International Journal of Informatics Society

08/09 Vol. 1 No. 2 ISSN 1883-4566

Informatics Society

Aims and Scope

The purpose of this journal is to provide an open forum to publish high quality research papers in the areas of informatics and related fields to promote the exchange of research ideas, experiences and results.

Informatics is the systematic study of Information and the application of research methods to study Information systems and services. It deals primarily with human aspects of information, such as its quality and value as a resource. Informatics also referred to as Information science, studies the structure, algorithms, behavior, and interactions of natural and artificial systems that store, process, access and communicate information. It also develops its own conceptual and theoretical foundations and utilizes foundations developed in other fields. The advent of computers, its ubiquity and ease to use has led to the study of informatics that has computational, cognitive and social aspects, including study of the social impact of information technologies.

The characteristic of informatics' context is amalgamation of technologies. For creating an informatics product, it is necessary to integrate many technologies, such as mathematics, linguistics, engineering and other emerging new fields.

Guest Editor's Message

Teruo Higashino
Guest Editor of the Second Issue of International Journal of Informatics Society

We are delighted to have the second and special issue of the International Journal of Informatics Society (IJIS) published. This issue includes selected papers from the Second International Workshop on Informatics (IWIN2008), which was held in Wien, Austria, Sep 9-11, 2008. This workshop was the second event for the Informatics Society, and was intended to bring together researchers and practitioners to share and exchange their experiences, discuss challenges and present original ideas in all aspects of informatics and computer networks. In the workshop, 21 papers were presented at 6 technical sessions. The workshop was complete in success. It highlighted the latest research results in the areas of networking, business systems, education systems, design methodology, groupware and social systems.

Each IWIM2008 paper was reviewed in terms of technical content and scientific rigor, novelty, originality and quality of presentation by at two reviewers. From those reviews, 12 papers are selected for publication of IJIS Journal. Among those 12 papers, six papers are related to computer networks. This second issue focuses on computer networks, and includes those selected six papers. The selected papers have been improved from their original IWIN papers based on the reviewers' comments. The rest of six papers will be published as the third issue of IJIS Journal.

We hope that the issue would be of interest to many researchers as well as engineers and practitioners in this area.

We publish the journal in print as well as in an electronic form over the Internet. This way, the paper will be available on a global basis.

Teruo Higashino is a professor at Osaka University, Japan. He received the B.S., M.S., and Ph.D. degrees in information and computer sciences from Osaka University in 1979, 1981 and 1984, respectively. He joined the faculty of Osaka University in 1984, and he has been a professor since 1999. From 2002, he leads Mobile Computing Laboratory, Graduate School of Information Science and Technology at Osaka University. His current research interests include mobile computing, wireless networks, communication protocols, ubiquitous systems, ITS (inter-vehicle communication), advanced sensing systems and so on. Dr. Higashino is Senior Member of IEEE, Fellow of Information Processing Society of Japan, and members of ACM and IEICE of Japan.

A proposal of feasible architecture for harmonizing IMS with MPLS-based traffic engineering

Takeshi Usui, Yoshinori Kitatsuji, Teruyuki Hasegawa, and Hidetoshi Yokota

KDDI Laboratories, Inc. 2-1-15 Ohara, Fujimino City, Saitama Prefecture 356-8502 JAPAN
ta-usui@kddilabs.jp

Abstract - IMS (IP multimedia subsystem) is a key technology for the next-generation network, to enable NSPs (network service providers) to provide various services over IP-based fixed and mobile networks. In order for the NSPs to provide stable network services, it is important to realize policy and QoS mechanisms in the transport network. In this paper, we propose feasible architecture of IMS harmonizing with MPLS (multiprotocol label switching) LSP (label switched path) selection. Our method uses IMS function to acquire the session profile for LSP selection. We further propose dual-phase capacity assignment, which achieves fair accommodation between the pairs of edge routers in our proposed architecture, and maximizes resource utilization.

Keywords: NGN, IMS, MPLS, Traffic Engineering

1 INTRODUCTION

Many fixed and mobile NSPs (network service providers) supporting PSTN (public switched telephone networks) services are now promoting convergence towards the NGN (next generation network) [1] architecture, in anticipation of cost-effective synergy between legacy and Internet services. NSPs will design and construct their IP-based NGN core network to provide various services on a single network infrastructure. These services also have various QoS requirements, for example, (a) VoIP (PSTN) traffic should be guaranteed, (b) some transaction or signaling/control traffic may be delay-sensitive, and (c) the Internet traffic can be best-effort.

Nowadays, NSPs are considering more traffic accommodation in the transport stratum to provide the network resources for various services. However, particular applications consume more bandwidth than before, and further, the traffic requirements of particular customers occupy most of the bandwidth in some NSP networks. It is therefore desirable to be able to accommodate as many customers as possible in a fair manner.

In the NGN architecture, IMS (IP multimedia subsystem) [2] is a key technology, where CSCF (call/session control function) [3] is responsible for call (i.e. communication session) control using SIP (session initiation protocol) [4]. NSPs can gain the QoS demand (e.g., bandwidth the delay) of each session before data transmission based on the SIP messages exchanged between UE (user equipment) and CSCF. Such a session demand is transferred to the policy control server (PCRF: policy and charging rules function) in order to determine whether the session can be accepted or not. However, IMS itself does not specify the transport stratum issues, (e.g., how to realize QoS in the core transport stratum). In addition, IMS does not assume any underlying mechanism with regard to the transport stratum.

On the other hand, many NSPs have introduced MPLS [5] in their transport networks to realize flexible traffic engineering, by setting up logical circuits (LSP [6]) between the pairs of edge routers reflecting various constraints and the operator’s policy. In addition, the NSP could collect the traffic amount per LSP directly related to the pair of edge routers. This information is convenient in that it enables a PCRF’s call admission control to realize more precise traffic engineering. From the viewpoint of the traffic control and management facilities in an NSP, it can be assumed that an MPLS is often adopted in their core networks. In this paper, we study the harmonization of an IMS with MPLS-based traffic engineering for the transport stratum. Our research goal is to provide a stable communication environment to customers and raising the traffic accommodation as well as maintaining fair resource utilization among the pairs of edge routers. We propose an efficient MPLS LSP configuration and extension of IMS function to achieve this goal.

This paper is organized as follows. We show several issues in QoS control in combining IMS and MPLS in Section 2 and design a traffic engineering policy in Section 3. We explain the details of the proposed architecture in Section 4 and evaluate the proposed capacity assignment method in Section 5. We show the conclusion in Section 6.

2 ISSUES OF QOS CONTROL IN NGN ARCHITECTURE

2.1 Session Control Procedure in IMS

The procedure for call/session establishment in a mobile packet-based network is standardized in 3GPP. SIP signaling originated by the UE is sent to CSCF,

ISSN1883-4566/09

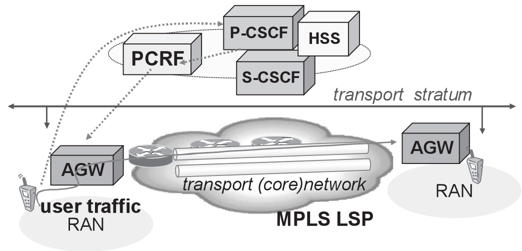


Figure 1 Functional structure of an NSP network using an IMS with MPLS

and CSCF responds to the UE as to whether the session can be accepted after obtaining a decision from PCRF.

Figure 1 abstracts a functional view of the NSP network using an IMS with MPLS. The transport stratum is composed of the RANs (radio access networks), AGWs (access gateways), and transport (core) network as shown in Figure 1. The AGW is located between the RAN and the core network and enforces QoS control for the user-data traffic (termed 'media traffic'). Gate opening/closing and marking the media traffic with the determined priority level. Packet marking is done by setting the bit value in the DSCP (diffserv code point) [7] or TOS (type of service) field in the IP header.

There are multiple signaling messages exchanged in establishing the session. The individual signaling messages go sequentially back and forth between UEs and CSCF. This implies that even if the one-way delay for a signaling message takes a few milliseconds, the completion of signaling takes several times longer than sending the single message. The delay requirement for the media traffic of SIP applications is less severe, comparing with the SIP signaling messages. Although the maximum domestic transmission delay (e.g., peaking at 10 milliseconds in Japan) may have little impact on the media traffic (application), but the round-trip time for exchanging the signaling messages is not small. Therefore, NSPs have to take these effects into account to minimize the signaling duration when designing and operating their networks. Based on these discussions, we presume that IMS-based services need, at least, the following traffic classes:

1. Class 1: both delay and loss sensitive, (e.g., signaling traffic)
2. Class 2: loss sensitive (e.g., VoIP traffic, and IPTV traffic)
3. Class 3: best-effort (e.g., Internet traffic)

2.2 MPLS Traffic Engineering

MPLS networks are composed of edge and core routers. Packets are transferred along one of the LSPs which are established between the ingress and egress edge routers. Once a packet enters the MPLS networks (the label for an LSP is assigned to the packet), core routers transfer the packet along the LSP. The mapped label value for an LSP expected by the egress router is delivered by RSVP (resource reservation protocol) [8] to an adjacent router along with the LSP. The adjacent router also delivers the mapped label for this LSP to its adjacent router towards the ingress router. Like this, the label delivery is conducted in a hop-by-hop manner.

MPLS traffic engineering provides benefits over IP network, that is to say, achieving the flexible control of the traffic in the transport stratum. The LSPs can either be routed explicitly (manually), or dynamically routed by the CSPF (constrained shortest path first) algorithm. In IP network, the shortest route to the destination is chosen by edge routers, even when it becomes more congested.

ABAF (automatic bandwidth adjustment function) has been specified and implemented [9] [10] [11] as one of a number of MPLS traffic engineering methods. This function not only automatically adjusts the LSP bandwidth but also dynamically reroutes the LSPs, when a certain physical link on the current LSP routes becomes short of capacity. The rerouting by the router is performed on an LSP basis; therefore, the ABAF may change the end-to-end delay of certain media traffic because of the sudden rerouting. In this situation the route is changed after the beginning of the communication, and the operators normally want to ascertain the route in the transport network. In ABAF, the route changes dynamically when no network failure occurs. Based on these issues, we assume that it is difficult for NSPs to adopt the ABAF in their MPLS networks.

From this discussion, it is clearly desirable that the edge routers have LSPs explicitly configured, have multiple LSPs for the traffic class, that the edge routers determine the route among them for arriving traffic and disperse the traffic over their networks, and that the total traffic in the network is taken into account for admission. The following function allows NSPs to meet these requirements: recognizing the demand for individual service traffic, selecting the LSP, and collecting the information on the utilization of the LSPs and the physical links.

When a pair of edge routers has multiple LSPs for a specific destination, most of the procedures for traffic dispersion are conducted by the ingress edge routers, as follows:

1. Rules to distinguish media traffic and determine one of the LSPs are stored in the ingress router beforehand.
2. The incoming traffic is distinguished into one of the traffic classes using the rules
3. The packet is marked in TOS or DSCP fields of IP header based on destination IP address and identified traffic class before the packet enters the ingress edge router.
4. The LSP is determined from the destination IP address and the mark of the packet, and the

MPLS forwarding table is looked up to find the label of the LSP.

Step 1 is realized using static rules or interworking with another entity to dynamically update the rules. The procedure for IMS enables the rules to be dynamically updated on a session basis as described in the next subsection.

2.3 Harmonizing IMS with MPLS

In this paper, we consider increasing the utilization of the transport (core) network by having IMS and MPLS cooperate. IMS provides session demands on UEs before the beginning of their communications. MPLS is used as the transport stratum and allows media traffic to be dispersion over LSPs and to provide fast reroute function [12]. IMS provides session demands on UEs before the beginning of their communications. Such information is useful to determine the target LSP for the communications; however, the following items should be considered for harmonizing IMS with MPLS-based traffic engineering:

1. deploying cooperative session control procedures between CSCF, PCRF, and AGWs,
2. recognizing the resource utilization of the transport stratum, and
3. an admission control method to deal with multiple traffic classes,

For item 1, although the QoS/policy control architecture is being standardized in 3GPP/3GPP2 [13], QoS/policy control in the core network is largely left for the deployment. In harmonization IMS with MSLS we, consider, LSP-based traffic statistics are collected, although physical link-based traffic statistics are after collected in the generic network operation. For item 2, the LSP traffic statistics concerning resource utilization are useful. For item 3, the method should take into account the fair accommodation described in Section 3.2 for the accepted amount of traffic at the ingress edges.

2.4 Related Work

ITU-T [1] standardizes the RACF (resource and admission control function) [14] as the QoS and admission control function in the NGN. The RACF has the same role as PCRF in 3GPP/3GPP2. However, the issue of how to adapt the RACF function to the control for the transport stratum also remains unresolved. We propose a function to control the transport stratum using MPLS.

Tamura et al. numerically examined that the optimal threshold for commencing traffic distribution over two LSPs and the optimal distribution over the two LSPs that will maximize the admitted traffic among n pairs of edge routers in reference [15]. This study presumed that each pair of edge routers would initially use a single LSP between them, and then begin to use a secondary LSP when the traffic exceeds the threshold. This study does not consider multiple traffic class. We presume that high priority traffic is always transferred into the shortest route even if the threshold is exceeded. We assume that the traffic demand can be recognized by tracking the traffic trend in every instance of the fixed time interval.

There is study to propose methods effectively minimizing the delay of signaling messages exchanged in IMS in order to ensure high communication quality [16] [17].In addition, various signaling methods in IMS have been investigated in 3GPP and academic research for this objective. However, there has been little study concerning the effective treatment of signaling in terms of traffic engineering, (e.g., the simultaneous treatment of signaling and media traffic) in IMS. Since the signaling includes various procedures during the session, a low-loss and minimum delay transport network is essential to ensure good communication quality that is perceived by users. Therefore, NSPs must take care when transferring signaling packets in the transport network.

3 DESIGN POLICY FOR TRAFFIC ENGINEERING

3.1 Traffic Class

We assume three traffic classes in our proposal, as described in Table 1. The primary class is for signaling which requires the minimum delay, while the standard class is for media traffic requiring sufficient bandwidth. We adopt IMS application traffic as standard-class traffic. Additionally, the best-effort traffic (e.g., Internet access) without a QoS requirement is taken into account.

Additional traffic classes for a more fine-grained treatment of traffic levels may be defined in certain NSPs. For example, the traffic for streaming applications requires a lower delay variation. We presume that such granular classes are treatable by a weighted round robin-based queuing discipline combined with priority queuing.

Table 1.Traffic Class

PRIORITY LEVEL	TRAFFIC CLASS	TRAFFIC TREATMENT
High priority	Primary class	Delay restriction needed - the shortest or sufficiently small delay routes
	Standard class	Loss sensitive - traffic distribution over multiple routes to gain capacity
Low priority	Best effort	Transferred into the shortest route if there is capacity.

We adopted at least three traffic classes in this paper to realize the traffic treatment in table 1. In this paper, a minimum three traffic classes was used in order to validate the effectiveness of the proposed architecture for harmonizing IMS with MPLS.

We assume that both primary and standard classes have a threshold, up to which traffic can be aggregated, while the remaining capacity can be allocated for the best-effort class. This threshold (termed *acceptable capacity*) is defined for each physical link. For the bandwidth requirement, we assume that the demand for the primary class is much less than the acceptable capacity for all physical links. However, the demand for the standard class exceeds the acceptable capacity. By using multiple LSPs, the standard class traffic can be transferred through these LSPs. The signaling traffic has a strong requirement for the minimum delay, so the primary-class traffic should take the shortest route among the LSPs between the pairs of edge routers. For the standard class, we assume the capacity requirement to be stronger than that of the delay. Standard-class traffic can be distributed over the multiple LSPs.

3.2 Fairness-aware capacity assignment policy

We consider that the bandwidth of any customer traffic is guaranteed at a certain minimum level. In this paper, fair accommodation means that a certain level of capacity is guaranteed on any pair of edge routers. This allows customers accommodated in any edge routers to have the successful rate of call/session establishment at some level even if the total demands in the networks severely exceed their capacity.

To increase the standard class traffic that can be admitted, while providing fair accommodation, we focus on two traffic assignments: one is for the lowest capacity among all capacity assignments for all pairs of edge routers, and the other is for the total capacity in the transport network. In order to satisfy the above principle, we propose dual-phase capacity assignment. In this capacity assignment, the first phase involves maximization of the lowest capacity assignment. During the second phase, the total capacity assignment is maximized from the remaining capacity of the physical links. For the first phase, an identical capacity is assigned to all the pairs of edge routers, and maximized, while for the second phase, we adopt the strategy of maximizing the total amount of the traffic admitted for the remaining capacity. The capacity of the physical link is assigned to the LSPs in order, starting with the smallest number of links composing each LSP. The way to maximize the total assigned capacity is as follows. When there is a certain capacity assignment $\boldsymbol{A}$, and assignment $\boldsymbol{A}$ has an LSP that can be composed of two or more distinct and shorter LSPs, another capacity assignment, whereby the former LSP is replaced by the latter LSPs has a greater total capacity than assignment $\boldsymbol{A}$ in terms of demand and based on the pair of edge routers. Details of the method for computing the capacity assignment that captures dual-phase assignment are described in Section 5.

4 PROPOSED ARCHITECTURE

4.1 MPLS LSP Configuration

Figure 2 shows the proposed network architecture. The AGW are connected to MPLS edge routers and filters the traffic from UEs. In typical MPLS network operation, single LSP with the shortest routes is established for every pair of (ingress and egress) edge routers. To best accommodate the traffic, traffic engineering is applied if another LSPs with detour routes are available. For example, in Figure 2, three LSPs along the shortest route (shortest LSPs) and an additional LSP along the detour route (detour LSP) are established between edge routers X and Y. Each shortest LSP (LSP-1, LPS-2 and LSP-4) is assigned to an individual traffic class (the primary-class, the secondary-class and the best-effort class, respectively). One of the shortest LSPs (LSP-2) and the detour LSP (LSP-3) are assigned to the standard traffic class. The standard class traffic is distributed over the two LSPs.

LSP-1 is for the primary-class traffic, while LSP-2 and LSP-3 are for standard-class traffic and LSP-4 is for best-effort traffic. LSP-2 preferably is separated from the standard-class LSP-2. The primary-class LSP, standard-class LSP-1, and the best-effort class LSP are established along with the shortest route between edge routers X and Y.

The traffic utilization for each LSP can be collected with SNMP [18]. Gathering statistics on the traffic of multiple LSPs (LSP1, 2, 3, 4) between any pair of edge routers, we can get the total traffic which is transferred among the edge routers. If we do not set up the LSP, we cannot acquire such traffic utilization easily by SNMP. It allows PCRF to acquire the traffic utilization for each LSP, and PCRF can use it for admission control.

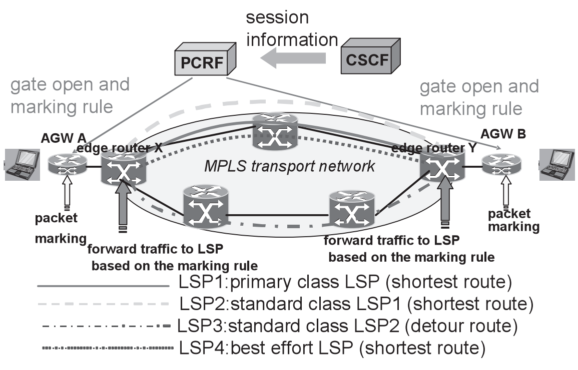


Figure 2 overview of network architecture

In our proposed architecture, we propose setting up multiple LSPs among each edge router. Our LSP configuration does not impose an additional load on the routers because the routers need not have functions like ABAF.

4.2 Admission Control Procedure

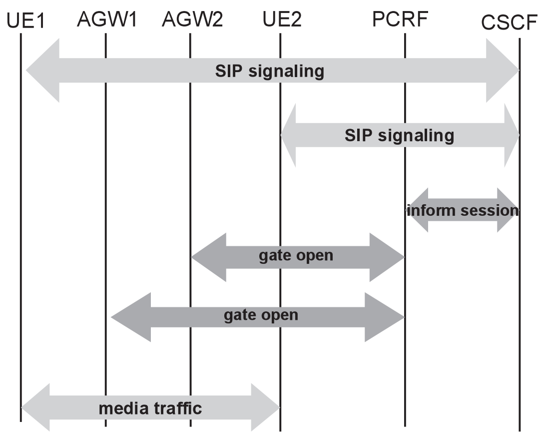


Figure 3 Session initiation procedure in IMS

Figure 3 illustrates the session initiation procedure in IMS standards. The AGWs set filters for QoS and policy control of the UEs' traffic. CSCF reports the session demands of the UEs to PCRF. Then PCRF requests the AGWs to open the gate for the UEs' media traffic, whereupon they begin the communication through the AGWs. In our proposal, the *gate open* procedure in the AGW is expanded to perform marking packets for traffic engineering. Based on the value of marking from PCRF, AGW performs marking packets for the media traffic. The detailed signaling procedures, e.g., the bidirectional media-traffic treatment, between UEs, AGWs, PCRF, and CFCS, are described in the following subsection.

Harmonization IMS with MPLS means that PCRF determines the admission and LSP selection for the media traffic with its demand which is provided by AGW and MPLS edge routers can acquire the rules used to distinguish the packets into traffic classes and marking before the media traffic arrives at the AGWs. We use the basic MPLS function, and need not expand the function of the MPLS edge router.

For media traffic session, the behavior of AGW is as follows:

1. The media traffic is marked in AGW to transfer it through the shortest LSP if the traffic is identified as primary class.
2. If the media traffic is identified as standard class, the traffic is distributed among multiple LSPs in MPLS edge router. Incoming media traffic is rejected if existing media traffic in each LSP has reached the acceptable capacity for one of the links composing the LSP.
3. Media traffic identified as the best-effort class is transferred into best-effort class LSP.

To prevent the loss of packet for the traffic going through of the primary and standard classes, admission control for the next call request of customers is important. The utilization of the physical links is regularly monitored and referred to decide whether media traffic is accepted, depending on LSP to which the media traffic is transferred. The monitoring is conducted by collecting the traffic counters for all LSPs, and the utilization of all the physical links is computed. We propose that PCRF performs the utility computation for admission control. PCRF can acquire the demand for arriving media traffic from CSCF.

We extend the standard session initiation procedure of IMS for the proposed traffic engineering, which is as follows:

1. The UE initiates the procedure with CSCF to establish SIP session.
2. CSCF queries PCRF to determine the LSP in the transport stratum through which the media traffic of the caller and the callee is transferred. Here, various parameters are informed to PCRF, (e.g., application type, IP addresses and port numbers).
3. PCRF distinguishes the media traffic into one of the traffic classes and determines whether the media traffic can be accepted. Here, PCRF refers to the utilization of the physical links along the LSPs assigned to the media traffic.
4. PCRF responds to CSCF if the LSP is determined.
5. CSCF report the establishment of the bearer to UEs.
6. PCRF sets up AGWs of the caller and the callee to open the gate for the media traffic and mark the media traffic.

PCRF and AGW routers have a common definition of the mapping between the mark (DSCP or TOS bit values) and the corresponding LSP.

In the transport stratum, the traffic controlled by IMS signaling can be mixed with the Internet traffic. We assume that non-IMS-based traffic is grouped into the best-effort class. So in this procedure, packets with the default mark (or no mark) are assigned to the best-effort class.

4.3 LSP selection procedure

In our proposal, we presume that whenever the call of standard-class traffic arrives, PCRF determines which standard class LSP the AGW should transfer the media traffic and whether it is accepted or not. PCRF calculates the capacity assignment for media traffic acceptance beforehand. The utilization of the physical links and LSPs which PCRF recognizes, is updated at specific time intervals (e.g., every 1 or 3 minute/s), since the traffic counter values for LSPs are collected at this interval. The capacity assignment for the media traffic acceptance for all the LSPs is also updated at this interval. The LSP selection for the standard traffic class is also conducted in proportion to the assigned capacity. On the other hand, PCRF responds with *cancel* to CSCF if any of the media traffic (e.g., a VoIP service using bidirectional traffic) is rejected.

