2009, 7448-7457.
32. Luo, J., Dong, F., et al, "A context-aware personalized resource recommendation for pervasive learning," *Cluster Computing*, 13 (2), 2010, 213-239.
33. Wang, S. L., and Wu, C. Y., "Application of context-aware and personalized recommendation to implement an adaptive ubiquitous learning system," *Expert Systems with Applications*, 2011.
34. Object Management Group Agent technology green paper. Technical Report agent/00-09-01, 2000.
35. JADE (Java Agent DEvelopment framework) web resource available at <http://jade.cse.it>.
36. IBM Japan Research Group. Aglets (2007). <http://www.trl.ibm.com/aglets>
37. Han S. and Youn H. Y., "Modeling and Analysis of Time-Critical Context-Aware Service Using Extended Interval Timed Colored Petri Net," *IEEE Trans. on Systems, Man, and Cybernetics-Part A: Systems and Humans*, Vol. 99, 2011, 1-11.