5 CAPACITY ASSIGNMENT

5.1 Modeling of dual-phase capacity assignment

We applied the LP (linear programming) approach to achieve the goal of our research. We presumed that the demand was a real number to simplify the LP computation. We considered a model for dual-phase capacity assignment for standard-class traffic: the first assignment maximizes the minimum capacity among all the pairs of edge routers, with achieving fair accommodation for all the standard classes at minimum level; and the second assignment maximizes the total capacity for the remaining capacity in the transport stratum.

Maximizing the minimum capacity is computed by solving the LP, which maximizes the identical capacity assigned to all the pairs of edge routers. We define the following objective function for the first phase assignment:

$$C = d_k = \sum_i d_{k,i}$$

where d_k, $d_{k,i}$ denote the assigned total capacity of the edge router pair *k*, and the capacity of LSP *i* for edge router pair *k*. Here, we define the following constraint conditions for the above objective function:

$$\sum_k \sum_i x_{e,k,i} - u_e \leq 0 \text{ for } e \in E$$

where $x_{e,k,i}$ and u_e denote the assigned capacity of LSP *i* for the edge router pair *k* in the physical link *e*, and the available capacity for the standard class traffic in the physical link *e*. Although identical capacities are assigned to each pair of edge routers, this capacity can be distributed via multiple LSPs of each pair of edge routers.

Similarly, the second phase assignment is also computed by solving the LP, which maximizes the total capacity $C = \sum_k \sum_i f_{k,i}$ under the constraint conditions $\sum_k \sum_i y_{e,k,i} + \sum_k \sum_i x_{e,k,i} - u_e \leq 0$ for $e \in E$, where $f_{k,i}$ and $y_{e,k,i}$ denote the additional assigned capacity of LSP *i* for the edge router pair *k*, and that in the physical link *e*.

5.2 Evaluation Method

To evaluate the effect of the proposed capacity assignment in PCRF, we performed the simulation and compared the capacity assignment in all pairs of edge routers between four traffic assignments by varying the number of routers in the MPLS network (edge and transit routers). The first is the dual-phase capacity assignment with single LSP for the standard class traffic in each pair of edge routers. This is termed "*1-path max-min*". The second is the dual-phase capacity assignment with two LSPs in each pair of edge routers (termed "*2-path max-min*"). The third is the dual-phase capacity assignment with three LSPs in each pair of edge routers (termed "*3-path max-min*"). For the fourth, only the second phase of the dual-phase capacity assignment is applied without any fair accommodation consideration. It has the two LSPs for each pair of edge routers, and is termed "*2-path shortest-first* ".

To emulate the NSP topology, we used BRITE (Boston University Representative Internet Topology) [19]. BRITE is a tool for emulating network topology in an AS (autonomous system). BRITE provides the BA (Barabasi-Albert) model [20], which is often used to emulate the topology. We specified the number of routers and the degree, (the number of physical links per individual router) and generated network topologies for the simulation.

We specified that all links in the generated network topology had identical link capacities. In addition, the shortest LSP and detour LSPs (when multiple LSPs were used) were computed for all the pairs of edge routers. We need to be careful of the crossover of some detour LSPs, when setting up a detour LSP in the network topology.

In generating the network topology, we defined 80% of the routers as edge routers, and 20% as core (transit) routers. This reflects the situation in the NSP's MPLS networks, where there are many edge routers at the head and tail ends of the LSPs, and a smaller number of core routers, which transit traffic for the edge routers by switching the LSP. In the simulation, the edge routers had a degree of at least "2" since the edge router normally has two physical interfaces to connect the core (upper) network in the NSP. The core routers had a degree of over three.

The four capacity assignments were compared in the same network topology provided by BRITE, while BRITE also varies the topology as it generates. We tried ten simulations for each number of routers, and the results were averaged. To solve the modeled formula of LP, we used GLPK (GNU Linear Programming Kit).

5.3 Evaluation Result

The results of the simulations are shown in Figures 4 to 7. Figures 4 and 5 show the average and lowest amounts of traffic admitted at all the edge routers. The X-axis is the number of edge routers, while the Y-axis is the average or lowest volume of standard traffic class for each of edge routers and shows the ratio of traffic admitted to link capacity in edge routers. The value is normalized by the link capacity.

The average ratio of traffic admitted to link capacity (Figure 4) shows that 2-path shortest-first assignment achieves the largest traffic accommodation. Compared with 2-path max-min assignment and 2-path shortest-first assignment, the average amount of traffic admitted for 2-path max-min assignment is more than 80% of that for 2-path shortest-first assignment. But the object of 2-path max-min assignment is fair accommodation. Comparing 1-, 2- and 3-path max-min assignments reveals that traffic accommodation becomes increasingly similar, as number of edge routers increases.

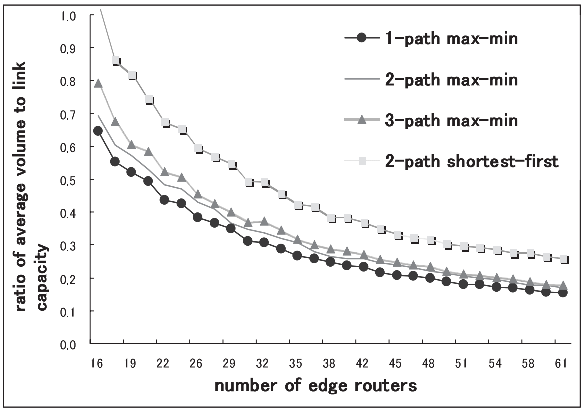


Figure 4 Average amount of traffic admitted in edge routers.

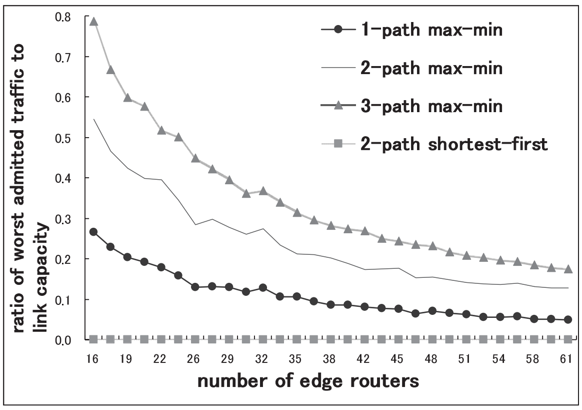


Figure 5 lowest amount of traffic admitted for each number of edge routers.

In the case of the lowest accommodation (Figure 5), 3-path max-min assignment achieves the largest traffic accommodation. 2-path max-min assignment admits about twice times as much traffic as 1-path max-min does. For 3-path max-min assignment, the admitted traffic is 1.2-1.3 times larger than with 2-path max-min assignment.

From the simulation data in Figure 5, 2-path shortest-first assignment generates a lot of "0" capacity assignment. 2-path-shorest-first assignment can achieve maximizing the capacity assignment in the network, but cannot assign minimum capacity assignment at some pairs of edge routers.

Figure 6 and 7 show the cumulative distribution functions for the ratio of traffic admitted to link capacity for 20 and 40 edge routers. These figures show that a relatively large number of edge routers cannot admit the demand for the 2-path shortest-first assignment. The results of the 2-path shortest-first assignment indicate that some of the edge routers cannot admit any traffic demand. For the other assignments, the fair accommodation is achieved.

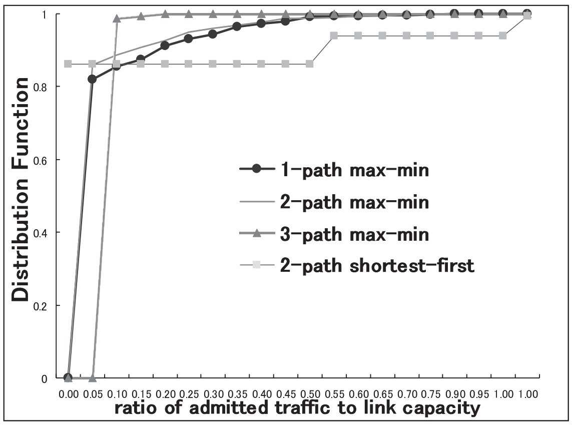


Figure 6 Cumulative distribution function in 20 edge routers

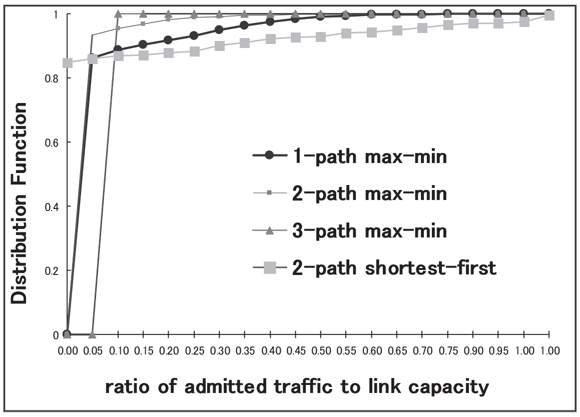


Figure 7 Cumulative distribution function in 40 edge routers

In terms of fair accommodation, it is too constraining to take the 2-path shortest-first assignment in the NSP's operation and service. These results indicate that the 2-path shortest-first assignment is impractical for an NSP's operation, while 2- and 3-path max-min assignments are effective in terms of fair accommodation.

Focusing on the number of LSPs in each pair of edge routers, the additional benefit of 3-path max-min assignment from 2-path max-min assignment is less than that of the change to 2-path max-min assignment from 1-path max-min assignment. And, comparing 3-path max-min with 2-path max-min, the benefit becomes more worthwhile as the number of edge routers increases. This is probably because the number of disjoint LSPs for the 3-path case is similar to that for the 2-path case. Moreover, this limitation may result from the number of degrees (two for edge routers and three for transit routers) in the simulation network topology. Generally, it is more difficult to set up multiple disjoint paths as the number of LSPs in each pair of edge routers increases. We presume that the progress of 4-path max-min assignment or cases using more paths is probably much less than that of 3-path max-min assignment from 2-path max-min assignment.

The differences of the admitted traffic ratio for each number of routers between the 1-,2- and 3-path max-min assignments in Figure 5 are is more than that in Figure 4. We can say from the results that the lowest value of capacity assignment is remarkably improved by setting more multiple LSPs between the pairs of edge routers. It achieves fair accommodation between each pair of edge routers more effectively

The number of variables in LP to solve the capacity assignment increases as the number of routers increases in the simulated topology. However, even with over 60 edge routers in our simulation, the computation time required to solve the modeled LP was generally less than one second. We used an off-the-shelf PC with a 2.00 GHz Intel Core™ 2 CPU and 1.99 GB memory for the simulation. Therefore, the computational load of our proposed capacity assignment is sufficiently low.

5.4 Discussion and Future Work

We set "two" as the value of the lowest degree in BRITE for the simulation. It is because edge routers usually have two physical interfaces for redundant access to core routers, taking account into the realistic network topology of NSPs. We presume that using "two" as the lowest degree is sufficient to simulate the realistic network topology. Using "one" as the lowest degree, we presume that the effect of 1-, 2- , 3-path max-min is not large. And, using "three" or larger value as the lowest degree, we presume that the effect has more progress.

However, increasing the number of LSPs for each pair of edge routers complicates network operation, in terms of LSP maintenance, e.g., recovering from failure and utility monitoring. The number of LSPs set up between pairs of edge routers entails a trade-off between operational cost and resource utilization. Issues for future consideration include performing validity of multiple LPSs at each pair of edge routers and increasing the number of degrees in the network topology. And, it is necessary to evaluate the performance of our proposed capacity assignment for a number call requests.

6 CONCLUSIONS

In this paper, we proposed the architecture for harmonizing IMS with MPLS-based traffic engineering. We showed the extended PCRF function as a way to utilize IMS session demands for resource management of the MPLS core network. IMS provides session demands on the media traffic and MPLS disperses media traffic following PCRF's determination of path assignment.

We presented the benefits of harmonizing IMS with MPLS and proposed the required functions, architecture, and procedure. To implement our proposal, the MPLS routers need not have additional functions. A variety of methods might achieve the objective of our work, however in this paper we proposed first of all that the signaling traffic is prioritized as primary-class LSP and that the media traffic is transferred into multiple standard-class LSPs.

With regard to traffic engineering in the proposed architecture for the standard-class traffic, we proposed a dual-phase capacity assignment to maximize the lowest value in the capacity assignment, and to maximize the remaining bandwidth for the standard-class traffic. In the evaluation of capacity assignment, we compare the effect of our proposed capacity assignments with the 2-path shortest-first, which is equivalent to the second phase of the proposed capacity assignment. We thus showed that the lowest amount of traffic admitted into the edge routers increased, corresponding to the number of LSPs per router pair. However, the average amount of traffic admitted into the edge routers when adopting one, two and three LSPs was almost identical, regardless of the number of edge routers.

REFERENCES

[1] ITU-T NGN release1 Y.2001 "General overview of NGN"

[2] ETSI "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN)", ETSI E2 282 001 V1.1.1 NGN Functional

Architecture Release 1.

[3] ETSI, “Telecommunications and Internet Converged Services and Protocols for Advanced Networking (TISPAN); IP Multimedia Call Control Protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP) Stage 3”, ETSI ES 283 003, V1.1.1, April 2006.

[4] M. Handley, H. Schulzrinne, E. Schooler and J. Rosenberg, "SIP: Session Initiation Protocol", IETF RFC 3261, June 2002

[5] E. Rosen, A. Viswanathan, R. Callon, ”Multiprotocol Label Switching Architecture”, IETF RFC3031, January 2001

[6] D. Awduche, J. Malcolm and J. Agogbua, “Requirements for Traffic Engineering Over MPLS”, IETF RFC2702, September 1999

[7] K.Nicols, S.Blake, F.Baker, ” Definition of the Differentiated Services Field (DS Field)in the IPv4 and IPv6 Headers”, IETF RFC2474, December 1998

[8] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", IETF RFC 3209, December 2001.

[9] Aumatic Bandwidth Adjustment for TE tunnels, ”QoS for IP/MPLS networks” Cisco Press ISBN 1587052334

[10] H. Ishi, K. Nagami, ”Proposal and evaluation for Automatic bandwidth setting in MPLS LSP” IEICE B Vol. J89-B No. 10 pp.1894-1901.

[11] F.Le Faucheur, ”Aggregation of RSVP(Resource ReSerVation Protocol) Reservations over MPLS TE/DS-TE Tunnels” IETF RFC4804, Feburary 2007.

[12] P. Pan, G. Swallow, A. Atlas, ”Fast Reroute Extensions to RSVP-TE for LSP Tunnels”, IETF RFC4090, May 2005

[13] 3GPP2 X.S0013, “All-IP Core Network Multimedia Domain”.

[14] ITU-T NGN release1 Y.2111

[15] H. Tamura, K. Kawahara, Y. Oie, ”Analysis of two-phase Path Management Scheme for MPLS Traffic Engineering” Proc. Of SPIE Performance Quality of Service, and Control of Next Generation Communication Networks2 pp. 194-203, Oct. 2004.

[16] S. Zaghloul, A. Jukan, W. Alanqar, "Extending QoS from Radio Access to an All-IP Core in 3G Networks: An Operator's Perspective", IEEE Communications Magazine.

[17] F. Wegscheider, "Minimizing unnecessary notification traffic in IMS presence system", 1st International Symposium on Wireless Pervasive Computing, 2006.

[18] C. Srinvasan, Bloomberg L.P.,"Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Management Information Base (MIB)", IETF RFC 3812, June 2004.

[19] A. Medina and I. Matta, “BRITE:A Flexible Generator of Internet Topologies”, Tech. Rep. BU-CS-TR-2000-005, Boston University, Boston MA, 2000.

[20] R. Albert and A. Barabasi, “Statistical mechanics of complex networks”, Review of Modern Physics, 2002.

(Received October 17, 2008)
(Revised July 15, 2009)

Takeshi Usui received his B.E. and M.E. from Keio University in 2002 and 2004, respectively. He began working for KDDI Co., Ltd. in 2004 and managed KDDI IP-VPN service and designed the backbone network of KDDI. In 2007, he began working for KDDI R&D Laboratories, Inc. His current research interests include mobile communications and next-generation networks. He is a member of the IEICE.

Yoshinori Kitatsuji received his B.E. and M.E degrees from Osaka University, Japan in 1995 and 1997, respectively, and received his D.E. degree from Kyushu Institute of Technology, Japan in 2007. In 1997, he joined the KDD Co., Ltd. He was a research fellow of Telecommunications Advancement Organization of Japan from 2001 to 2003, and was an expert researcher in Kyushu Research Center, National Institute of Information and Communications Technology from 2004 to 2006. He is currently a research engineer of Mobile Network Laboratory in KDDI R&D Laboratories, Inc. His research interests include performance monitoring and traffic engineering of computer communication networks, and architecture of mobile communication. He is a member of the IEICE.

Teruyuki Hasegawa received the B.E. and M.E. Degrees of electrical engineering from Kyoto University in 1991 and 1993 respectively, and Ph.D degree of information science and technology from Tokyo University in 2008. Since joining KDD (now KDDI) in 1993, he has been working in the field of high speed communication protocol and multicast system. He is currently the senior manager of Future Communication Architecture Lab. in KDDI R\&D Laboratories Inc. He received The Meritorious Award on Radio of ARIB in 2003. He is a member of IEICE and IPSJ.

Hidetoshi Yokota received his B.E., M.E. and Ph.D. degrees from Waseda University, Tokyo, in 1990, 1992, and 2003, respectively. He began working for KDDI R&D Laboratories, Japan, in 1992. From 1995 to 1996 he was with SRI International, in Menlo Park, CA as an International Fellow. He received the IEICE Young Engineer Award, the IPSJ Yamashita SIG Research Award and the IPSJ Best Paper Award in 1998, 2005 and 2006, respectively. His current research interests include mobile communications and next-generation networks. He is also involved with several standardization activities including IETF and 3GPP.

Determining the Relay Node Encode Packet in Multipath Routing Environment

Tomonori Kagi *and Osamu Takahashi**

*Graduate School of Systems Information Science, Future University Hakodate, Japan
**Faculty of Systems Information Science, Future University Hakodate, Japan
{g2107006, osamu}@fun.ac.jp

Abstract **-** Wireless networks, such as mobile ad hoc ones have low reliability for reasons such as phasing, noise, and packet collisions. FEC-based methods in ad hoc networks have been improved because of their increased reliability when used with multipath routing. However, the number of transmission packets from the source node has been increased. Therefore, we propose using an efficient and reliable packet transmission method: using multipath routing constructs from multiple node disjoint routes and applying network coding, which allows packet encoding at a relay node. Because the encoding packet is generated by a relay node, the source node does not need to encode the packets, and it sends only unencoded packets to each route. Thus, the number of packets transmitted by the source node does not increased. In addition, we also evaluated which node was most suitable to encode a packet, the location of the path that should be used to encode it and the delivery ratio by the number of packets used for encoding.

Keywords: Wireless network, MANET, Network Coding, Multipath Routing

1 INTRODUCTION

Recently, progress in wireless communication technology has meant that wireless modules have been mounted on various devices. These ad hoc networks are instantly deployable wireless networks, which rely on radio waves instead of base stations or communication infrastructure support. Because radio waves have a short propagation range, the route becomes “multihop” when a communication peer is not within range. In general, the reliability is low in ad hoc networks because of network topology, unstable radio environment, and packet collisions. By “reliability” we mean the probability that data generated at a source node in the network can be routed to the intended destination.

Packet-level forward error control (FEC) and automatic repeat request (ARQ) are two methods widely used to recover the lost packets in networks with unreliable links.

Automatic repeat request is an error recovery method that uses acknowledgment packets (ACKs) and a timer to transmit data reliably. The acknowledgment packet is a message sent by the receiver to the sender to indicate that it has correctly received a packet. If the sender does not receive an acknowledgment before a specified period of time (timeout), the sender usually retransmits the packet until it receives an acknowledgment or exceeds a predefined number of retransmissions. However, the ARQ method is not considered applicable in networks that have low reliability and that are highly mobile, such as ad hoc ones. This is because the transmission delay increases as a result of retransmissions by the sender for missing ACKs. In addition, because of its use of unidirectional links, ARQ is unfit for wireless networks [5].

Forward Error Control is an error correction method that is used in data transmission in which the sender generates an error correction code, adds it to the original packet, and then sends both the error correction code and the original packet. Using this method allows the receiver to detect and correct errors without the need to ask the sender for retransmission of the packet.

The use of FEC-based methods in ad hoc networks has been studied [6] [7] [8] and found to improve the reliability when used with multipath routing. However, the number of transmission packets of the source node is increased. For example, as shown in Figure 1, the source node S generates a code from Data 1 and Data 2 by encoding them. The source node then sends the code. In this case, the number of packets transmitted by the source node is three (Data 1, Data 2, and Code). Thus, the transmission frequency at the source node is increased.

We proposed using a method that involves Network Coding [2] that allows packet encoding at a relay node to decrease the number of packet transmitted by a source node and the number of packets that flow into the network accompanying it. In addition, we carried out a computing simulation to evaluate it [1].

The results from the simulation show that our proposed method transmits data more efficiently and more reliably than the current method does. However, when multiple paths are constructed from multiple relay nodes, how to decide on which node should be encoded has yet not been discussed. Thus, we theoretically evaluated which relay nodes should encoded and also on which paths packet from the delivery ratio and packet overhead, and the delay. In addition, to validate our theoretical evaluation, we conducted simulations.

Our method is discussed in Section 2. A prototype implementation of our proposal is described in Section 3. We evaluated our proposal by comparing related protocols in Section 4, and we summarize our work in Section 5.

2 PROPOSED METHOD

In this section, we describe our proposed method using Network Coding with Multipath Routing.

2.1 Basic Operation Model

The construction of the multiple route method is Split Multipath Routing [4], which is also known as extended Dynamic Source Routing [3].

Multiple paths are constructed, and then the source node sends data packets to neighbor nodes on all routes simultaneously. For example, as shown in Figure 1, when two paths are constructed a data packet is forwarded on one path, and a certain relay node on another path encodes a packet and forwards it.

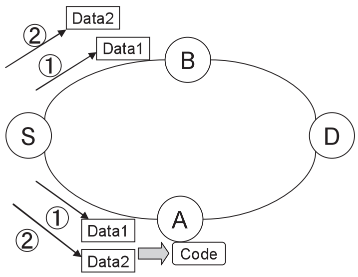


Figure 1 Proposed method

Therefore, the destination node is able to receive encoding packet even if there is no source node that encodes the packet and then transmits the encoding packet.

3 CHOICE OF NODE

Our proposal method is modeled by using a mathematical expression.

In addition, we evaluated whether a data packet should be forwarded and the position of the path where the encoding packet should be generated and forwarded. These conditions were on the basis of a theoretical formula.

Similarly, we evaluated which relay node should encode a packet, when the path is constructed by two or more relay nodes.

In order to estimate the efficiency of our proposal, we evaluated the packet delivery ratio, the packet overhead, and the transmission delay. We define these parameters as follows.

- Packet delivery ratio

The packet delivery ratio is defined as the number of correctly received data packets at the destination node divided by the number of original data packets sent by the source node.

- Packet overhead

The packet overhead is defined as the number of all node transmission packets, including data packets and encoded packets.

- Transmission delay

The transmission delay is defined as the period from which the source node generates a packet until the time when the destination node receives it. For encoding models, the transmission delay is defined as the period from when the source node generates a packet until the time the destination node decodes the encoded packets and retrieves the original ones.

3.1 Evaluation Model

The evaluation model is shown in Figure 2

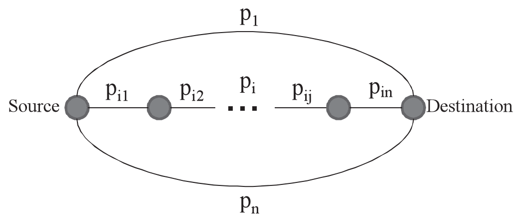


Figure 2 Evaluation model

Packet Loss Rate between nodes in the *i*th path is given by:

$$\left(p_{i1} \quad p_{i1} \quad \ldots \quad p_{ij} \quad \cdots \quad p_{in}\right) \qquad (1)$$

Packet Loss Rate between the source node and the destination node in each path is given by:

$$\left(p_1 \quad p_2 \quad \ldots \quad p_i \quad \cdots \quad p_m\right) \qquad (2)$$

The number of hops between the source node and the destination node in each path is given by:

$$\left(H_1 \quad H_2 \quad \ldots \quad H_i \quad \cdots \quad H_l\right) \qquad (3)$$

The packet loss rate in each path, which is computed from the packet loss rate between nodes, is given by:

$$p_i = 1 - \prod_{j=1}^{Hi} (1 - p_{ij}) \qquad (4)$$

3.2 Coding Scheme

When we constructed the redundancy packets, we assumed the that the encoding we used would have parameters (N, K, and t), where (i) N is the number of transmitted packets in a group; (ii) K is the number of the data packets in this group; and (iii) t is the erasure recovery capability, i.e., the maximum number of lost packets within the group that can be reconstructed on the basis of the received packets.

3.3 Precondition

In order to simplify the evaluation, the following items are defined as a precondition.

・Number of routes: 2
・Transmission path for data packets: path1
・Transmission path for encoding packets: path2
・Encoding parameters: (N, K, 1)

3.4 Packet Delivery Ratio

The relay node, which encodes is determined for the better packet delivery ratio. The following parameters are defined.

One of the following two requirements needs to be filled in order to send a packet to destination node correctly.

(Requirement 1) Destination is received data packet.

(Requirement 2) Destination node is received packet required decoding.

The probability of meeting requirement 1 is given by:

$$P_1 = 1 - p_1 \quad (5)$$

We then calculated the probability of meeting requirement 2.

The encoding packet is generated and delivered to a destination node.

The probability that the source node can transmit one packet to the relay node that is an encoding packet is given by:

$$P_{e1} = \prod_{i=1}^{H_e} (1 - p_i) \quad (6)$$

The variable He is the number of hop to a relay node, which encodes the packet.

The probability that the destination node receives two, three, ..., K packets required for coding is approximated by:

$$P_{e2} = \sum_{n=0}^{X} \prod_{j=1}^{H_e} (1 - (1 - p_{2j}))^n (1 - p_{2j}) \approx 1$$

(7)

The probability that the source node can transmit a packet required for encoding to the relay node, which is encoding packet is given by:

$$P_e = P_{e1} P_{e2} = \prod_{i=1}^{H_e} (1 - p_i) \quad (8)$$

The relay node encodes packets if the packet required for encoding is received.

The probability that the relay node can encode and forward the packet to the destination node is given by:

$$P_{ed} = \prod_{j=H_e+1}^{H_2} (1 - p_{2j}) \quad (9)$$

From (8) and (9), the probability that the encoding packet is generated by a source node is forwarded to the destination node is given by:

$$P_{sd} = P_e P_{ed} = 1 - p_2 \quad (10)$$

The probability data packet is decoded from an encoding packet and another data packet required for decoding without receiving a data packet is given by:

$$P_2 = p_1 (1 - p_1)^{K-1} P_{sd} = p_1 (1 - p_1)^{K-1} (1 - p_2)$$

(11)

On the basis of (5) and (11), the packet delivery ratio is obtained by:

$$P = 1 - p_1 + p_1 (1 - p_1)^{K-1} (1 - p_2) \quad (12)$$

(1) Determination of relay node encode

We evaluated which relay node should be encoded.

Formula (12) does not contain the variable *He*. Thus, we were able to set up a packet delivery ratio regardless of the number of hops to a relay node that encodes the packet.

The packet delivery ratio is the same regardless of node encodes packet.

(2) Determination of the Path

Next, we evaluated whether a data packet should be transmitted on the path and in which conditions.

The formula which is transformed from (12) is given by:

$$P = 1 - p_1 \{1 - (1 - p_1)^{K-1} (1 - p_2)\} \quad (13)$$

For both two possible relative values the path $p_1 < p_2$ or $p_1 > p_2$, we evaluated, the packet delivery ratio increases.

If the path in which p_1 become smaller is determined, $(1 - p_1)^{K-1}(1 - p_2)$ increases. Therefore, P increases as p_1 decreases. Thus, we determined the path such that $p_1 < p_2$. If the packet loss rate is defined as equal among all the nodes, the path which forwards data packet is determined such that $H_1 < H_2$. Thus, we determined which data packet is forwarded on the path with fewer hops.

3.5 Packet Overhead

Next, we evaluated the packet overhead.

Packet Overhead O is set by setting X as all data packet, which should be sent is given by:

$$O = \frac{X}{K} \left\{ K \left(\sum_{j=1}^{H_1} (1 - p_{1j})^{j-1} + \sum_{j=2}^{H_e} (1 - p_{2j})^{j-1} \right) + \sum_{j=H_e+1}^{H_2} (1 - p_{2j})^{j-1} \right\}$$

(14)

(1) Determination of relay node encode

We decided on the basis of our evaluation, which relay node should be encoded.

We see that the packet overhead O decreased as the amount of *He* decreased as a result of (14). Thus, the relay node adjacent to the source node should encode the packet.

(2) Determination of the path

We evaluated whether a data packet should be transmitted on the path and in which conditions from number of hops.

Packet overhead increases because the number of packets that are forwarded decreases as the packet loss rate increases.

However, packet delivery ratio decreases as the number of packets that is forwarded decreases. Therefore, we did not find the packet delivery ratio because it is factor.

We see that O decreases when the path is determined such that $H_1 < H_2$. Thus, the data packet should be forwarded on the path in which number of hops is fewer.

3.6 Delay

To simplify the evaluation, the following items are defined as assumptions.

We evaluated the delay on the basis of the following parameters.

t_s : transmission interval[s]

t_n : wireless delay[s]

(fixed among all nodes)

The average delay, T_s, which forwards a data packet to a destination node is given by:

$$T_s = H_1 t_n (1 - p_1) \quad (15)$$

The average delay, T_e, the period from when a data packet is decoded from an encoding packet to another data packet is given by:

$$T_e = p_1(1-p_2)(1-p_1)^{K-1}\prod_{j=1}^{H_e}(1-p_{2j})(K-1) \times \sum_{i=1}^{X}((\max(H_1,H_2)t_n + t_s i)(1-\prod_{j=1}^{H_e}(1-p_{2j}))^{i-1}) \quad (16)$$

On the basis of (15) and (16), average delay is obtained by:

$$T = \frac{T_s + T_e}{P} = \frac{p_1(1-p_2)(1-p_1)^{K-2}(K-1)\left(\max(H_1,H_2)t_n + t_s\prod_{j=1}^{H_e}(1-p_{2j})^{-1}\right) + H_1 t_n}{p_1(1-p_2)(1-p_1)^{K-2}+1} \quad (17)$$

(1) Determination of relay node encoding

We evaluate which relay node should code.

We see that the delay T decreases because $\prod_{j=1}^{H_e}(1-p_{2j})^{-1}$ decreases as He decreases since (17).

Thus, the relay node adjacent to the source node should encode the packet.

(2) Determination of Path

Next, we evaluated whether the data packet should be forwarded on the path and in which conditions.

(16) with H_e=1 is given by:

$$T = \frac{p_1(1-p_2)(1-p_1)^{K-2}(K-1)\left(\max(H_1,H_2)t_n + t_s\right) + H_1 t_n}{p_1(1-p_2)(1-p_1)^{K-2}+1} \quad (18)$$

We evaluated the delay only when the number of hops was changed.

We assumed that p1 and p2 do not change even if the number of hops changes.

For either values of H, i.e., when $H_1 > H_2$ or when $H_1 < H_2$, we found that the delay decreases. In either case, T remains unchanged because max(H_1, H_2) does not change. Therefore, the delay T decreases as H1 decreases. Thus, the data packet should be forwarded on the path that has fewer hops .

The path on which the data packet is forwarded does not determine the change of the packet loss rate because of the dependence on the number of hops and on each delay time.

4 EXPERIMENT

We verified the validity of the result of the theoretical evaluation by carrying out a computor simulation. We used ns2, a discrete event simulator, [9]. The simulation topology is shown in Figure 3.

The simulation environment is shown in Table 1.

We only considered packet loss from data packets when we evaluated the data transmission rate

Packet Loss Rate is defined as being an equal value among all the nodes.

Table 1 Simulation parameters

Field [m]	1000 × 1000
Number of Nodes	7
Radio range [m]	250
Speed [km/h]	0
Simulation time [sec]	500
Data size [bytes]	512
Transport Protocol	UDP
Time between generating packet [s]	0.25

Packet Loss Rate [%]	0 ~ 50
Encoding parameter	(3,2,1)

4.1 Simulation results

In three evaluation models shown in Table 2, the theoretical result obtained from the formula and the simulation result are shown in Figure 6 through Figure 4.

Table 2 Evaluation model

	Path data packet is forwarded	Path encoding packet is forwarded	He
Model 1	Path 1	Path 2	1
Model 2	Path 1	Path 2	3
Model 3	Path 2	Path 1	1

・Packet Delivery Ratios

The results obtained from Formula (12) (Theory) and the simulation result (Simulation) are shown in Figure 4.

To determine the relay node that encodes the packet, the packet delivery ratio remains unchanged regardless of the number of hops to a relay node which encodes the packet as well as the result proven by the formula.

To determine the path on which the data packet is forwarded, Model 1 has a higher packet delivery ratio than that of Model 3. Thus, this proves that the method to forward data packet should be forwarded on the path in which the packet loss rate is low (the path with few hops) has higher packet delivery ratio.

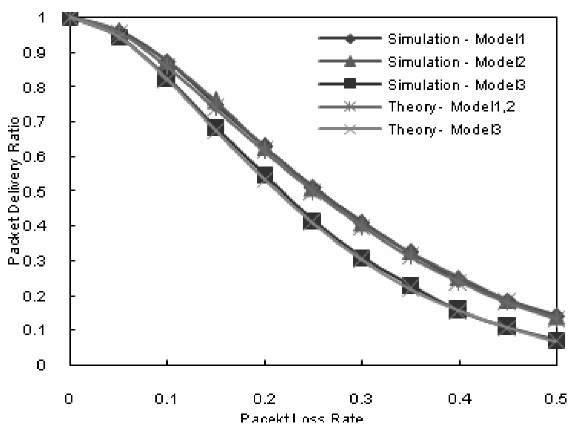


Figure 4 Packet Delivery Ratio

・Packet Overhead

Figure 5 shows the packet overhead.

The result obtained by (14) is the same as that obtained by using a simulation.

Model 1 has the smallest, and Model 3 has the largest packet overhead. Thus, we proved that the way to forward a data packet is to use the path in which the total number of hops is low and the number of hops to the relay node that encodes the packet is low has lower packet overhead.

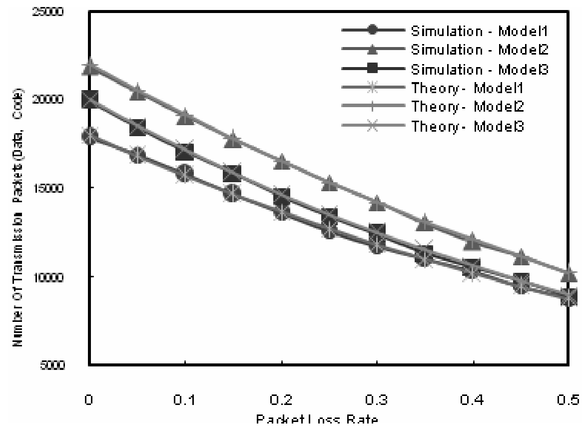


Figure 5 Packet Overhead

・Delay

The delay is shown in Fig. 6: the graph shown was obtained by using Formula (17) with t_s= 0.25 and t_n = 0.01 and the simulation result.

The graph obtained by using the formula (Theory) is not equivalent to the simulation result (Simulation) because we did not include the time needed to construct a path nor the validity of the set-up wireless communication delay between each node.

However, the result for the magnitude relation is same as that obtained by the simulation, i.e., by determining the path and the relay node encode.

Model 2 has a higher delay in comparison with that obtained by using Model 1. Thus, the delay increases with the number of hops to the relay node that encodes

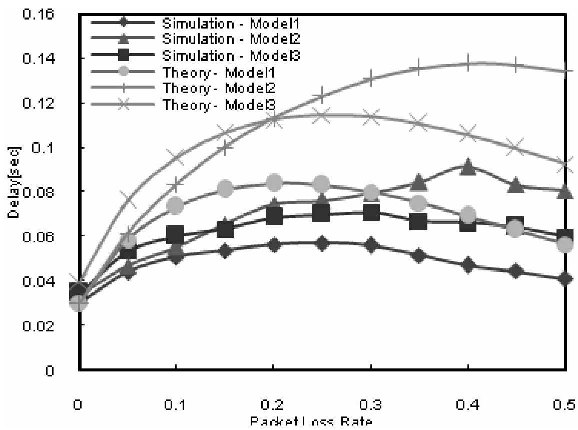


Figure 6 Delay

the packet. As a result, the relay node nearest the source node encodes the packet.

In addition, Model 3 has a higher delay than Model 1. Thus, this proved that the delay decreases when a data packet is forwarded on a path with fewer hops.

4.2 Summary

The data packet that should be forwarded on a particular path and in which conditions, and which relay node should encode a packet is shown in the results of the evaluation (Table 3).

Our method has a higher packet delivery ratio, a lower packet overhead.

The number of hop to a relay node that encodes packet has no relation to the packet delivery ratio. Furthermore, the relay node adjacent to the source node should encode the packet.

Table 3 Result

	Packet Delivery Ratio	Packet Overhead	Delay
Path data packet is forwarded (Packet Loss Rate)	Low	-	-
Path encoding packet is forwarded (Hop)	Few※	Few	Few
Number of hops to a relay node which encodes packet	independent	1	1

*When Packet Loss Rate between all nodes is equal.

5 CONCLUSION

We evaluated our proposed method by using network coding with a multipath routing environment.

We evaluated only two paths: the first path in which data packet is forwarded, and the second one in which the encoding packet is forwarded and limited by encoding parameters with (N, K, 1).

To determine the path on which the data packet is forwarded, we proved that the data packet should be forwarded on a path with few hops when the packet loss rate is equal among all nodes.

To determine which relay node should encodes the packet, we theoretically proved that the relay node adjacent to the source node encodes packet is better.

In addition, we proved the validity of the theoretical evaluation with a simulation.

6 FUTURE WORK

We evaluated a limited number of paths. In addition, all data packets and encoding ones are distributed and forwarded in each path.

However, to ensure that our work takes load balancing into consideration we should evaluate how a packet should be scheduled when more than three paths are constructed.

In addition, we proved that the packet delivery ratio is better when a data packet is forwarded on the path in which the packet loss rate is lower. However, the path is not determined if this rate is not measured. Thus, we need to investigate how to measure the Packet Loss Rate on each path.

REFERENCES

[1] T. Kagi, O. Takahashi, " Efficient Reliable Data Transmission using Network Coding in MANET Multipath Routing Environment", IPSJ, MBL44, pp.209-216(2008)

[2] R. Ahlswede, N. Cai, S. R. Li, and R. W. Yeung. "Network information flow", *IEEE Transactions on Information Theory*, (2001)

[3] D. Jhonson, Y. Hu, and D. Maltz, "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4", IETF-Request-for-Comments, rfc4728.txt. (2007).

[4] S. J. Lee, and M. Gerla "Split Multipath Routing with Maximally Disjoint Paths in Ad hoc Networks", ICC, pp. 3201-3205(2001)

[5] M. Kosugi, H. Higaki, "Data Message Transmission in MANET with Uni-Directional Links", IPSJ, MBL41, pp.103-106(2007)

[6] W. Lou, W. Liu, and Y. Zhang, "Performance Optimization using Multipath Routing in Mobile Ad Hoc and Wireless Sensor Networks", *Combinatorial Optimization in Communication Networks*, edited by M. Cheng, Y. Li, and D-Z. Du, Kluwer, (2006)

[7] R. Ma and J. Ilow, "Reliable Multipath Routing with Fixed Delays in MANET Using Regenerating Nodes," In IEEE LCN'03, (2003).

[8] R. Ma and J, Ilow, "Regenerating Nodes for Real-Time Transmissions in Multi-Hop Wireless Networks," In IEEE LCN'04, (2004).

[9] ns-2. http://www.isi.edu/nsnam/.

(Received September 27, 2008)
(Revised July 22, 2009)

Tomonori Kagi received master's degree in System Information Science from Future University Hakodate in 2009, and joined NTT DOCOMO Inc.. He is currently working at NTT DOCOMO Inc.

Osamu Takahashi received master's degree from Hokkaido University in 1975. He is currently a professor at the Department of System Information Science at Future University Hakodate. His research interest include ad-hoc network, network security, and mobile computing. He is a member of IEEE, IEICE, IPSJ.

A Design on Integrated Protocol for Communications and Positioning

Tomoya Takenaka†, Hiroshi Mineno‡, and Tadanori Mizuno†

†Graduate School of Science and Technology, Shizuoka University, Japan
‡Faculty of Informatics, Shizuoka University, Japan
{tomoya, mineno, mizuno}@mizulab.net

Abstract - The purpose of this paper is to introduce an integrated protocol for communications and positioning. The motivation for designing this integrated protocol comes from the recent research results for using location information. The objective of the integrated protocol is to enable simultaneous data exchange and location discovery. We describe the protocol stack of the integrated protocol and look at the facilities required for each layer. For the MAC layer, resource control for positioning is introduced. For the NWK layer, simultaneous localization and routing is discussed. Target tracking, an application of the integrated protocol, is also investigated.

Keywords: positioning, data communications, wireless multi-hop networks, ad-hoc networking

1 Introduction

Emerging wireless networking capabilities and micro-electronics technologies enable the provision of the various types of networks, such as ad-hoc and sensor networks. Zigbee [1] is one of these emerging standardized sensor network products. Once sensor nodes are deployed, they can automatically gather the sensing information for a observer. In such sensor networks, sensing data is expected to be bundled with location information to locate the event. To know the node positions, positioning techniques have been discussed in areas such as cellular communications [3] and wireless multi-hop networks [4].

The relationship between data communication protocols and positioning protocols is shown in Fig. 1. Location information is not only used for sensor networks, but also for improving networking performance. Location information helps to reduce redundant packets [12], [13]. In location-aided routing (LAR) [13], the expected zone based on the location of a destination node is defined to reduce packet flooding. A number of redundant packets is dropped by limiting the flooding zone.

In addition, location information contributes to improving energy efficiency. In geographical adaptive fidelity (GAF) [14], nodes can go to sleep to conserve energy. Since GAF defines the virtual grid based on the location information to find the nodes necessary for data delivery, it can maintain the data delivery over an extended network life time.

Another application of location information is in medium access control (MAC) protocol for collision avoidance. Location information enables nodes to know the direction of their neighbor nodes. In this case, the wireless communication range can be shrunk to avoid media access collision. In

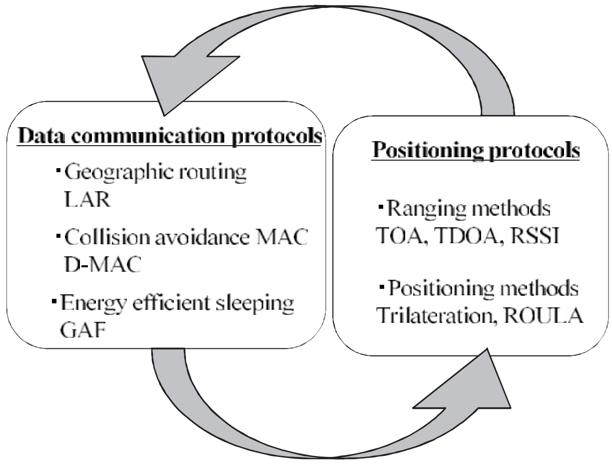


Figure 1: Relationship between data communication and positioning protocols.

[15], the authors proposed that a node sends packets by using a limited wireless range with a directional antenna to a receiver when the transmitter knows the receiver's location.

While location information is useful for data communication protocols, a positioning protocol that provides node positions must be able to obtain location information anywhere. As discussed in a lot of different literature, the global positioning system (GPS) is a simple solution for obtaining node positions. However, GPS cannot always provide location information, such as inside buildings.

A positioning protocol consists of two steps. First is estimating the distance (or ranging) by using time-of-arrival (TOA), time difference of arrival (TDOA), or received signal strength indicator (RSSI). Second is positioning the nodes to calculate the coordinates. Positioning methods for wireless multi-hop networks have been previously discussed [16], [17].

To enable nodes to obtain their positions at any place, a positioning protocol is required. The positioning protocol itself requires an overhead of ranging message that includes communicating nodes. Therefore, it is inefficient to design each data communication protocol and positioning protocol separately. Therefore, we designed an integrated data communication and positioning protocol.

ISSN1883-4566/09

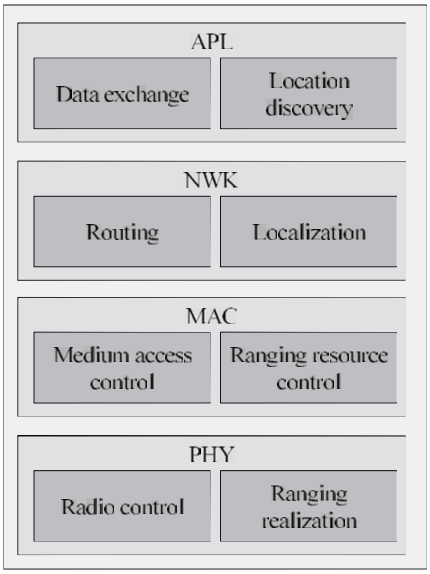


Figure 2: Protocol stack of integrated protocol for communications and positioning.

1.1 Outline of paper

Section 2 describes an overview of the integrated protocol for communications and positioning. The physical layer (PHY) and MAC layer are described in Section 2.2 and 2.3, respectively. Section 2.4 presents an overview of the network (NWK) layer, including an issue of the layer and our solution.

Target tracking which is an application of the integrated protocol is introduced in Section 3. A problem statement of target tracking is given in Section 3.1. We also propose cooperative target tracking method that uses the ranging capability in Section 3.2. The performance of cooperative target tracking is evaluated in Section 4.

Section 5 summarizes the paper and mentions our future work.

2 Integrated protocol for communications and positioning

2.1 Overview

The integrated protocol for communications and positioning is operated under wireless multi-hop network topology.

The protocol stack of the integrated protocol is shown in Fig. 2. The objective of the integrated protocol is to enable simultaneous data exchange and location discovery. It enables users to obtain data and location information in the application layer (APL).

To enable data communications, each layer has conventional data communication facilities. Ranging realization, ranging resource control, and localization are added to support location discovery. We then describe functionalities of the PHY, MAC, and NWK layers.

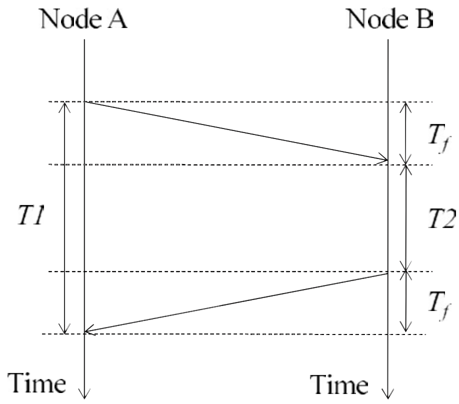


Figure 3: Ranging message sequence for TW-TOA.

2.2 PHY

Ranging realization is achieved in the PHY. IEEE 802.15.4a [9] standardized the ranging capabilities, which enable the nodes to estimate the node distance, although the implementation is optional. We then use the same mechanism of ranging capabilities as described in IEEE 802.15.4a specification.

Figure 3 shows the message sequence for the ranging realization by using a two-way time-of-arrival (TW-TOA) between two nodes, A and B. The propagation time T_f to estimate the node distance can then be written as

$$T_f = \frac{1}{2}(T1 - T2), \tag{1}$$

where $T1$ is the round trip time for node A and $T2$ is the reply time for node B. Here, we only consider the true times of the message arrivals. A distance can be derived from the propagation time, and TW-TOA enables the estimation of the node distance. TW-TOA can be achieved by observing the first arrival signal of the received signals, and it requires at least two messages between nodes.

One notable observation on the IEEE 802.15.4a specification is that it enables the node to estimate distances in the PHY. However, IEEE 802.15.4a does not support a positioning functionality. Therefore, a facility to calculate the node position is required to be defined on an upper layer.

2.3 MAC layer

Figure 4 shows the time slots in the MAC protocol for the integrated protocol. The ranging capability enables the tracking and navigation applications. In such applications, data may be needed to be sent without collisions. The functionality in the MAC layer is to schedule and control the ranging resource to conduct the tracking and navigation applications. The contention access period (CAP) is collision-based packet scheduling, and the contention free period (CFP) is collision-free packet scheduling, which is the same mechanism as described in IEEE 802.15.4 [8] specification. Although CFP is required to the synchronization between nodes, data can be sent without collision. The CFP provide a guaranteed time

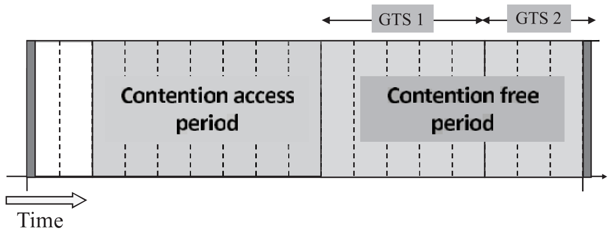


Figure 4: Time slots in MAC protocol for integrated protocol.

slot (GTS) to send packets periodically. Thus, CFP can be used for the tracking and navigation applications.

A challenging functionality for the MAC protocol is to design a mechanism to assign the time slot to send data with the precise positioning accuracy. In existing data communication protocol, delay and reachability are parameters to indicate the quality of service (QoS). Then the nodes will cooperate for improve the delay or reachability. In the integrated protocol, positioning accuracy is the parameter to indicate the QoS. Each time slot in the MAC layer is selected and scheduled in terms of the precise positioning accuracy. Then nodes will cooperate for precise positioning accuracy.

Figure 5 shows the example data relaying when three wireless nodes $A, B,$ and C are connected in the networks. Assume that node A will deliver the data to the node B. When latency for the data delivery is critical for the network, node A directly send the data to the node B. However consider that when an obstruction is existed between A and B and node distances are required to being precisely measured for the positioning. The distance measurement between A and B is degraded for the non-line-of-sight (NLOS). In this case, node A should use the node C to precisely measure the distances for node B. Therefore, a node has to change the destination for data relaying when data and positioning requests are mixed. The data flows are required to be controlled in accordance with QoS priority. We will present the case that nodes should cooperatively relay for precise positioning in a target tracking in Section 3. The simulation results presented in Section 4 reveal that the cooperative relaying through a line-of-sight (LOS) link improves the positioning accuracy.

2.4 NWK layer

The NWK layer provides the simultaneous routing and localization capability. The localization protocol estimates node positions. The motivation for developing localization protocol is wanting to know the node positions in multi-hop networks with only a small number of anchor nodes. An anchor node is one whose position is known in advance through such as GPS.

We previously developed optimized link state routing-based localization (ROULA) [17]. ROULA is independent of anchor nodes and can determine the correct node positions in a non-convex network topology. In addition, ROULA is compatible with the optimized link state routing (OLSR) protocol [21] and uses the inherent distance characteristic of multipoint relay (MPR) nodes.

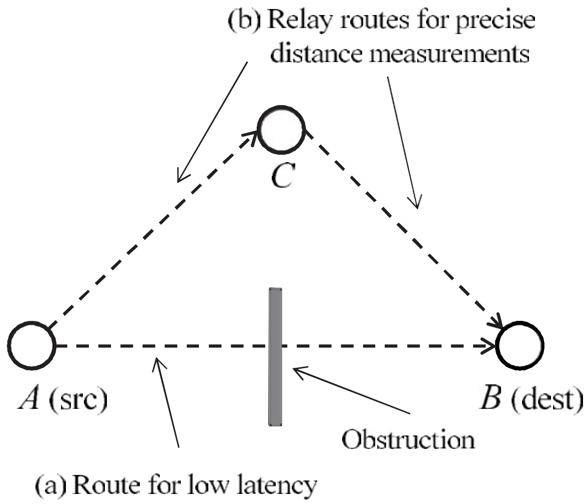


Figure 5: Example data relaying for (a) low latency and (b) precise distance measurements.

Our objective in developing the integrated protocol is to achieve simultaneous data exchange and location detection. The localization protocol consists of estimating distances and positioning. ROULA sends hello messages to estimate node distances, hence it can extract an overhead of routing protocol generated in the NWK layer.

Before discussing which routing protocol is suitable for the NWK layer, we first introduce the existing routing protocols. The routing protocol is one of the major issues in wireless multi-hop networks. In the Internet Engineering Task Force (IETF), the Mobile Ad-hoc Networks (MANET) Working Group [2] has been organized to address this issue.

There are mainly two types of routing protocols. One is a reactive protocol and the other is a proactive protocol. Ad hoc on-demand distance vector routing (AODV) [20] is one of the reactive protocols. In AODV, control messages are generated according to requests to detect and maintain the routes. Zigbee [1], which is sensor network product for the industry, uses the AODV protocol.

OLSR [21] is one of the proactive protocols. OLSR sends the control messages periodically to detect the shortest paths to nodes in the network. Nodes in OLSR select the MPR nodes as relay nodes. OLSR enables efficient flooding of messages by using MPR nodes.

Figure 6 shows both the OLSR and ROULA function modules. OLSR and ROULA exchange hello messages to find one-hop nodes. In addition, OLSR uses topology control (TC) messages to find the routes in the overall network. TC messages periodically flood the network, and thus they are compatible with the messages gathering local coordinates from all the nodes in the ROULA protocol.

We selected the OLSR protocol for the NWK layer. We are currently porting the ROULA protocol into the OLSR protocol. We are investigating how ROULA can be efficiently

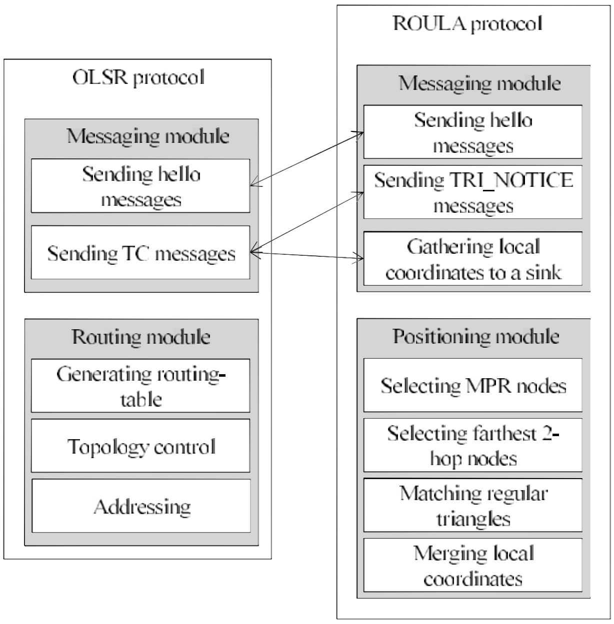


Figure 6: NWK function modules for integrated protocol. Arrows show compatibilities between OLSR and ROULA protocols.

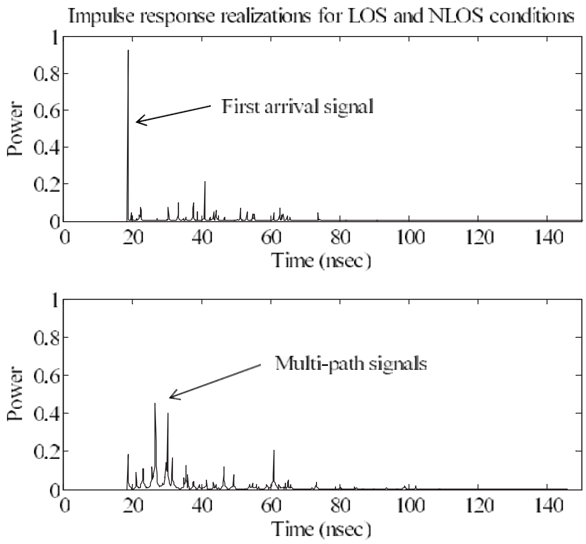


Figure 7: Time of arrival signals through LOS (top) and NLOS links (bottom).

integrated in the OLSR protocol [18], [19].

3 Target tracking application

3.1 Problem statement

The proposed integrated protocol can be used in such as equipment monitoring and security, emergency and logistics applications. Target tracking is an application of the integrated protocol. We focus on investigating target tracking using a ranging capability and describe how nodes on the integrated protocol are operated for target tracking.

One of the situations in which target tracking is used is inside a hospital. Target tracking allows for the position of tags with ranging capabilities to be monitored. Therefore, the positions of patients and doctors equipped with the tags are known at once even in emergency situations.

The problem with target tracking is how to estimate node positions sequentially. Although the problem to be solved is in a mobile node environment, we state the problem as a static location estimation for brevity.

Let us consider a two-dimensional positioning problem. Assume that at any time, the positions (x_i, y_i) for $i = 1...k$ reference nodes are known and the positions (x_i, y_i) for $i = k+1...n$ nodes are unknown. A typical location estimation using least-square is given by

$$\hat{p} = \arg\min \sum_{i=1}^{k} (r_i - \sqrt{(x_i - x)^2 + (y_i - y)^2})^2, \quad \text{for } k \geq 3, \tag{2}$$

where $\hat{p}(x, y)$ is the estimated node position and k is the number of reference nodes. The range measurements obtained from TOA estimation is r_i. Equation (2) provides a good solution for the estimating the position when the range measurements are done through LOS links. However, once the range measurements are done through NLOS links, the estimated position will be biased.

Let us consider that the range measurements are

$$\hat{r}_i = r_i + \begin{cases} e_i^{los}, & i = 1, 2, ..., M \\ e_i^{los} + b_i^{nlos}, & i = M+1, ..., N \end{cases}, \tag{3}$$
$$\text{where } e_i^{los} \sim \mathcal{N}(0, \sigma^2),\ b_i^{nlos} \sim \mathcal{E}(\mu).$$

LOS measurement noise e_i^{los} is modeled as a zero mean Gaussian distribution with variance σ^2. NLOS bias b_i^{nlos} is a positive distance bias introduced due to LOS blockage, and is modeled as an exponentially distributed random variable with mean μ.

Figure 7 illustrates some typical impulse response realizations for ranging at the receiver for LOS and NLOS conditions. In the case of a LOS link, the first arrival signal is normally precisely detected and it is identical to the shortest path signal of the sight (i.e. actual distance). However, when an obstruction blocks the LOS link, the first arrival signal received at the receiver may not be identical to the signal of the shortest path. Reflections from scatters are reached at the receiving nodes. Therefore the range measurement through a NLOS link results in introducing a bias error.

3.2 Cooperative target tracking

3.2.1 Notations and assumptions

Let us introduce the notations for the three nodes that we used in the proposal listed in Table 1. A target node (TN) is a node

Table 1: Notations for three nodes.

Notation	Description
Target node (TN)	Position should be tracked.
Mobile node (MN)	Relay for TN positioning
$\mathrm{RN_i} \mid i = 1 \ldots 3$	Position is known.

that should be tracked and estimated its position. A mobile node (MN) is a node that has the capability to move and has a role in assisting the TN tracking. A MN can be a human or a mobile robot. Reference nodes ($\mathrm{RN_i} \mid i = 1 \ldots 3$) are the nodes whose positions are known. Figure 8 shows the example topology of cooperative target tracking. In the target tracking, we make the following assumptions.

- TN, MN, and RN have TOA ranging capability.
- Identifications for LOS/NLOS links are achieved by using simple hypothesis testing of received signals in a mobile node environment [7].
- NLOS link is generated when an obstruction crosses a LOS link, as illustrated in Fig. 8.
- Both TN and MN are connected to three RNs, and TN is connected to MN.

3.2.2 Procedure

As discussed in Section 2.3, positioning accuracy is prioritized parameter for the integrated protocol. Introducing the positioning accuracy as a QoS parameter in the integrated protocol motivates the nodes in the network to cooperate for precise positioning. Conventional cooperative data relaying may introduce the delay because of data multi-hopping. However, we present here the cooperative relaying has a benefit for precise positioning. A cooperative target tracking using mobile nodes is then proposed to obtain precise positioning in an NLOS environment.

We describe the proposal by assuming static snapshot illustrated in Fig. 8. Three RNs, an MN, and TN are located on a field. The problem is to estimate the TN position. An obstruction is located between TN and $\mathrm{RN_2}$, resulting in the estimated position having bias due to the range measurement through the NLOS links. To avoid such a positioning situation that includes NLOS links, MN moves to an area to obtain LOS links from the TN and RNs when the TN has an NLOS link for positioning.

In Fig. 8, MN is located so that it has LOS links from the three RNs and TN. We describe an area that is guaranteed to obtain LOS links in Section 3.2.3. In this case, the RNs estimate the TN position by relaying the MN positioning. First, the RNs estimate the MN position with Equation (2) through LOS links of the indexes (1) as described in Fig. 8. Once MN obtains its own position, it is considered as a pseudo reference node. Then, $\mathrm{RN_1}$, $\mathrm{RN_3}$, and MN estimate the TN position with Equation (2) through the LOS links of indexes (2) as described in Fig. 8.

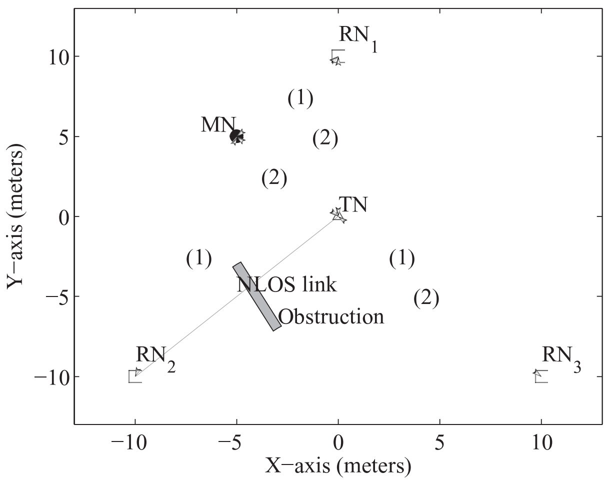


Figure 8: Illustration of target tracking. Messages are indexed in number order for cooperative target tracking.

Table 2: Notations for two methods.

Notation	Cooperation	Description
Conv.	No	Non-cooperative target tracking with only three RNs using EKF
Proposal	Yes	Cooperative target tracking with three RNs and a MN using EKF

3.2.3 Guaranteed area to obtain LOS links

To assist TN positioning, MN moves to the guaranteed area to obtain LOS links from three RNs and the TN. Figure 9(a) presents the guaranteed area to obtain LOS links from the three RNs. When the MN is placed in the shaded areas, it is guaranteed to obtain LOS links from the three RNs. Figure 9(b) presents the guaranteed area to obtain LOS links from the three RNs and the TN.

At present, we only consider a situation where the area to obtain LOS links from three RNs and a TN always exists, and the MN is placed that area.

To cope with general cases, we are investigating the following point:

- The condition for a guaranteed area where the LOS links can be obtained from k RNs and a TN (x, y), where $k \geq 3$, and x and y are variables.

4 Performance evaluation

4.1 Simulation setting

A performance evaluation through simulation proved the effectiveness of our proposed cooperative target tracking. We implemented extended Kalman filtering (EKF) [5] to estimate the moving TN positions. A node's motion can be considered

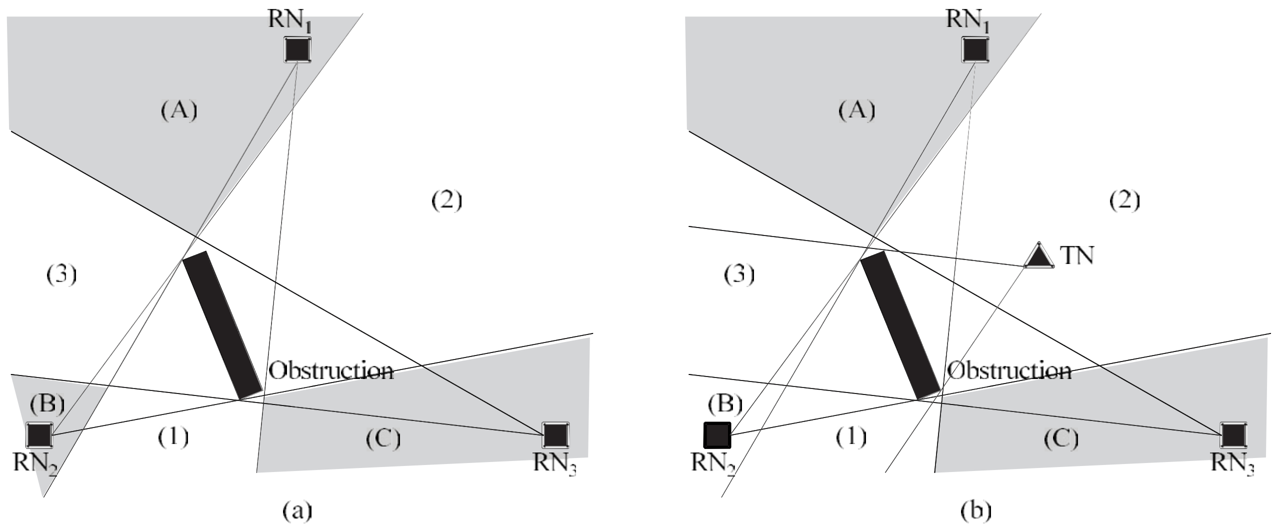


Figure 9: Illustration of areas (shaded) that are guaranteed to obtain LOS links from (a) three RNs and (b) three RNs and TN.

as a dynamic system for the function of time. Therefore, EKF can be applied to target tracking [5], [6].

Table 2 lists notations for two methods whose performances were compared. A conventional method represents non-cooperative target tracking using EKF that three RNs estimate moving TN through NLOS links. A proposed method represents cooperative target tracking using EKF that three RNs and MN jointly estimate moving TN through LOS links.

In addition, we implemented Cramer-Rao lower bound (CRLB). CRLB is the bound on a unbiased estimator [10]. This bound provides the best achievable performance [5]. Therefore, it can be used as a performance benchmark of the target tracking accuracy. Since covariance matrix of EKF in the absence of a process noise equation is identical to CRLB [11], we used the diagonal elements of the covariance matrix for calculating the CRLB.

4.2 Simulation results

The cooperative target tracking was performed by using a simulation. For LOS range measurement and NLOS bias error, we used $\sigma^2 = 0.1$ and $\mu = 1.6$ (m).

Figure 10 illustrates the topologies of the networks and estimated positions represented by triangles for the conventional method (left) and the proposed method (right). RNs are placed at $\mathrm{x}_1 = [0, 10]$, $\mathrm{x}_2 = [-10, -10]$, and $\mathrm{x}_3 = [10, -10]$ (m). The start point of TN is at $[0, 0]$ (m). TN moves in a straight line at a constant velocity of 1.0 (m/s). The observed time is 5 (s) with a sampling interval of 0.1 (s). The link between TN and RN_2 is blocked by the obstruction.

As shown in Fig. 10, the estimated positions using the conventional method had biased positioning errors from RN_2. We found that the estimated positions by the proposed method were much closer to the actual TN trajectories.

Figure 11 plots the cumulative distribution function (CDF) of the positioning errors that are defined as

$$\sqrt{(x_t - x_t^A)^2 + (y_t - y_t^A)^2}, \tag{4}$$

where (x_t, y_t) denotes the estimated position at time t and (x_t^A, y_t^A) denotes the actual position at time t. The conventional method had large positioning errors. The proposed method had less positioning errors and approached the CRLB.

Here, we only presented the one scenario. Various scenarios including several obstructions and random target motions will be investigated in the future. In addition to the positioning accuracy, the impact of delays in using the time slot for positioning in the MAC and data collections are needed to be considered.

5 Conclusion

We presented an overview of an integrated protocol for communications and positioning. The objective of this integrated protocol is to enable simultaneous data exchange and location discovery. We designed each layer of the integrated protocol. For the MAC layer, the resource control for positioning was introduced. For the NWK layer, we described the compatibility of OLSR with the localization protocol for enabling simultaneous routing and localization. We also discussed cooperative target tracking. Using a simulation, we found that cooperative target tracking achieved less positioning errors and approached the CRLB.

In future work, we will conduct detailed evaluations on node cooperation for precise positioning and implementation on simultaneous routing and localization.

REFERENCES

[1] Zigbee Alliance, http://www.zigbee.org/.

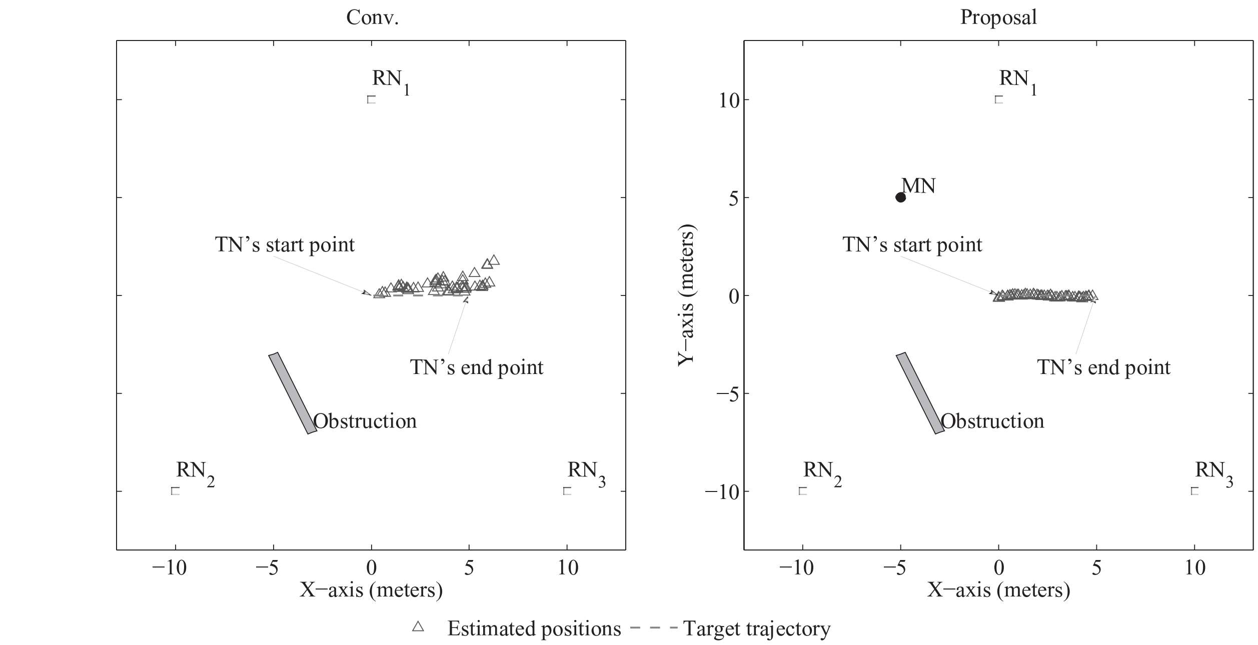


Figure 10: Estimated positions for conventional (left) and proposed (right) tracking methods.

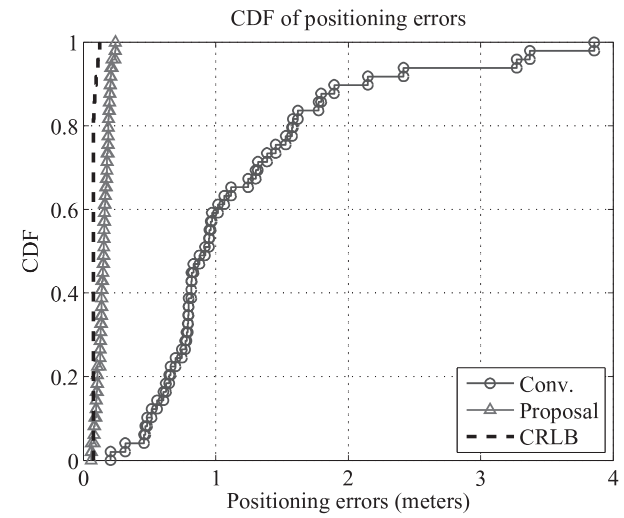


Figure 11: CDF of positioning errors.

[2] MANET Working Group, http://www.ietf.org/html.charters/manet-charter.html.

[3] J. J. Caffery and G. L. Stuber, "Overview of Radiolocation in CDMA Cellular Systems," *IEEE Communications Magazine*, vol. 36, pp. 38–45, 1998.

[4] N. Patwari, J. Ash, S. Kyperountas, A. O. Hero, R. M. Moses, and N. S. Correal, "Locating the Nodes: Cooperative Localization in Wireless Sensor Networks," *IEEE Signal Processing Magazine*, vol. 22, pp. 54–69, 2005.

[5] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, "Estimation with Applications to Tracking and Navigation," *Wiley-Interscience*, 2001.

[6] A. Smith, H. Balakrishnan, M. Goraczko, and N. Priyantha, "Tracking Moving Devices with the Cricket Location System," *Proc. of ACM International Conference On Mobile Systems, Applications And Services (Mobisys)*, pp. 190–202, 2004.

[7] J. Borris, P. Hatrack, and N. B. Manclayam, "Decision Theoretic Framework for NLOS Identification," *Proc. of Vehicular Technology Conference (VTC)*, vol. 2, pp. 1583–1587, 1998.

[8] IEEE Std 802.15.4, "Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)," *IEEE Computer Society*, 2006.

[9] IEEE Std 802.15.4a, "Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs), Amendment 1: Add Alternate PHYs," *IEEE Computer Society*, 2007.

[10] H. L. V. Trees, "Detection, Estimation and Modulation Theory, Part 1," *Wiley-Interscience*, 2001.

[11] J. H. Taylor, "The Cramer-Rao Estimation Error Lower Bound Computation for Deterministic Nonlinear Systems," *IEEE Transactions on Automatic Control*, vol. 24, pp. 343–344, 1979.

[12] J. Navas and T. Imielinski, "GeoCast – Geographic Addressing and Routing," *Proc. of ACM/IEEE Mobicom*, pp. 66–76, 1997.

[13] Y. Ko and N. H. Vaidya, "Location-aided Routing (LAR) in Mobile Ad Hoc Networks," *Proc. of ACM/IEEE Mobicom*, pp. 66–75, 1998.

[14] Y. Xu, J. Heidemann, and D. Estrin, "Geography-

informed Energy Conservation for Ad Hoc Routing," *Proc. of ACM Mobicom*, pp. 70–84, 2001.
[15] Y. Ko, V. Shankarkumar, and N. H. Vaidya, "Medium Access Control Protocols Using Directional Antennas in Ad Hoc Networks," *Proc. of IEEE Infocom*, vol. 1, pp. 13–21, 2000.
[16] A. Savvides, C. Han, and M. B. Strivastava, "Dynamic Fine-grained Localization in Ad-hoc Networks of Sensors," *Proc. of ACM/IEEE Mobicom*, pp. 166–179, 2001.
[17] T. Takenaka, H. Mineno, Y. Tokunaga, N. Miyauchi, and T. Mizuno, "Performance Analysis of Optimized Link State Routing-based Localization," *Journal of Information Processing Society of Japan (IPSJ)*, vol. 48, no. 9, pp. 3286–3299, 2007.
[18] K. Soga, T. Takenaka, Y. Terashima, Y. Tokunaga, H. Mineno, and T. Mizuno, "Proposal of the Fusion Technique of Localization and the Routing Protocol," *Proc. of IPSJ National Convention*, pp. 187–188, 2008 (in Japanese).
[19] K. Soga, T. Takenaka, H. Mineno, Y. Tokunaga, Y. Terashima, and T. Mizuno, "Proposal and Evaluation of Uniting Technique of Localization and Routing," *Proc. of Multimedia, Distributed, Cooperative and Mobile (DICOMO) Symposium*, pp. 120–126, 2008 (in Japanese).
[20] C. Perkins, E. Belding-Royer, and S. Das, "Ad Hoc On-Demand Distance Vector Routing," *IETF RFC 3561*, 2003.
[21] T. Clausena and P. Jacquet, "Optimized Link State Routing Protocol (OLSR)," *IETF RFC 3626*, 2003.

(Received October 15, 2008)
(Revised July 22, 2009)

Tomoya Takenaka received the B.I. and M.I. degrees in Informatics from Shizuoka University, Japan in 2005 and 2006, respectively. He is currently working towards the Ph.D. degree at Shizuoka University. He worked for Mitsubishi Electric Research Laboratories (MERL), Massachusetts, USA, as an intern from April to September 2007. He has been a Research Fellow of the Japan Society for the Promotion of Science (JSPS) since April 2009. His current research interests include localization and network protocol in wireless multi-hop networks. He is a member of IPSJ and Informatics Society.

Hiroshi Mineno received his B.E. and M.E. degrees from Shizuoka University, Japan in 1997 and 1999, respectively. In 2006, he received the Ph.D. degree in Information Science and Electrical Engineering from Kyushu University, Japan. Between 1999 and 2002 he was a researcher in the NTT Service Integration Laboratories. In 2002, he joined the Department of Computer Science of Shizuoka University as an Assistant Professor. His research interests include sensor networks as well as heterogeneous network convergence. He is a member of IEEE, ACM, IEICE, IPSJ and Informatics Society.

Tadanori Mizuno received the B.E. degree in industrial engineering from the Nagoya Institute of Technology in 1968 and received the Ph.D. degree in computer science from Kyushu University, Japan, in 1987. In 1968, he joined Mitsubishi Electric Corp. Since 1993, he is a Professor of Shizuoka University, Japan. Now, he is a Professor of graduate school of Science and technology of Shizuoka University. His research interests include mobile computing, distributed computing, computer networks, broadcast communication and computing, and protocol engineering. He is a member of Information Processing Society of Japan, the institute of electronics, information and Communication Engineers, the IEEE Computer Society, ACM and Informatics Society.

Modeling Language for LDAP and Automatic Production System

Kenji Saotome[*], Seiichi Kondo[**], Akihisa Onuma[***],
Takashi Komiya[***], Masahiko Ishino[****], Tsukasa Kudo[**],
Sanshiro Sakai[*****] and Tadanori Mizuno[******]

[*]Hosei Business School of Innovation Management, Japan
[**]Mitsubishi Electric Information Systems Corporation, Japan
[***]Mitsubishi Electric Corporation, Japan
[****]Fukui University of Technology, Japan
[*****]Faculty of Informatics, Shizuoka University, Japan
[******]Graduate School of Science and Technology, Shizuoka University, Japan

Abstract **-** LDAP Directory Service begins to be used as a tool for the development of Enterprise Information System. Nevertheless, there are not the standards of the model for the design of the Directory and generally incomplete diagrams of the Directory have been illustrated. The methods to design the Directory are expected. Then, we proposed the modeling language extending UML for the design of Directory Information Tree (DIT). We developed for trial the system that automatically generates the programs that manage the Directory, and evaluated it. We found that this system is enough applicable and efficient.
Keywords: Directory, LDAP, UML, MDA, Modeling.

1 INTRODUCTION

In late years the standardization of the directory service advances, and the directory service products which manage the data of the information system of the company come to be released, and it begins to be used. However, when the needs to build the original data structure by the system matter occurred, it becomes custom to show the incomplete figure of the degree to exemplify hierarchical structure and do it with a design document because there is not a standard design model peculiar to the directory. Therefore by the conventional construction technique of the directory, it becomes difficult to get the desired directory for the program developer and the user because we are not able to realize the smooth mutual understanding between the directory designer and the program developer and the user using it. There is the danger that a problem occurs just before the operation after the development.

We already proposed *Directory Modeling Language* specialized in the directory service based on the UML model, and produced the system(*Automatic Production System*) which generated *Directory Management Program* from this modeling language automatically and demonstrated the applicability of this system [1].

This research domain is the directory, programs automatic production and UML expansion, but the other researches corresponding to these all domains are not found. Therefore the theme of this research is an advanced one [2][3] [4].

In this time, we added the notation of the schema definition of the object class and the attribute type to the modeling language and realized the automatic generation function of the schema, and the system implementing the specifications of the practical use level was completed. By this report, we report about Directory Modeling Language and the specifications of Directory Management Program that Automatic Production System generates and the evaluation of this system.

The directory of this report has the structure of ITU-T (International Telecommunication Union-Telecommunication Standardization Sector) Recommendation X.500 series and the interface of LDAP(Lightweight Directory Access Protocol) defined by IETF(Internet Engineering Task Force)[5][6][7]. The directory modeling language to propose is based on UML(Unified Modeling Language) standardized in OMG(Object Management Group) [8][9].

2 MODELING LANGUAGE SPECS

This section describes the syntax specifications and the semantic specifications of Directory Modeling Language.

2.1 Syntax Specification

The model of Directory Modeling Language is expressed by the *class diagram* of UML. In the class, the *stereotype* and the *tagged value* is specified to show the role of the class. We show below the syntax specifications of this modeling language..

2.1.1 <<LDAP>> Class

(1)For a *model*, there must be only one *class* specifying an <<LDAP>> stereotype.

(2)In this class, a {LDAPRoot} tagged value must be specified. In {LDAPRoot} tagged value, a *character string* is specified.
(3)In this class, the *association* must not be specified.

2.1.2 <<LDAPEntry>> Class

(1)In this model the class name must be unique among the classes specified <<LDAPEntry>> stereotype.
(2)In the class, one or more {LDAPObjectClass} tagged values must be specified. In {LDAPObjectClass} tagged value, a character string is specified.
(3)The class has one or more *attributes*.
(4)The attribute consists of the attribute name and the *type*. The attribute name must be unique in this class.
(5)The type must be **String**, **byte[]**, **Collection**, **Collection<String>** or **Collection<byte[]>**.
(6)<<LDAPRDN>> stereotype must be specified for one or more attributes in a class.
(7) The association between classes is able to be specified, and the *multiplicity*, the *navigability* and the *role* name at the association end are able to be specified. When there are one or more associations in a class, the role name at each association end must be unique each other.
(8)The multiplicity of the association is specified the following one. The notation (*) means more than 0.

- 1
- *

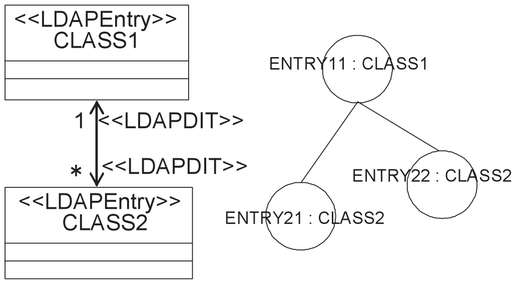
(9) The arrow of both directions or the arrow of single direction is specified for the navigability of the association.
(10) At the association end, one stereotype of the following kinds must be specified.

- <<LDAPDIT>>
- <<LDAPDN>>
- <<LDAPAttr>>

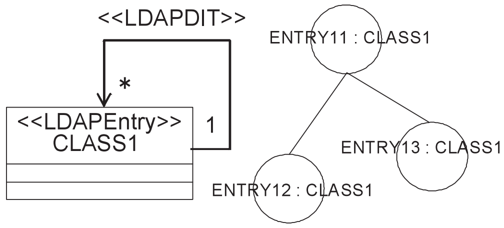
(11)When <<LDAPDIT>> stereotype at the association end is specified, <<LDAPDIT>> stereotype must be specified at another association end if nessesary. The multiplicity at this association must be * vs. 1.
(12) At the association end of <<LDAPAttr>> stereotype, {LDAPKey} tagged value must be specified.

2.1.3 <<LDAPDefAttributeTypeS>> Class

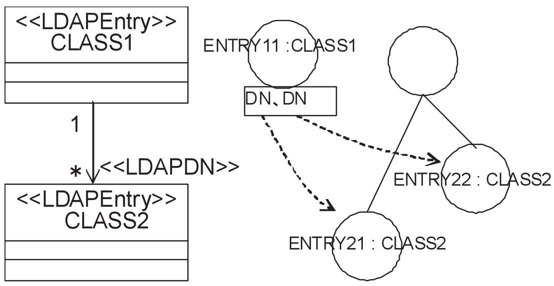
(1)In this class, a {LDAPSyntax} tagged value must be specified. A character string is specified in {LDAPSyntax} tagged value.
(2)The class has one or more attributes.
(3)The attribute consists of the name and the type.
(4)The attribute name must be unique among the attribute names described in the classes of <<LDAPDefAttributeTypeS>> stereotype.
(5)In the type, **void** or **Collection** is specified .



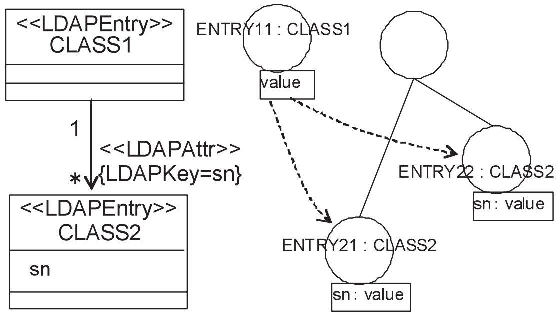
(a) <<LDAPDIT>> and DIT.



(b) <<LDAPDIT>> and DIT(Self Assoc).



(c) <<LDAPDN>> and DIT.



(d) <<LDAPAttr>> and DIT.

Figure 1 : Modeling Language and DIT.

(6)In this attribute, if necessary, {LDAPEquality}, {LDAPOrdering} or {LDAPSubstr} tagged value is specified.
(7)In this class, the association must not be specified.

2.1.4 <<LDAPDefObjectClass>> Class

(1)In this class, a {LDAPSuperior} tagged value must be specified. In {LDAPSuperior} tagged value, a character string is specified.
(2)A class has one or more attributes.
(3)The attribute consists of the attribute name and the type.
(4)The attribute name must be unique in a class.
(5)**void** must be specified if specified.
(6)For each attribute, <<LDAPMust>> stereotype is able to be specified.
(7)In this class, the association must not be specified.

2.2 Semantic Specifications

2.2.1 Class

2.2.1.1 <<LDAP>> Class

In the class specified <<LDAP>> stereotype, the information of the whole model is specified.

In {LDAPRoot} tagged value specified in this class, *DN(Distingushed Name)* which is top entry of all directory entries belonging to the classes of this model is specified.

2.2.1.2 <<LDAPEntry>> Class

The class specified <<LDAPEntry>> stereotype defines the directory entry. In the class, an *object class* of the directory constituting an entry belonging to the class is specified in the {LDAPObjectClass} tagged value. One or more {LDAPObjectClass} tagged values are able to be specified, but **top** object class is omitted.

The each attribute name defined in the class must be the *attribute type* of the directory to belong to one of the object classes specified in {LDAPObjectClass} tagged value.

The type of the attribute defined in the class is the type of JAVA used in APIs which Automatic Generation System generates.

When the *attribute syntax* shows binary data such as **Binary**, **Octet String** or **Certificate**, **byte[]** is specified when a single value, and **Collection<byte[]>** is specified when multi values. When the attribute syntax shows string data such as **Directory String**, **Boolean** or **Integer**, string is specified when a single value, and **Collection** or **Collection<String>** is specified when multi values.

<<LDAPRDN>> stereotype shows that it is an attribute type to become *RDN (Relative Distinguished Name)* of the entry.

2.2.1.3 <<LDAPDefAttributeTypeS>> Class

The class specified <<LDAPDefAttributeTypeS>> stereotype defines the user-defined attribute type with the *attribute syntax*. The attribute syntax is specified in {LDAPSyntax} tagged value of the class. The attribute syntax (the **SYNTAX** keyword in the attribute type definition of the directory) is specified in {LDAPSyntax} tagged value.

The attribute of the class shows the attribute type of the directory, and the attribute name shows the name of attribute type. In the type, **void** must be specified when attribute type of single value, and **Collection** must be specified when attribute type of multi values.

If necessary, {LDAPEquality}, {LDAPOrdering} and {LDAPESubstr} tagged value are able to be specified in each attribute of the class. {LDAPEquality}, {LDAPOrdering} and {LDAPESubstr} tagged value show the *matching rule* of **EQUALITY** , **ORDERRING** and **SUBSTR** each.

For the character string to specify in these tagged values, the keyword of the matching rule of the attribute type definition of the directory must be specified.

2.2.1.4 << LDAPDefObjectClass >> Class

The class specified <<LDAPDefObjectClass>> stereotype defines user-defined object class. In {LDAPSuperior} tagged value specified in the class, the superior object class (the keyword **SUP** of the object class definition) is specified.

The attribute of the class shows the attribute type to belong to this object class. The attribute name shows the attribute type name. <<LDAPMust>> stereotype in each attribute shows that this attribute type is the required attribute type of this object class.

2.2.2 Association

The association shows the association between the entries. Either of <<LDAPDIT>>, <<LDAPDN>> or <<LDAPAttr>> stereotype is specified at the association end to show the implementation of the association of the directory.

2.2.2.1 Association of <<LDAPDIT>>

<<LDAPDIT>> stereotype shows that the association is implemented by *DIT (Directory Information Tree)*. The multiplicity of this association must be * vs. 1. This association is implemented by the method that an entry belonging to the class of the association end specified “1” is posted as the direct upper entry of the entry belonging to the class at the other association end.

When there are more than 2 associations by <<LDAPDIT>> stereotype in a class, this class cannot have more than 2 association that have the “*” multiplicity among these association ends at this class side. Figure 1(a) shows an example of the class to use <<LDAPDIT>> stereotype and DIT implemented. The implementation of the association by DIT is the most natural structure for the directory. As to express the organization of the company, the self association to associate with the own class is the typical example associated by DIT of the directory. Figure 1(b) shows the class diagram expressing the self association and DIT implemented.

2.2.2.2 Association of <<LDAPDN>>

The <<LDAPDN>> stereotype shows that the association is implemented by DN. An entry belonging to a class has DN of an entry belonging to the class at the other association end. This implementation resembles the method that is used for the relations of the group and the member to be called the static group in the directory. Figure 1(c) shows the class diagram to use the <<LDAPDN>> stereotype and DIT implemented. The dotted lines in this figure are the notation to show the associated entries, and are not to constitute the real DIT.

2.2.2.3 Association of <<LDAPAttr>>

The <<LDAPAttr>> stereotype shows that the association is implemented using an attribute value of the attribute type. {LDAPKey} tagged value is specified on the association end. In {LDAPKey} tagged value, the attribute name to use for the association among the attributes of the class at the other association end is specified. When the value for this association in the entry is equal to the attribute value in the attribute type specified by the {LDAPKey} tagged value belonging to the class of the other association end, it is considered that these entries are associated. This implementation resembles the relations of the primary key and the foreign key of the relational data base. Figure 1(d) is the figure of the class diagram specified <<LDAPAttr>> stereotype and DIT implemented.

3 DIRECTORY MANAGEMENT PROG

3.1 Outline

We developed for trial the system(Automatic Production System) to generate automatically the program(Directory Management Program) to manage the directory from the model of Directory Modeling Language.

Directory Management Program has the following specifications so that assuming trial.

- Only import by batch of all data
- Only access by the application program

Directory Management Program consists of *Directory Loading Compiler*, *Directory Access API* and *User-defined Schema Information.*

The example of figure 2 assumes the directory which manages the information of the organization of the company, the region where it is located at and the employee who belonged to it. We select the standard attribute type as much as possible, but define the user-defined attribute type when we cannot use it. In figure 2, **addExtTelNumber**, **addExtFaxTelNumber**, **addCn** and **addSn** of **Employee** are the user-defined attribute type. **addExtTelNumber** and **addExtFaxTelNumber** show the extension telephone or FAX number. **addCn** and **addSn** are additional **Cn** and **Sn**. For example these are defined for the purpose of keeping the full name that includes GAIJI with the character code set (Shift-JIS etc) except utf-8.

Figure 3 describes the definition information of these user-defined attribute types. **addExtTelNumber** and **addExtFaxTelNumber** are defined as the attribute type of the <<LDAPDefAttributeTypeS>> class which name is **DirectoryString**. **addCn** and **addSn** are defined as **Binary** because they are expressed with the character code set except utf-8. The **addOrgPerson** object class that these user-defined attribute types belongs to is defined in the <<LDAPDefObjectClass>> class.

3.2 Directory Loading Compiler

3.2.1 Characteristics

Directory Loading Compiler has the following characteristics.

(1)Incomplete information is not left in the directory because the loading script can detect the structural error of DIT at the compilation time.

(2)The loading script can check the relation of definition and reference about the association.

(3)To use the LDIF form it is necessary to be specified the DN of the each entry itself and the related entry, but to use the loading script the loading data can be specified without being conscious of DN

3.2.2 Specifications

Directory Loading Compiler interprets the loading script for the loading directory data by batch, and generates the loading data of the LDIF form. Figure 4 is the script described in the specifications of Directory Loading Compiler generated by the model of Figure 2. **Region**, **Section** and **Employee** of Figure 4 is the class name, and the entry corresponding to this each class is specified in **#entry()**. In **#entry()**, the attribute type and the attribute value is specified in the form of **parameter=value**. For example, **st="Tokyo"** of the entry of **Region** shows that **st** is the attribute type and **Tokyo** is the attribute value.

At the following **{}** of **#entry()**, the entry being the lower entry in DIT is specified. The entry of **ou="sales department"** and **ou="Development Division"** of **Section** class becomes a direct lower entry which RDN is **st="Tokyo"**. In this way, the DN of each entry is decided by expressing DIT structure with the nest and RDN of each entry.

The role name in the class diagram is used to express the association except DIT. The **section** parameter

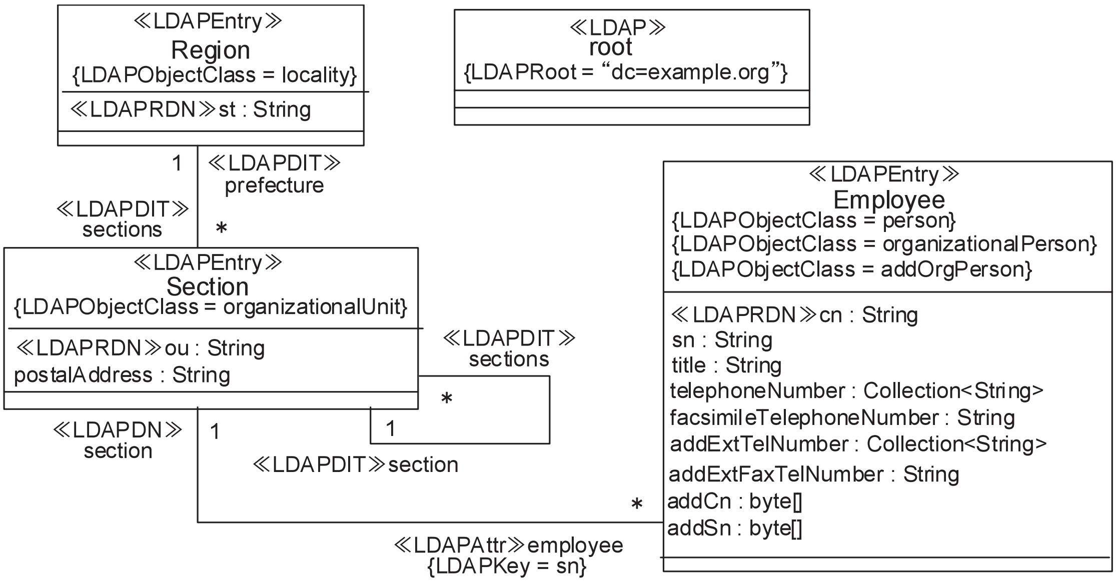


Figure 2 : DIT Model by Directory Modeling Language.

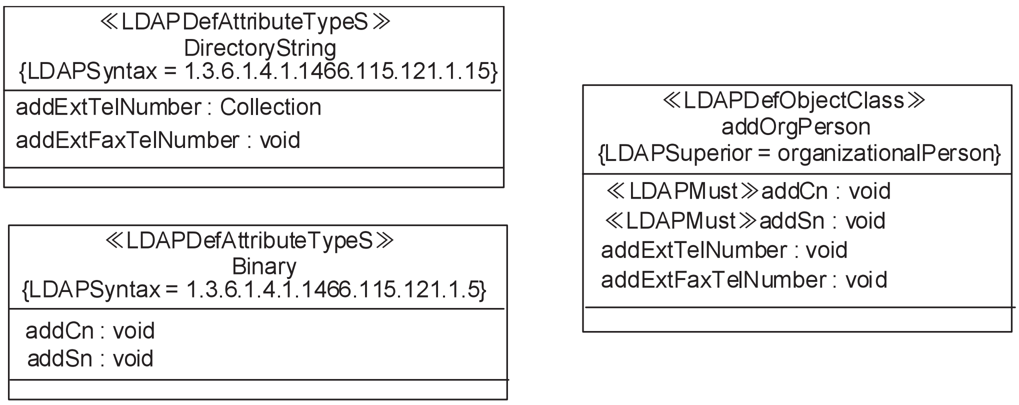


Figure 3 : Scheme Definition Model by Directory Modeling Language

specified in the entry of **Employee** class is the role name specified at the <<LDAPDN>> stereotype of the class diagram, and the associated entry at this parameter is specified.

3.3 Directory Access API

3.3.1 Characteristics

Directory Access API has the following the characteristics.

(1)The API uses JNDI (JAVA Naming and Directory Interface) which is JAVA standard interface of LDAP directory access and is implemented as the API which does not depend on the directory server.

(2)Directory Access API can be accessed with the interface of the design pattern adopted in EJB (Enterprise JavaBeans).

3.3.2 Specifications

Directory Access API has the functions to get the entry, the attribute value of the entry and the associated entry to the entry and so on.

Figure 5 shows the function of the API to access the entry belonging to **Section** class of Figure 2. **findByPrimaryKey()** method of **SectionHome** class gets the instance of **Section** class by the attribute type of RDN. **findAll()** method get all the entries belonging to **Section** class. **getDn()**, **getOu()** and **getPostalAddress()** of **Section** class get the attribute value of the entry. **getRegion()**, **getSection()**, **getSections()** and **getEmployee()** get the associated entry. **SectionDTO** class is the JavaBeans implemented **Java.io.Serializable** and has all of the attribute values.

```
Region {
   #entry (st="Tokyo") {
      Section {
         #entry (ou="Sales", postalAddress="Shinjuku-ku, Tokyo", employees=<sn="A"> <sn="B">) ;
         #entry (ou="development", postalAddress="1, Chiyoda-ku, Tokyo", employees=<sn="C">) {
            Section {
               #entry (ou="development-1", postalAddress="2, Chiyoda-ku, Tokyo", employees=<sn="D"> <sn="E">) ;
               #entry (ou="development-2", postalAddress="3, Chiyoda-ku, Tokyo", employees=<sn="F"> <sn="G">) ;
 ……..……
}
Employee {
   #entry (cn="A", sn="A", title="chief", telephoneNumber="03-1111-0001", facsimileTelephoneNumber="03-1111-1001",
      addCn="gmA=", addSn="gmA=", addExtTelNumber="11-0001", addExtFaxTelNumber="11-1001", section=<ou="Sales">) ;
   #entry (cn="B", sn="B", title="staff", telephoneNumber="03-1111-0002",  facsimileTelephoneNumber="03-1111-1002",
      addCn="gmE=", addSn="gmE=", addExtTelNumber="11-0002", addExtFaxTelNumber="11-1002", section=<ou="Sales"> ) ;
 ……..……..
   #entry (cn="I", sn="I", title="staff", telephoneNumber="06-2222-0002",  facsimileTelephoneNumber="03-2222-1002",
      addCn="gmg=", addSn="gmg=", addExtTelNumber="22-0002", addExtFaxTelNumber="22-1002", section=<ou="Personnel"> ) ;
}
```

Figure 4 : Input of Directory Loading Compiler.

3.4 User-defined Schema Information

3.4.1 Characteristics

User-defined Schema Information has the following characteristics.

(1)The User-definition Schema Information does not depend on the directory server because it is generated by the standard LDIF form.

3.4.2 Specifications

The User-defined Schema Information consists of the data of the LDIF form to register the user-defined attribute type generated from the information of the class of <<LDAPDefAttributeTypeS>> stereotype and the user-defined object class generated from the information of the class of <<LDAPDefObjectClass>> stereotype. Figure 6 is the LDIF formed data of the attribute type definition generated from the <<LDAPDefAttributeTypeS>> class of Figure 3. Figure 7 is the LDIF formed data of the object class definition generated from the <<LDAPDefObjectClass>> class of Figure 3.

4 EVALUATION

In comparison with the existing application program using the directory, we constructed the directory, tested the function and measured the performance.

The object of the evaluation is Information Leakage Prevention Solution of Mitsubishi Electric Corporation (Mitsubishi Solution System as follows)[10]. We describe the functionally equal directory with Directory Modeling Language for the organization and the employee information of the directory of Mitsubishi Solution System, compare the performance of the each program that we make with Directory Access API and the API of Mitsubishi Solution System.

```
SectionHome
-----------------------------------------------
SectionHome ( ctx: DirContext ) : SectionHome
findByPrimaryKey ( ou: String ) : Section
findAll () : Collection

Section
-----------------------------------------------
Section ( ctx: DirContext, dn: String, ou: String,
     postalAddress: String, employees: Collection )
     : Section
getDn () : String
getOu () : String
getPostalAddress () : String
getRegion () : Region
getSection () : Section
getSections () : Collection
getEmployee () : Collection

SectionDTO
-----------------------------------------------
SectionDTO ( ou: String,
     postalAddress: String ) : SectionDTO
getOu () : String
setOu ( ou: String ) : void
getPostalAddress () : String
setPostalAddress ( postalAddress: String ) : void
```

Figure 5 : Methods of Class "Section".

4.1 User-defined Schema definition

We defined 35 user-defined object classes and 136 attribute types of Mitsubishi Solution System and were able to confirm the same schema definition.

By this definition, the classes in the model that we described by Directory Modeling Language were 7 <<LDAPDefAttributeTypeS>> classes and 35 <<LDAPDefObjectClass>> classes.

What 136 attribute types are able to be expressed by 7 <<LDAPDefAttributeTypeS>> classes shows that the attribute syntaxes of these attribute types are 7 kinds.

4.2 Building DIT

We were able to confirm that the DIT approximately same as Mitsubishi Solution System was build.

There were 9 <<LDAPEntry>> classes, 30 attributes and 13 associations in the model that we described.

4.3 Performance Measurement

4.3.1 Measurement Method

4.3.1.1 Data Structure

We assume the company of 5,000 employees and compare the performance using data for the measurement such as Table.1.

4.3.1.2 Evaluation Programs

We assume the implementation of the address book by the directory and use the following two programs for the performance comparison.

(1) Program1

This is the program that is assumed to find the employees from the organization hierarchy. While referring in sequence from the main office to the lower organization, this program displays the data of the organizational unit and all the employees belonging to there. The entries to access are 781 organizational units and 5156 employees.

(2) Program2

This is the program that is assumed to directly find an employee by the attribute. This program generates an employee name at random and acquires the data of the employee with the employee name as a key. The entries to access are 5000 employees, 5000 organizational units and 4,999 upper heads except the president.

4.3.1.3 Measurement environment

We connect two following computers in 100BaseT and access from PC that the evaluation programs execute to the directory server in LDAP interface.

```
dn: cn=schema
changetype: modify
add: attributetypes
attributeTypes: ( addExtTelNumber-oid
 NAME 'addExtTelNumber'
 DESC 'User Defined Attribute'
 SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
 X-ORIGIN 'user defined' )
attributeTypes: ( addExtFaxTelNumber-oid
 NAME 'addExtFaxTelNumber'
 DESC 'User Defined Attribute'
 SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
 SINGLE-VALUE
 X-ORIGIN 'user defined' )
attributeTypes: ( addCn-oid
 NAME 'addCn'
 DESC 'User Defined Attribute'
 SYNTAX 1.3.6.1.4.1.1466.115.121.1.5
 SINGLE-VALUE
 X-ORIGIN 'user defined' )
attributeTypes: ( addSn-oid
 NAME 'addSn'
 DESC 'User Defined Attribute'
 SYNTAX 1.3.6.1.4.1.1466.115.121.1.5
 SINGLE-VALUE
 X-ORIGIN 'user defined' )
```

Figure 6 : LDIF for Definition of Attribute Types.

```
dn: cn=schema
changetype: modify
add: objectclasses
objectClasses: ( addOrgPerson-oid
 NAME 'addOrgPerson'
 SUP organizationalPerson STRUCTURAL
 MUST ( addCn $ addSn )
 MAY ( addExtTelNumber
 $ addExtFaxTelNumber )
 X-ORIGIN 'user defined' )
```

Figure 7 : LDIF for Definition of Object classes.

(1)The PC that the evaluation programs execute

H/W	CPU: Intel Pentium4 2.8GHz	
	Memory: 760MB, HDD: 35GB	
S/W	Java 1.4.2 06	
	Windows XP Professional	

(2)The directory server

H/W	CPU: Intel Xeon 3.2GHz
	Memory: 2GB, HDD: 292GB
S/W	SunONE Directory Server 5.2
	Windows Server 2003

4.3.2 Result

Table 2 shows the measurement result. In Table 2, MMS is Mitsubishi Solution System, and DMP is Directory Management Program.

4.3.3 Discussion

We found that Directory Modeling Language has enough description ability of the application system, because with Directory Modeling Language, we was able to build DIT approximately same as Mitsubishi Solution System.

We found that Directory Access API is implemented by approximately same method as the API of Mitsubishi Solution System in the access of the directory, because about the number of the entries sent to PC, the difference for program1 is 781 and the difference for program2 is only 81.

We found that Directory Access API can work with practical performance, because the total of the execution times of Program1 and Program2 is same as Mitsubishi Solution System.

We show the evaluation for the execution times of both. The differences of Directory Management Program and Mitsubishi Solution System about the processing to influence performance are the follows.

- The APIs of Directory Management Program gets all attribute's values of the entry, but some of the APIs of Mitsubishi Solution System can get a part of the attribute's values. Therefore Directory Management Program becomes disadvantageous in the quantity of data sent to PC.
- When Mitsubishi Solution System acquires the entry, it performs the access of the directory without the transfer to check the contradiction of the directory structure. The check is unnecessary for Directory Management Program because this system creates the correct data with Directory Loading Compiler. Therefore Mitsubishi Solution System becomes disadvantageous in the number of the access of the directory.

In Program1, Mitsubishi Solution System uses the many APIs to get a part of the attribute's values of the entry for the implementation of this system. The influence surfaced in Program1.

Program2 sends many entries to PC. Mitsubishi Solution System performs many accesses of the directory to check the contradiction of the directory structure. The influence surfaced in Program2.

5 CONCLUSION

We found that this system is enough applicable and efficient. By using the result of this work, the improvement of the quality, the productivity and the maintenancibility can be expected.

We will implement OCL(Object Constraint Language) for the expansion of Directory Modeling Language in the future[11]. OCL is the function taken for the standard in UML2.0 and the language to describe the limitation and the query of the UML model.

Table 1 : Data Structure of Measurement

organization	org units	members	Title
	1	1	president
head office	5	5	vice president
division	25	25	general manager
deaprtment	125	125	director
section	625	5,000	manager, staff
TOTAL	781	5,156	

Table 2 : Result of Measurement

Program	system	time (minutes)	num of entries sent to PC
Program1	MMS	20	19,842
	DMP	24	19,061
Program2	MMS	54	40,081
	DMP	50	40,000

Now we have the most basic function of the acquisition of the entry such as the follows.

- **findByPrimaryKey()**
  find the entry by the primary key(RDN) in the class.
- **findAll()**
  find the all entry in the class.

When the search by the free search condition is necessary, it assumes that user oneself learn the implementation of the Directory Access API and realize it. In the application system, it is necessary to acquire the entry by the complex search condition of the attribute value etc. We will implement the search with the free search condition by OCL.

REFERENCES

[1] K.Saotome, S.Kondo, A.Oonuma, T.Komiya, S.Sakai, T.Mizuno : "Modeling Language for LDAP and Automatic Production System of Directory Management Program", Information Processing Society of Japan, Database, Vol47, No.SIG13(TOD31), pp.28-39 (2006).

[2] Dragan Milicev : "Automatic Model Transformations Using Extended UML Object Diagrams in Modeling Environments", IEEE TRANSACTIONS ON SOFTWEAR ENGINEERING VOL.28 NO4, pp.413-431, 2002.

[3] Ludovic Apvrille, Jean-Pierre Court : "TURTLE A Real-Time UML Profile Supported by a Formal Validation Toolkit", IEEE TRANSACTIONS ON SOFTWEAR ENGINEERING VOL.30 NO7, pp.473-487, 2004.

[4] MDA(Model Driven Architecture), UML Profile for EDOC, UML Profile for EAI. http://www.omg.org/mda/specs.htm#Profiles

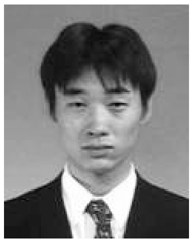
[5] ITU-T(International Telecommunication Union-Telecommunication Standardization Sector) . http://www.itu.int/ITU-T/
[6] IETF(Internet Engineering Task Force) . http://www.ietf.org/
[7] Tim Howes : "Understanding and Deploying Ldap Directory Services", Macmillan Technical Pub, 2003.
[8] OMG(Object Management Group). http://www.omg.org/
[9] UML(Unified Modeling Language). http://www.uml.org/
[10] Information Leakage Prevention Solution of Mitsubishi Electric Corporation. http://global.mitsubishielectric.com/bu/security/rd/rd07_02.html
[11] OCL(Object Constraint Language). http://www.omg.org/technology/documents/formal/ocl.htm

(Received October 17, 2008)
(Revised July 17, 2009)

Kenji Saotome received a BE from the Osaka University, Japan in 1979, and a Dr.Eng in Information Engineering from the Shizuoka University, Japan in 2008. From 1979 to 2007, he was with Mitsubishi Electric, Japan. Since 2004, he has been a professor of Hosei business school of innovation management. His current research areas include LDAP directory applications and single sign-on system. He is a member of the Information Processing Society of Japan.

Seiichi Kondo received the master's degree in information science from Kyoto University, Japan, in 1984. Currently, he is a senior engineer in Mitsubishi Electric Information Systems Corporation. His research interests include information security systems, identity management, and access control.

Akihisa Onuma received a degree from the Waseda University in 1986, currently, he is a engineer in Mitsubishi Electric Corporation, Japan. His current research areas include data modeling and single sign-on system.



Takashi Komiya received a MA from the SAGA University, Japan in 2000, currently,he is a senior engineer in Mitsubishi Electric Corporation, Japan. His current research areas include overall information security systems, especially identity management and authentication systems .

Masahiko Ishino received the master's degree in science and technology from Keio University in 1979 and received the Ph.D. degree in industrial science and engineering from graduate school of Science and technology of Shizuoka University, Japan, in 2007. In 1979, he joined Mitsubishi Electric Corp. Since 2009, he is Professor of Fukui University of Technology. Now, His research interests include management information system, industrial engineering, application system of data-mining, and information security system. He is a member of Information Processing Society of Japan, Japan Industrial Management Association, Japan Society for Management Information.

Tsukasa Kudou received the master's degree in Engineering from Hokkaido University in 1980 and received the Dr. degree in industrial science and engineering from Shizuoka University, Japan, in 2007. In 1980, he joined Mitsubishi Electric Corp. He was the researcher of parallel computer architecture and the engineer of business software. Since 2005 he is the engineer of Mitsubishi Electric Information Systems Corp. Now, his research interests include database application and system quality assurance. He is a member of IEIEC and Information Processing Society of Japan.

Sanshiro Sakai received the B.E., M.E and Ph.D degrees from Shizuoka University, Japan, in 1979, 1981 and 1984, respectively. Currently he is a professor in the Faculty of Informatics, Shizuoka University. His current research interests are in computer supported collaborative learning, programming language education environments and understanding of computer programs. He is a member of Information Processing Society of Japan (IPSJ), Institute of Electronics, Information and Communication Engineers (IEICE), and Japanese Society for Information and Systems in Education (JSiSE).

Tadanori Mizuno received the B.E. degree in industrial engineering from the Nagoya Institute of Technology in 1968 and received the Ph.D. degree in computer science from Kyushu University, Japan, in 1987. In 1968, he joined Mitsubishi Electric Corp. Since 1993, he is a Professor of Shizuoka University, Japan. Now, he is a Professor of graduate school of Science and technology of Shizuoka University. His research interests include mobile computing, distributed computing, computer networks, broadcast communication and computing, and protocol engineering. He is a member of Information Processing Society of Japan, the institute of electronics, information and Communication Engineers, the IEEE Computer Society, ACM and Informatics Society.

Stepwise Approach to Design of Real-Time Systems based UML/OCL with Formal Verification

Takeshi NAGAOKA†, Eigo NAGAI†, Kozo OKANO†, and Shinji KUSUMOTO†

†Graduate School of Information Science and Technology, Osaka University
Yamadagaoka 1–5, Suita City, Osaka, 565–0871
{t-nagaok, e-nagai, okano, kusumoto}@ist.osaka-u.ac.jp

Abstract - This paper provides a stepwise method for the design of real-time systems with timeliness QoS guarantees. In the proposed method, the system components are designed using UML diagrams and are provided with the timeliness QoS annotated with OCL. The basis of this technique is to formally ensure that the required timeliness QoS is satisfied under the provided timeliness QoS, given the network property and the UML diagrams. In order to avoid the state-explosion problem during performance model checking, which can logically check the satisfiability, the problem is separated into two steps. The first step checks the satisfiability using an abstract model of each of the components derived automatically from the provided QoS. The second step independently performs model checking for each of the components using a more detailed version of the behavioral model of a given component. Such an approach reduces the number of total states to check. Furthermore, the approach can be extended into hierarchical design, which leads to good scalability. Experimental results are also included in this paper.

Keywords: timeliness QoS, UML/OCL, model checking, component based systems, hierarchical design

1 Introduction

This paper presents a new method to verify consistency of timeliness QoS of component-based designed real-time systems. We assume that timeliness QoS is not only given to a whole system (Required QoS) but also associated with each component of a given system (Provided QoS).

Timeliness QoS is a time aspect of QoS (Quality of Service) features[1]. In the paper, we treat jitter, latency and throughput as timeliness QoS.

Nowadays, most real-time systems are designed with help of UML diagrams[7]. Especially components and their relation through signal communication can be represented in a class diagram of UML. In UML based design, such timeliness QoS can be annotated in OCL[8]. The annotation is associated to each of components as a provided QoS and also to a network link as a network property. Recently, SysML (System Model Language)[13] also attracts interest. SysML extends from UML and presents mixed systems consisting of physical devices and software and network systems. Therefore, SysML supports diagrams for signal flows and physical flows. For behavioral diagrams, SysML supports four diagrams as same as UML. Among them, state diagram has powerful representation including parallel and hierarchical states.

The proposed method is revised version of paper [15]. The method in [15] uses Linear Programming (LP) for some of verification. The approach has a disadvantage that connection among components has to be acyclic, and it cannot be applied to hierarchical design. The method of this paper uses abstract QoS automata instead of using LP; thus it improves the former disadvantage.

The heart of the technique is formally to ensure that the required timeliness QoS is satisfied under the provided timeliness QoS, given network property and the class diagram.

In order to check the satisfiability, there are several approaches. Formal approaches are very useful. Model checking is one of such an approaches. Notion of Test Automata[3], [5] and its application is also useful. However, one of disadvantage of the method is the state-explosion problem.

In order to avoid state-explosion problem while performing model checking, we separate the problem into two steps. The first step checks the satisfiability using abstract model of each of components derived automatically from the provided QoS. The second step performs model checking each of components independently using more detailed version of behavioral model of a component. Such an approach efficiently reduces the number of total states to check. Moreover the approach can be extended into hierarchical design; therefore it has good scalability.

SaveCCM[14] is a technique for Component based Development (CBD). In a description of a component, it allows user to define ports where signals input or output and to represent behavior in a timed automaton[2]. An IDE over Eclipse is available. Therefore, our proposed method has an affinity for SaveCCM.

The paper organized as follow. Section 2 gives timeliness QoS. Section 3 shows how to design component based realtimed systems in UML/OCL. Section 4 demonstrates the proposed method which consists of two steps. Section 5 provides an experimental example. We conclude the paper in Section 6.

2 Timeliness QoS

Main building blocks of our model are components. Each component has one or more *interfaces* to the environment, where all interactions between components is conducted via the interfaces. Since, we are mostly dealing with real-time systems and timeliness QoS, we shall assume that the interaction of a component with its environment is carried out via

ISSN1883-4566/09

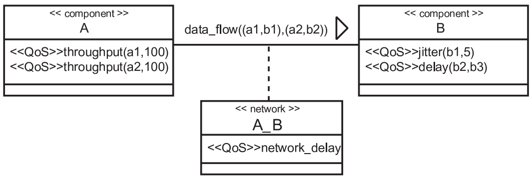


Figure 1: A Configuration of Components in UML Class Diagram

input and *output* signals. As a result, interfaces of a component specify signals that the component receives or emits.

Each component is associated with a number of input and output signals. In this paper, signals are denoted by x, y and z. Time of occurrence of a signal is denoted via a non-negative sequence of rational numbers. For example, the time of occurrence of a signal x is denoted with $x_1, x_2, \ldots$ representing time of first, second, ... occurrence of x.

Timeliness QoS expressions [9] such as jitter, throughput and latency can be expressed via first-order logic formulas on the set of time of occurrence of signals.

Throughput of at least (most) K within the time period T, for signal x can be written as the first order formula $\forall i \in \mathbb{N}: \quad x_{i+K-1} - x_i \leq T$ ($\forall i \in \mathbb{N}: \quad x_{i+K-1} - x_i \geq T$), respectively.

Notice, paper [5] refers to the above QoS constraint as Non-Anchored throughput.

Jitter , also called Non-Anchored jitter , of a signal x can be defined by the expression $\forall i \in \mathbb{N}: \quad T - m \leq x_{i+1} - x_i \leq T + M,$ where T is the period of the jitter and m, M are constant rational numbers.

Latency of at most T unit of time between two signals x and y as $\forall i \in \mathbb{N}: \quad 0 < x_i - y_{Ki+K'} \leq T$. A special case of the above definition (for $K = 1$ and $K' = 0$) is the well-known definition of latency $\forall i \in \mathbb{N}: \quad 0 < x_i - y_i \leq T$ that applies to the time difference of the i-th occurrence of x and y.

3 UML/OCL based design of real-time systems

A real-time system can be designed as a set of components where signal communication links exist among pairs of components. We can describe such components in a UML class diagram in Fig. 1.

Type Component can be specified by Stereotyping "component," by which user can easily extend UML specification. A signal communication can be specified with Association.

Each of components has provided QoS, which can be represented via OCL annotation. Each of network links (which has association class with stereotype "network") also has network properties represented via OCL annotation. The network properties is the same as timeliness QoS. Attribute region of each class, includes special variables for QoS with "QoS" stereotype (in Fig. 1).

The following is the syntax of the variables.

Throughput Variable := "throughput(" signal "," period ")" ;
Jitter Variable := "jitter(" signal "," period ")" ;
Latency Variable := "delay(" output "," input ")" ;

A class with "component" stereotype has three categories of timeliness QoS (Jitter, throughput and latency), while a class with "network" has two categories of timeliness QoS (Jitter and latency).

The OCL description is given as follows[9].

QoS description := "context" className invariant* ;
invariant := "inv: self." constraint ;
constraint := variable op constant;
variable := Throughput Variable | Jitter Variable | Latency Variable
op := "$>$" | "$<$" | "$\geq$" | "$\leq$" ;

For example, the following are examples for Fig. 1, where – means a comment line.

context A
inv: self.throughput(a1,100) ≥ 20
– signal a1 is emitted at least 20 times in the period 100 units of time
inv: self.throughput(a2,100) ≤ 10
– signal a2 is emitted at most 10 times in the period 100 units of time

context B
inv: self.jitter(b1, 5) < 1
– signal b1 has jitter 1 with period 5 units of time
inv: self.delay(b2,b3) < 5
– latency between receiving signal b2 and sending signal b3 is less than 5 units of times
context A_B
inv: delay ≤ 100
– latency (network delay) between component A and component B is less than 100 units of time

4 The Verification Method

The verification consists of two steps; First Step and Second Step. If some of the components are not simple enough, then repeat the process again from First Step on each of the components. The following is the abstract level of steps of the proposed method.

input:

- system required timeliness QoS represented in OCL;
- component level provided timeliness QoS represented in OCL; and
- network configuration represented in UML/OCL class diagram.

output:

- component level behavioral specification represented in UML/OCL state-chart which satisfies required timeliness QoS under the configuration;

- or failure.

1. First step
   (a) We generate test automaton from the required timeliness QoS.
   (b) We generate abstract QoS automaton from each of the provided timeliness QoS.
   (c) We generate configuration automaton from the network configuration.
   (d) We check the consistency from parallel composition of the above automata.
   (e) If the result is deadlock then return failure. We have to reconfigure the requirement or provided conditions.
   (f) If the result is not deadlock then go to Second Step.
2. Second Step
   (a) If the component is not small enough to represent simple state-chart, then refine the component by;
      i. renaming provided QoS of the component to required QoS;
      ii. design sub components and provided QoS of each of them;
      iii. design network configuration; and
      iv. we repeat the First Step until the component is enough to small.
   (b) If the component is small enough to represent simple state-chart, then we describe state chart of the component.
   (c) We translate a test automaton from the provided timeliness QoS.
   (d) We design network of timed automata from the state-chart.
   (e) We check the consistency from parallel composition of the above automata.
   (f) If the result is deadlock then return failure. We have to reconfigure state-chart.
   (g) If the result is not deadlock then return success.

4.1 The First Step

Verification inputs are the following.

- Required QoS;
- a set of components with provided QoS; and
- a configuration automaton which represents network properties.

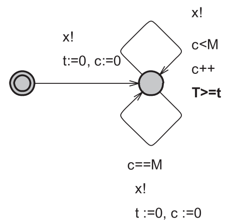


Figure 2: Abstract QoS automaton for Anchored Throughput

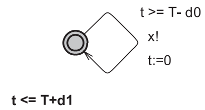


Figure 3: Abstract QoS automaton for Non-Anchored Jitter

The output is whether a given required QoS is satisfied under a given set of components with provided QoS and a given set of network links with network properties.

In usual methods, designer models behavior of each component in a network of timed automata and for the whole network of timed automata. Then the designer performs model checking, which often results in state explosion.

Here, we give a new method, in which timed automata (We call each of them an **abstract QoS automaton**) is derived automatically from the provided QoS. The important point is that derived automata are so small that state-explosion is avoided. Here, we give a translate rule for each provided QoS.

4.1.1 Throughput

A translated abstract QoS automaton for throughput is shown in Fig. 2. The automaton transmits signal x at least M times during the period T. The variable c and clock variable t are used for such the control. When "throughput must be at least k frames per P ms" is given as a provided QoS, the corresponding abstract QoS automaton is generated with substitution $M = k$ and $T = P$.

4.1.2 Jitter

A translated abstract QoS automaton for jitter is also shown in Fig. 3. The automaton transmits signal x with the priod T. The allowed jitter is $[T-d_0, T+d_1]$. Using the clock variable t, it transmits signal x at every $[T - d_0, T + d_1]$ period.

When "jitter must be $[-d_0', d_1']$ with a period T'" is given as a provided QoS, the corresponding abstract QoS automaton is generated from Fig. 3 with substitution $T = T'$, $d_0 = d_0'$ and $d_1 = d_1'$.

4.1.3 Latency

For signal x and y, let m and M be the minimum and maximum latency, respectively. A translated abstract QoS automaton for latency with above parameter is shown in Fig. 4. The

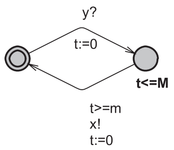


Figure 4: Abstract QoS automaton for Latency

automaton transmits signal x after receiving signal y with the latency $[m, M]$.

Unfortunately, the automaton does not accept input y until it emits output x. To avoid the problem, a set of the same automata is needed. The number of automaton decided from throughput property of the components.

When "latency for signal x and y must be $[m, M]$" is given as a provided QoS, and also parameter T representing period of signal y is given, the corresponding abstract QoS automata are generated from Fig. 4. The number of copies is T.

4.1.4 Configuration Automaton

A configuration automaton models interfaces among components. As each component has several inputs and outputs, such an I/O is represented as a channel in the configuration automaton. Each channel synchronizes with some I/O of some components with provided QoS (abstract QoS automaton). Abstract QoS automata and the configuration automaton communicate each other as described above.

4.1.5 Test Automaton

For a given required QoS, we can verify whether the required QoS is satisfied with the system by generating a corresponding test automaton from the required QoS. Fig. 5, 6, and 7 show templates of test automata for throughput, jitter, and latency, respectively. For such template, substituting each parameter with a concrete value specified by the required QoS, we can obtain a test automaton.

Throughput For a non-anchored throughput of which a signal e occurs at least k times in a period T and at most k times in a period $T0$, a network of test automata consisting of k processes of timed automata in Fig. 5 observes the throughput.

The test automaton observes if $T0 \leq T(e, i+k) - T(e, i) \leq T$ holds for some i, where $T(e, i)$ means the time of i-th occurrence of signal e. With k copies of such test automata, they can observe if $\forall i(T0 \leq T(e, i+k) - T(e, i) \leq T)$ holds. In other words, thay can observes at least k times signal e occures during $[T0, T]$. Parameters of the test automaton are k, T and $T0$.

In the network of test timed automata, the variables c is shared among automata globally. Each of timed automata is activated by turns along the value of variable K. When there exists common divisor k for $T, T0$ and n, we can reduce the number of copies of the test automaton to k/n with the parameters T/n, $T0/n$ and k/n. The discussion of such reduction technique is described in Sec. 5.4.

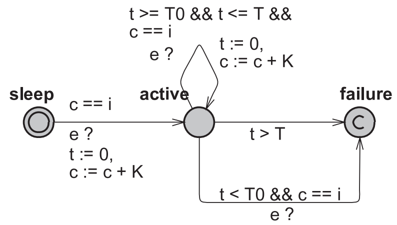


Figure 5: Test Automaton for throughput

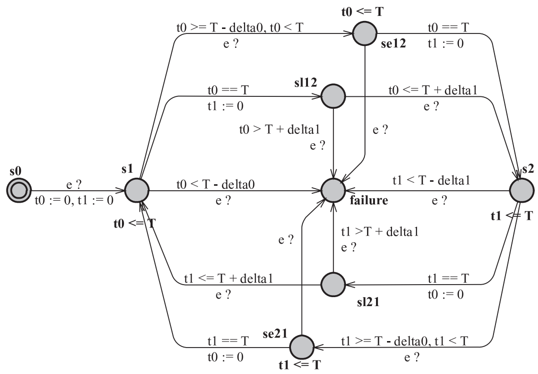


Figure 6: Test Automaton for jitter

Jitter Figure 6 shows a test automaton for anchored jitter. It observes whether a signal e occurs periodically in the period $[nT - delta_0, nT + delta_1]$, where $n = 1, 2, 3, \ldots$.

The automaton in Fig.6 has two clocks $t0$ and $t1$. A path form $s1$ to $s2$ via $se12$ or $sl12$ observes that the time of j-th occurrence of signal e is during the period $[jT - delta0, jT + delta1]$ using clock $t0$, while a path form $s2$ to $s1$ via $se21$ or $sl21$ observes that the time of $j+1$-th occurrence of signal e is during the period $[(j+1)T - delta0, (j+1)T + delta1]$ using clock $t1$.

Latency Figure 7 provides a component of test automata for latency between a signal x and y. The test automaton shown in Fig.7 observes if $\forall i(T(y, i) - T(x, i) \leq T)$ holds, where $T(e, i)$ means the time of i-th occurrence of signal e.

We have to use T/D copies of such test automata, where D is period of signal x. Variable cx and cy are shared variables with them, which serve to count the occurrence of signal x and y.

4.1.6 Verification

The behavior of such media with timeliness property also is modeled in a network of timed automata, we call such an automaton a configuration automaton. Parallel composition of

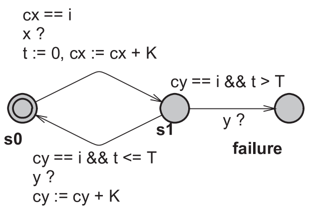


Figure 7: Test Automaton for latency

an abstract automaton for every component and the configuration automaton and test automaton for specified timeliness QoS decides whether the whole system satisfies the specified timeliness QoS. For the detail of process of the verification, refer [10].

4.1.7 Category Based Model Checking

Verification is performed for every timeliness QoS category (latency jitter and throughput). The idea and approach is very simple. When we want to check only latency as the required QoS, we build an abstract automaton for provided QoS of latency only. The divided and conquer approach, reduces the size of states.

4.2 The Second Step

For each of components, Second step has the following two cases depending on the component's abstraction

- We repeat First step to the given component recursively.
- We design detail behavior of the component and verify whether provided QoS is ensured by the design.

If the size of the given component is large and designer has to design the given component from more detail components, then repeats First step. Hereafter, we describe the later case.

At Second step (of the later case), verification is independently performed for each component. Before Second step, the designer has to give detailed behavior of each component. Such behavior is given in UML state-chart. In order to give time constraints on events, the state-chart has clocks.

Verification inputs are the following.

- component's behavior given in UML state-chart with clocks; and
- component's provided QoS.

The output is whether provided QoS is satisfied under a given UML state-chart with clocks.

The verification is performed based on test automaton. We have to translate a UML state-chart with clocks to a network of timed automata.

A state-chart can represent hierarchical architectures; while a network of timed automata is a simple flat structure model.

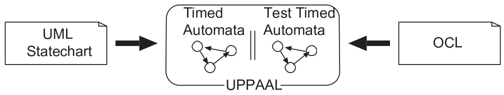


Figure 8: Verification on UPPAAL based on Test Automata

In general, hierarchical structure can be flatten, but such translation increases the number of states. There are several translations, and this paper adapts the one in [4]. The translation itself is an algorithm to translate Hierarchical Timed Automata (HTA) to a network of timed automata used by UPPAAL[11].

Thus, we have to translate state-charts to HTA. Fortunately, syntax and semantics of state-chart and HTA are both similar, the translation is simple.

We add the following constraints on the state-chart.

- the state-char diagram has clocks; and
- arcs in the stat-chart has clock constraints in a form of the one same as Timed automata in UPPAAL.

We also use test automata to check timeliness QoS. Test automata for jitter, latency and throughput are given in 4.1.5.

The verification can be performed with UPPAAL. Thanks to test automata, we just check deadlock property for each QoS. Logical expression for deadlock property in UPPAAL is "A[] not deadlock."

5 Experiment

The proposed method is applied to an example.

5.1 The example

Media Server is an application delivering video stream and audio stream to Digital Television and Audio System[6], [12]. Each of output devices required timeliness QoS (throughput). Figure 9 shows the class diagram of the application, which consists of twelve components.

In order to compare the proposed method to the old method, which uses LP solver to First Step, we merge the twelve components to three components (Server 3 components , Audio client 4 components, and Video clients 5 components).

5.2 First Step

The following is the provided QoS.

- Throughput of Component MS–Server is equal or greater than 100 frames/s.
- Processing latency of Component MS-Storage is equal or less than 5ms.
- Network latency between MS–Server and Digital–TV is equal or less than 100ms.
- Network latency between MS–Server and Audio–System is equal or less than 150ms.

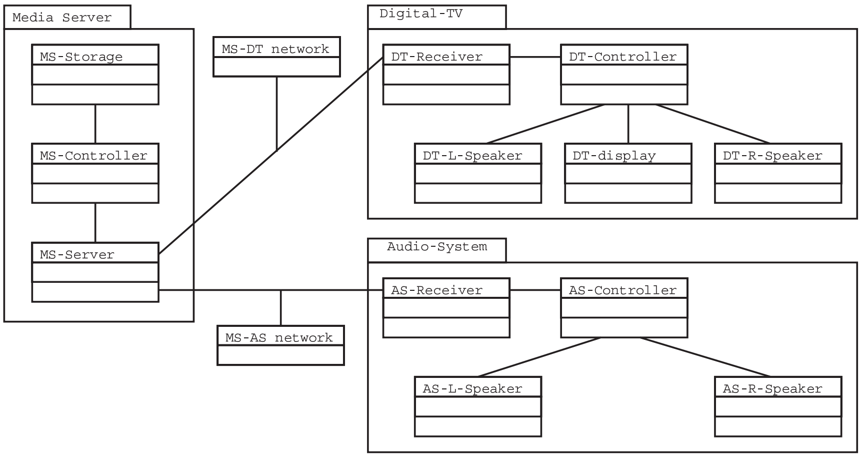


Figure 9: Class Diagram of Media Server

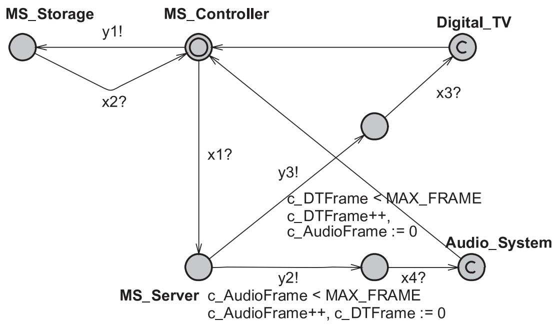


Figure 10: The configuration automaton

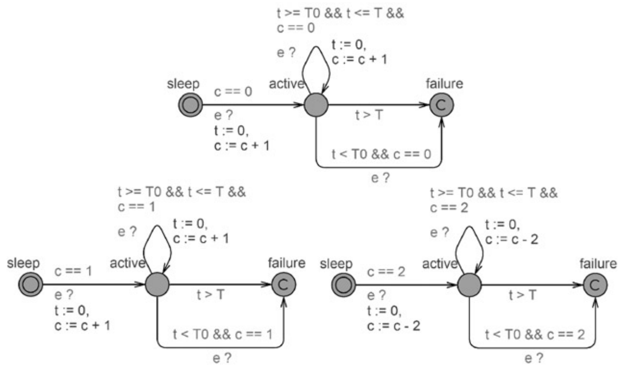


Figure 11: The network of test automata for the given required QoS

We give the following requirement for the Required QoS for the system.

- Throughput of Digital Display (DT Display) must be at least 30 frames/sec.

For these provided QoS, and a configuration automaton derived from the UML class diagram, and Required QoS for the whole system, we apply the verification along with First Step.

Figure 10 shows the Configuration automaton for the experiment. The Configuration automaton in Fig.10 represents connection among the components. It uses channels to communicate abstract QoS automata providing the provided QoS mentioned above. For example channel x is used for communication to an abstract QoS automaton with throughput 100 frames/sec at a transition between MS–Controller and MS–Server. In order to avoid unfairness that frame communication occurs only between MS–Server and Digital–TV (or only between MS–Server and Audio–System), we use a parameter MAX_FRAME, which is used in a condition that the maximum successive occurrence of signals between the same components. We use a condition $MAX\_FRAME = 1$ for the experiment.

Figure 11 shows the test automaton for throughput as Required QoS. The test automaton observes throughput of 3 frames per 100ms. As required QoS, the required value of throughput is 30 frame/sec, We have to need 30 processes of throughput test automata to observe the throughput exactly.

At the First Step, we have performed verification experiments for two configurations: (1) an abstract QoS which outputs ten frames per 100 msec, and (2) an abstract QoS which outputs 30 frames per 300 msec, respectively, are used for abstraction of provided QoS for MS.Server.

The provided QoS of MS.Server is at least 100 frames/sec. To prevent the jitter of frame signals, we adopt 100ms and 300ms as the period in the experiments. The values are set in the configuration (1) and (2).

Each of two experiments is performed with several numbers of test automata: 3, 6, 9, 12, 15 and 30. We have obtained CPU times and sizes of memory consumed. The experiments are performed in the following environment: CPU is Intel Core 2 Duo 2.33GHz, OS is Windows Vista Business and M.M. is 2GB. We used UPPAAL4.1.0 as a model

Table 1: The result (1) of first step

# of P	result	CPU time	Used memories
3	not valid	0.3 ms	23.9MB
6	not valid	1.4 ms	24.4MB
9	valid	28 ms	24.8MB
12	valid	50.6 ms	25.9MB
15	valid	83.6 ms	26.8MB
30	valid	480 ms	34.4MB

Table 2: The result (2) of first step

# of P	result	CPU time	Used memories
3	not valid	0.6 ms	23.9MB
6	not valid	0.7 ms	24.4MB
9	not valid	1.2 ms	24.7MB
12	not valid	1.6 ms	25.0MB
15	not valid	2.1 ms	25.1MB
30	valid	957 ms	40.4MB

checker. Table 1 and Table 2 show the results of (1) and (2), respectively. The column of "# of P" shows the number of processes (the number of test automata).

In the previous experiment, we have performed First Step with Linear Programming solver. In the experiment, we have performed it in 78 ms (although it has been performed in different environment).

5.3 The Second Step

After First Step, we design inner behavior of each component.

In the example, recursive application of First Step is not performed, because each component is small enough. Behavioral specification is described in UML state-chart. The design must meet the provided QoS. Figure 12 shows behavioral specification of Component MS-Storage.

In order to verify timeliness QoS for each component, Statechart must to be translated into a network of timed automata and also timeliness QoS is converted into test automata.

Figure 13 depicts translated result of *on* part in Fig.12.

The translating times are summarized as follows.

Translation time	: 1153 ms
The number of states(before)	: 89
The number of states(after)	: 179

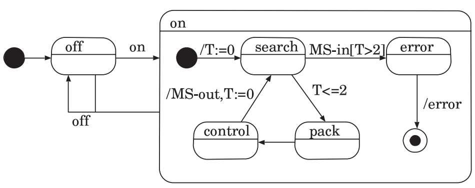


Figure 12: A UML Statechart Diagram of Component MS-Storage

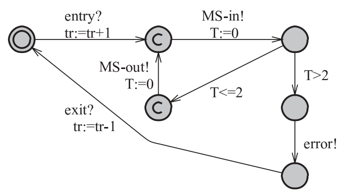


Figure 13: An UPPAAL Timed Automaton of Component MS-Storage

For every component and for every timeliness QoS, verification is performed. The total CPU time is about one seconds.

We found that for every component, the verification is performed within a few seconds with UPPAAL, without state explosion. Also we found that there is no deadlock.

5.4 Discussion

Table 1 shows that when we perform the experiment with nine test automata, it outputs the correct result. The result of Tab.2, however, shows that we cannot obtain the correct result until the number of test automata increases to 30. When we perform the experiment with 30 test automata, the CPU time increases exponentially. Therefore, we can conclude that there is a trade-off between degree of precision and CPU times. As shown in both tables, the CPU times of the experiments with 30 test automata are too large. Thus, as we consider the trade-off, in this experiment, the trade-off point is at which the number of process is 15.

Though we cannot exhibit that our proposed method is better than that of linear programming based method with respect to the CPU time, the performance of the proposed method is within useful reasonable time. The linear programming method has many constraints on configuration, while the new proposed method is flexible and is able to apply recursively along with component hierarchy, which are the advantage of the proposed method.

Our proposed method has more acceptable inputs than the former method. The difference between this and that of general class is very small. It is although not faster than the former method, it is more flexible than the former method.

6 Conclusions

This paper proposed a stepwise verification method for design of real-time system with UML/OCL focusing on timeliness QoS aspects. The method uses abstract QoS timed automata in order to reduce the possibility of state explosion.

The method can be applied to a design with complex connection of components.

Future works include simultaneous verification of several kinds of timeliness QoS, and utilization of feedback information such as verification counter-examples.

Acknowledgement

This work is being conducted as a part of Stage Project, the Development of Next Generation IT Infrastructure, supported by Ministry of Education, Culture, Sports, Science and Technology.

REFERENCES

[1] R. Staehli, F. Eliassen, J. Aagedal and G. Blair "Quality of Service Semantics for Component-Based Systems," 2nd Int'l Workshop on Reflective and Adaptive Middleware Systems, pp.153-157, 2003

[2] R. Alur and D.L. Dill: "A Theory for Timed Automata," In Theoretical Computer Science 125 pp.183-235, 1994.

[3] L. Ageto, P. Bouyer, A. Burgueño and K. G. Larsen: "The Power of Reachability Testing for Timed Automata," LNCS, Vol.1530, pp.245-256, 1998.

[4] A. David and M. O. Möller: "From HUPPAAL to UPPAAL: Translation from Hierarchical Timed Automata to Flat Timed Automata," BRICS Technical Report Series, RS-01-11, 2001.

[5] H. Bowman, G. Faconti and M. Massink: "Specification and verification of media constraints using UPPAAL," In Proceedings of Design, Specification and Verification of Interactive Systems'98, pp.261-277 Springer, 1998.

[6] K. Havelund, A. Skou, K. G. Larsen and K. Lund: "Formal Modelling and Analysis of an Audio/Video Protocol: An Industrial Case Study Using UPPAAL," In Proceedings of the 18th IEEE Real-Time Systems Symposium, pp.2-13, 1997.

[7] Object Management Group: Unified Modeing Language Specification version 2.1, available at http://www.omg.org/.

[8] B. Bordbar, J. Derrick and A. G. Waters: "A UML approach to the design of open distributed systems," In Chris George and Huaikou Miao, editors, Formal Methods and Software Engineering, LNCS, Vol.2495, pp.561-572, 2002.

[9] UML Profile for Modeling Quality of Service and Fault Tolerance Characteristics and Mechanisms, Request for Proposal, available at `http://www.omg.org`

[10] B. Bordbar and K. Okano, "Verification of Timeliness QoS Properties in Multimedia Systems," 5th International Conference on Formal Engineering Methods (ICFEM '03), LNCS 2885, pp.523-540, 2003

[11] Johan Bengtsson, Kim G. Larsen, Fredrik Larsson, Paul Pettersson, and Wang Yi: "UPPAAL . a Tool Suite for Automatic Verification of Real-Time Systems," LNCS, Vol.1066, pp.232-243, 1995.

[12] D. Akehurst, J. Derrick, and A. G. Waters: "Design and Verification of Distributed Multi-media Systems," LNCS, Vol.2884, pp.276-292, 2003.

[13] The official OMG SysML site, http://www.omgsysml.org/

[14] J. Carlson, J. Håkansson and P. Pettersson: "SaveCCM: An Analysable Component Model for Real-Time Systems," Proc. of FACS'05, Electronic Notes in Theoretical Computer Science, Vol.160, pp.127-140, 2005.

[15] E. Nagai, A. Makidera, K. Okano, K. Taniguchi: "A Method to Develop Distributed Real–Time Applications Based on UML/OCL (in Japanese)," In the IEICE Transactions, Vol.J89-D No.4, pp683-692, 2006.

(Received October 14, 2008)
(Revised July 15, 2009)

Takeshi Nagaoka received the M.I. degree in Computer Science from Osaka University in 2007. He currently belongs in a doctoral course. His research interests include abstraction techniques in model checking, especially timed automaton.

Eigo Nagai received the BE, and M.I. degree in Information and Computer Sciences from Osaka University, in 2005, and 2007, respectively. During his Master Course, his research interest was real-time application for model checking, especially timed automata. He is currently a researcher in NEC.

Kozo Okano received the BE, ME, and Ph.D degrees in Information and Computer Sciences from Osaka University, in 1990, 1992, and 1995, respectively. Since 2002 he has been an associate professor in the Graduate School of Information Science and Technology, Osaka University. In 2002, he was a visiting researcher of the Department of Computer Science, University of Kent at Canterbury. In 2003, he was a visiting lecturer at the School of Computer Science, University of Birmingham. His current research interests include formal methods for software and information system design. He is a member of IEEE, IEICE of Japan and IPS of Japan.

Shinji Kusumoto received the BE, ME, and DE degrees in information and computer sciences from Osaka University in 1988, 1990, and 1993, respectively. He is currently a professor in the Graduate School of Information Science and Technology at Osaka University. His research interests include software metrics and software quality assurance technique. He is a member of the IEEE, the IEEE Computer Society, IPSJ, IEICE, and JFPUG.

A Dynamic Control Scheme of Context Information based on Multi-agent

Hideyuki Takahashi[†], Yoshihisa Sato[†,‡], Takuo Suganuma[†,‡], and Norio Shiratori[†,‡]

[†]Research Institute of Electrical Communication, Tohoku University, Japan
[‡]Graduate School of Information Sciences, Tohoku University, Japan
{hideyuki, yosi-q, suganuma, norio}@shiratori.riec.tohoku.ac.jp

Abstract - In ubiquitous computing environments, an effective handling of Quality of Service (QoS) is needed to provide context-aware service in tune with the variation of context information. But QoS for overall system may fall if either the quantity or quality of context information exchanged among entities is insufficient or excess. Based on user context, resource context and QoS, we propose a dynamic control scheme for context information delivery. Based on highly autonomous and cooperative multi-agent system, we compose the entities to configure context-aware services. Our proposed scheme can efficiently adapt to the various changes in the real-world environment and maintain the QoS to the user satisfaction. The effectiveness of our proposed scheme is evaluated from the perspective of adaptability depending on the varying relationship between real-world and ubiquitous computing environment.

Keywords: Ubiquitous computing, Context information management, Quality of Service, Quality of Context, Multi-agent system

1 INTRODUCTION

In recent years, ubiquitous computing (ubicomp) [1] environment is emerging, where various kinds of sensors, handheld terminals, and wireless networks cooperatively work to support daily lives of the people. As a distinctive service (e.g., wellness management service, telemedicine support service, and mobile information service) provided by this environment, context-aware services come to the front [2]–[4]. The context-aware services are the services in ubicomp environment based on the “context” which is the situation of each entity. Here, the entity is the element in ubicomp environment such as users, hardware devices, software, and networks.

The context of an entity is delivered in the form of information in the network, and used by other entities. We call this information as “Context Information” (CI). The CI is exchanged at very frequent intervals, then the entities should transfer and handle massive amounts of CI, as well as handling data for main services, using the shared resources. Especially in case of application that consumes a large amount of computational and network resources, such as multimedia communication service, the resources for service provisioning itself decrease due to excessive circulation and processing of CI. Therefore a critical degradation in the QoS in overall system may result.

To solve this problem, Tokairin et al. [10] proposed a Context Management Scheme to keep the QoS. Their scheme is based on the concept of quality of context (QoC), and increases QoS in ubicomp environment by managing CI delivery effectively. However, even if the types of entities are same, the behavior of each entity is different depending on the physical environments where they exist in the real world situation. Hence, the autonomous and adaptive control scheme of CI delivery is needed, according to the varied physical environment in the real world.

Our goal is to develop an effective CI managing scheme for context-aware services to provide appropriate QoS according to varied physical environment and user’s requirements. This paper presents a dynamic control scheme of CI delivery based on multi-agent. To realize this we compose entities which configure context-aware services as highly autonomous and cooperative agents in this scheme. By employing the proposed scheme, the system can provide the context-aware services in varied physical environment flexibly by using the situational adaptability of each agent and their organizational behaviors.

Especially, we performed an initial experiment with the prototype system of a ubiquitous live streaming video service with our proposed scheme. From the results, we confirmed that the system can control the CI delivery according to provided QoS and the change of location and speed of a user entity in the physical environment. We also confirmed the effectiveness of our scheme in perspective of user-level QoS of the video streaming such as frame-rate and timeliness of service provisioning, according to the changes of entities in the real-world environment.

2 RELATED WORK AND PROBLEMS

2.1 Related Work

In this section, we present related works and summarize their problems. Some studies have explored concentrating on the acquisition and selection of context information (CI) for context-aware service provision [5], [6]. CHANSE [5] aims for easy configuration of context-aware service by using centralized management server of CI. In case of failure to get requested CI for configuration of service due to break down of a sensor device, the system can keep CI provisioning by selecting alternative CI from other available sensors. This mechanism gives a good solution to increase availability of context-aware service; however, it is required to describe static process of electing CI when a device is newly introduced. The dynamic reconfiguration of sensor devices is also expected

ISSN1883-4566/09

when the device providing CI breaks down. ContextDistillery [6] proposes a framework that aims for abstraction of the up-to-dateness's variety of each CI. However, it is required to describe the process of selecting CI and up-to-dateness of CI statically in its design phase. The dynamic provision of CI to adapt to the operational situation of the system is expected in the real world.

Additionally, some studies have investigated focusing on Quality of Context (QoC) [7]–[9]. Buchholz et al. advocated the notion of QoC [7]. They introduced and defined from "precision", "probability of correctness", "trust-worthiness", "resolution", and "up-to-dateness" for QoC parameters. They also refer to relation of QoC, QoS, and Quality of Device (QoD). Sheikh et al. [8] define QoC aiming to deal with complicated specification of CI effectively in the middleware for context-aware service. They defined QoC parameters for more practical use than [7], they are, "Temporal Resolution", "Spatial Resolution", and "Probability of Correctness". They mainly discuss the signification of usage of QoC parameter, however, it is not clear how the QoC may be used in system operation for real world applications. An adaptive middleware framework [9] is proposed to provide and select CI depending on the application. They define QoC by "Precision" and "Refresh Rate". This framework computes Utility Function based on the QoC; CI are selected according to the computed amount. This is the pioneering work to use QoC in the real system. However it is difficult to control "Precision" and "Refresh Rate" dynamically because these QoC parameters are assumed to be preliminarily defined and advised.

On the other hand, a flexible QoC control scheme during the system operation is proposed [10]; this scheme aims to provide the advanced QoS control ability. They propose context information management based on multi-agent system; while they focus on "up-to-dateness" as a QoC parameter. This scheme enables the control of CI delivery based on resource status and user's physical location. It is possible to manage QoS of overall system. However, because applicable real environment and services are restricted by the static value of the threshold of resource status and the fixed area of the user's location, which are specified in the design phase of the system, thier scheme has some limitations in scalability and flexibility of QoS control ability.

2.2 Problems

We tackle the control of the context information (CI) delivery. Here, we point out two current technical problems. We assume that ubicomp environment comprises physical environment (real space) and ubiquitous computing space (ubicomp space). We describe the problems of adaptability from viewpoints of relationship between real space and ubicomp space.

- **Effective CI delivery based on the run-time behavior of entities (P1)**: Suppose the cases where context-aware services are provided by using computational and network resources shared by main services and CI delivery in ubicomp space. The resources required to provide its main service is degraded due to the resource limitations of ubicomp space when the amount of CI exchanged among entities becomes huge. Consequently, the main service doesn't work properly or the QoS may be greatly decreased. The effective CI delivery in ubicomp space, deeply considering the run-time behavior of entities in real space, is essential.

- **Flexible CI delivery adapted to change of real space in long-term range (P2)**: We described the problem about reactive adaptability of CI delivery control in run-time in (P1); however, we must deal with the changes of real space in more long-term range. These changes may occur with reorganization of furniture in a room, replacement of the role of a room, modification of work flow, etc. We also need to take the deployment problem of a ubicomp space to a real space into account. It is necessary to customize and adjust the ubicomp space delicately in order to make it workable on the target real space because of diversity of real space. Therefore, it is required to adapt the QoC control scheme autonomously during the service provisioning when a ubicomp space is deployed on an arbitrary real space, and even when the real space changes in long-term range. Then, it is possible for the system to improve its QoC control ability by itself gradually over time.

3 DYNAMIC CONTROL SCHEME OF CONTEXT INFORMATION BASED ON MULTI-AGENT

3.1 Relation of QoS and QoC

A general model of context-aware service provision system is shown in Figure 1. The real space comprises four kinds of entities: hardware entities such as PC, PDA, and RFID Tag, software entities such as video transmission/receiving system working on PC, network entities such as wired/wireless networks that connect the entities, and user entities as human user of the system. Here "Context Information" (CI) is defined as the information which carries the situations of these entities.

The entities provide a service for user entities in cooperation with other entities. If the system controls a service based on a specific CI, we call this service "context-aware service." First, the system collects CI from each entity in real space, as shown in Figure 1. "CI Delivery and Handling System" processes the CI and passes only the necessary CI to "Service Provision System" that provides the main service. Service Provision System organizes the entity group to provide the service based on the CI, and the system starts the service for the user entity.

Both the systems in ubicomp space sometimes use the same resources; where resources mean computational resources and network resources. In this situation, if enough amounts of resources are not available, the conflict on the resources occurs

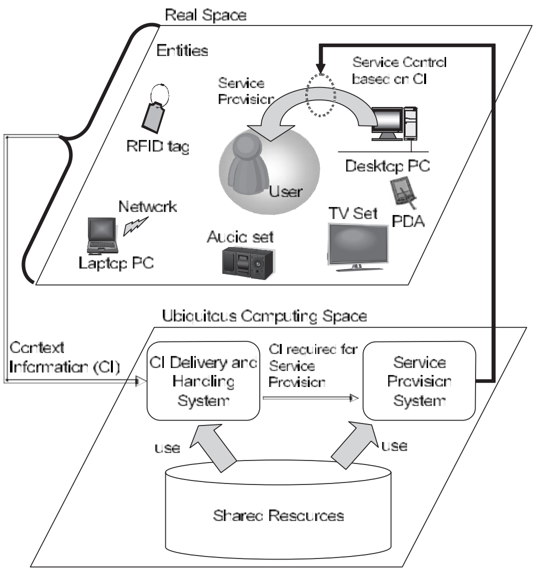


Figure 1: A model of context-aware service provision system in ubiquitous computing environment

and then we cannot get enough performance as described in Section 2.2(P1). Therefore, we need to tune the system to improve the provided QoS as much as possible.

We define QoS and QoC in ubicomp environment as measurement metrics to tune this system effectively. QoS represents the quality of service that ubicomp space provides to the users. We define QoS as following three parameters;

$$QoS =< Service{-}Quality, Timeliness, In{-}Placeness >$$

$Service - Quality$ means a quality of providing service itself. $Timeliness$ is an indicator that shows how the service starts in right timing from temporal viewpoint. $In - Placeness$ is a measure that shows how the service is provided in appropriate place from special viewpoint. Second and third items are essential in ubicomp environment, and it is important to provide service that satisfies user requirement to these things.

Additionally, QoC is defined based on [7];

$$QoC =< precision, correctness, trust{-}worthiness, resolution, up{-}to{-}dateness >$$

Here, QoC means the quality of CI exchanged among entities and has tight relationship with QoS. A large amount of resources are consumed when CI is delivered and processed with high QoC. Then the resource that is used by the main service is reduced, and the value of each parameter of QoS decreases in the end; concretely speaking, $Service{-}Quality$ decreases. On the other hands, when QoC is decreased, the context awareness is also degraded. It causes troubles such as service delay in establishment (low $Timeliness$), unexpected starting of service in the place where the users do not exist (low $In{-}placeness$), and so on in these cases. These are fatal errors in context-aware services.

We focus on the relationship between QoS and QoC in this paper. We ensure QoS-aware context service provision by controlling QoC based on the condition of entities in real space and provided QoS, and realize better QoS as much as possible.

3.2 Overview of the Dynamic Control Scheme of Context Information Delivery

This paper describes a "Dynamic Control Scheme of Context Information Delivery". This scheme needs to realizes ubicomp space that can provide effective context-aware services by adapting to various changes in real space. By employing the proposed scheme, the system can improve QoS by managing QoC based on the relationship between QoC and QoS described in Section 3.1. This scheme has the following two functions.

- **Dynamic QoC Control Function based on the behavior of entities in real space (F1)**: This function follows the short-term behavior of entities on the second time scale and automatically adjusts QoC to provide higher QoS and keep it as stable as possible. For example, in the case of ever-changing of the location CI of user entity and hardware entities caused by the user's movement in a room, this function avoids to decrease QoS by increasing/decreasing the QoC of the location CI according to the user's location and movement. This function solves (P1) described in Section 2.2.
- **Adaptation Function for long-term changes in real space (F2)**: This function tunes working parameters of (F1) according to the changing situation of entity for long-term (like in a matter of days) in real space. For example, this function can make the adjustment of algorithm of (F1) adaptively in case the trend of entity's working conditions are changed by the rearrangement of furniture and IT devices inside the room. This can solve (P2) of Section 2.2.

In addition, there needs to be another adjustment for further long-term changes in real space, like the change of number of entities or their quality, occurred by replacement of entity members. In this paper, we omit this type of adjustment.

Compared with existing studies, Tokairin et al. [10] have investigated the QoC adjustment based on user location considering trade-off between QoS and QoC. They suggest a method to vary CI along with the area where the user exists. The areas are determined in advance, and they assign value of QoC per area statically. This gives a partial function of (F1). However, this method lacks of flexibility and extensibility against the changes of real space. Our proposed scheme realizes (F1) fully to apply various real spaces easily. Moreover our scheme can overcome existing method in terms of the adaptability by introducing (F2) which is difficult to realize in existing works.

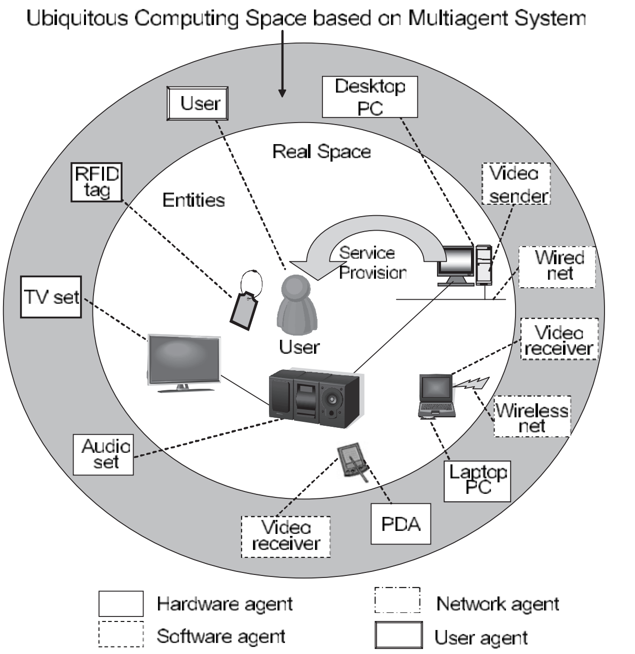


Figure 2: Ubiquitous computing space based on multi-agent

3.3 Architecture based on Multi-agent

The problems (P1) and (P2) in Section 2.2 originate from the limitations of autonomy and cooperativeness of each entity, and the lack of infrastructure for a flexible system construction to support context-aware services. We therefore compose these entities as highly autonomous and cooperative agents, and construct the ubicomp space as a multi-agent system. Figure 2 shows the Ubiquitous computing space based on multi-agent. Each entity is monitored and controlled by the individual agent. Each agent manages CI of each entity and exchanges the CI among the agents by using inter-agent communication protocols. The agent also has knowledge on the specification of the target entity. According to this specification knowledge and monitoring results, the agent can recognize the behavioral situation of the entity. The agents can produce an organization based on the contract among the agents, to configure service organization of entities dynamically [11], [12].

We expect the following advantages by using the concept of multi-agent.

- Effective service composition in ad hoc manner based on the dynamic selection and synthesis of CI by agent's ability of self-awareness and cooperativeness
- Autonomous acquisition of CI reflecting operational situation of entity by the reflection and autonomy ability of agent
- Advanced provision, delivery and distributed management of CI by the agent's cooperation ability
- Improvement of extensibility and flexibility in the system construction by the modularity and the organizational behavior of agent

3.4 Design of Dynamic Control Scheme of CI

(F1) and (F2) are functions to keep QoS high and stable by monitoring the observable CI that affects to QoS of context-aware service, and by adjusting QoC of the operable CI. In particular, we define (F1) as a mapping function that calculates the value of QoC by using appropriate observable parameter of CI as "variable" and pre-defined "coefficients" of each application. We also define (F2) as a function that updates the coefficients based on evaluation after a single service provisioning is finished. This mapping function is defined for every application and is maintained in the agent that has the decision making role.

In this paper, we consider the design of this scheme by using an application example in which the movement of a user is regarded as the change of situation of the real space. This service can be provided at the right place according to the user's location. Here, we regard the location information of the user entity as the observable CI. We also regard the up-to-dateness of location information of user entity as the QoC of operable CI.

In order to trace the movement of a user correctly, the faster the user moves, the higher up-to-dateness of the location information we need. This is because the system must keep the difference small between the user's actual position and the location information. Furthermore, we need to change the up-to-dateness of (QoC) based on not only user's moving speed but also the distance between the user and a service terminal. When the service terminal is far away, we can set low QoC and reduce the consumption of the resource, because the possibility of service provisioning is rather low. We also need high QoC and check the user's location frequently when the user closes to the service terminal, because the possibility of service provisioning increases. By definition of the mapping function that uses the CI as variables, we do not have to set the fixed value manually for mapping between observed CI and QoC. Therefore, we can change QoC continuously based on the changes of real space.

3.5 An Example of Design of Proposed Scheme

We illustrate an example of design of the proposed scheme by using the ubiquitous video streaming service as shown in Figure 3. We consider the following scenario in this service. The user moves in a room while receiving movies with the handheld PC. When the user moves, the system switches the display device and migrates the video player service to a desktop PC that can display the video in higher quality. In this scenario, the desktop PC that can provide the service calculates distance between the user and desktop PC. Moreover it calculates the user's speed from movement distance of the user and an elapsed time from the previous point. The system sends

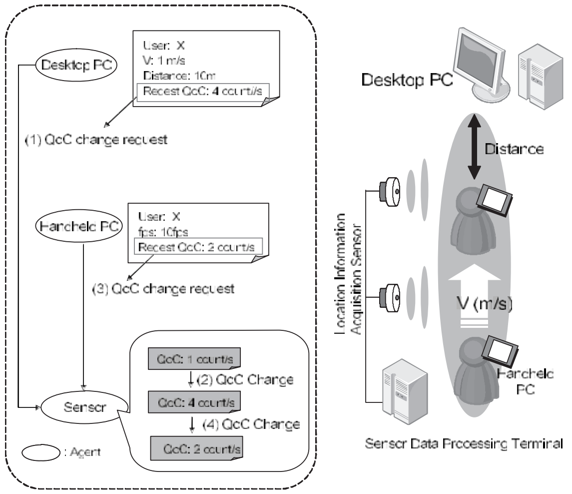


Figure 3: An example of dynamic QoC control function

a QoC change request according to the user's speed and the distance between the user and the desktop PC.

Furthermore, QoC should be controlled considering status of QoS to the user. Suppose that the desktop PC shown in Figure 3, for instance, sets the high QoC, whereas the handheld PC runs out of resources because of the excessive delivery of CI. Thereby it may degrade service quality of video. By monitoring QoS parameters such as frame rate of video, when the actual decrease of QoS occurred, the system requests to change up-to-dateness (QoC) lower. This QoS-based control of QoC can keep the behavior of the entire system stable.

Here is an example of QoC calculation function with a scenario shown in Figure 4. A user moves from point A to point B with handheld PC connected to the Wireless network. The distance between point A and point B are the cover area of the ultrasonic location sensor (ZPS). The service migration occurs when the user get close to the desktop PC at point B. Here, we use the frequency of location information update as the $up{-}to{-}dateness$ parameter of QoC, and frame rate (fps) as the $Service{-}Quality$ parameter of QoS as mentioned in Section 3.1. In this case, we calculate QoC as follow:

$$QoC = \frac{w}{(v+p)\{(X - x') + \alpha\}} \qquad (1)$$

where X is the maximum distance between point A and point B as shown in Figure 4, x' is the distance between the user and the terminal that is the destination of the service migration, v is the speed of the user, p is a constant to consider when the user does not move, and w and α are constants to set initial value of QoC when the user stays at point A. The Adaptive Function (F2) is achieved by adjusting each coefficient value X, w, p, and α.

Moreover, the system changes QoC when the average of the frame rate decreases more than that requested by the user. We calculate and update the new QoC value by following simple expression:

$$QoC = \frac{QoC'}{2} \qquad (2)$$

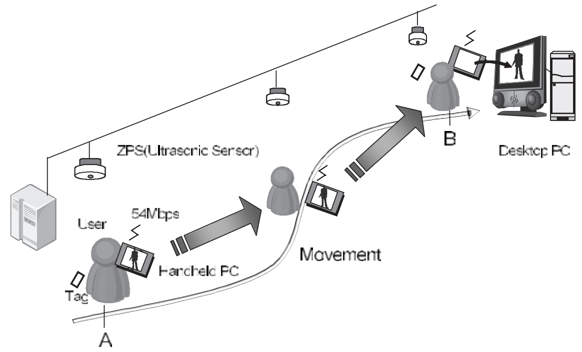


Figure 4: A scenario of QoC control experiment

where QoC' is the QoC before the request is issued.

4 IMPLEMENTATION

4.1 An Application to Live Streaming Service for Ubicomp Environments

We developed an experimental system of live streaming service for ubicomp environments. This is a system that transmits a live video captured by multiple cameras to users at remote locations over the network. This system can play the video on a display device such as TV monitor or display of PC which is located at the nearest position to the users. The system has functions to switch the displays and to migrate the video player service according to the user's position and request of QoS. For example, we expect various application areas like a supervising system for the appearance of the elderly person at home, user's baby in the day-care center, and the indoor pet.

4.2 Design of Experimental System

Figure 5 presents the agent's organization used in this experimental system. The role of each agent is shown as follows:

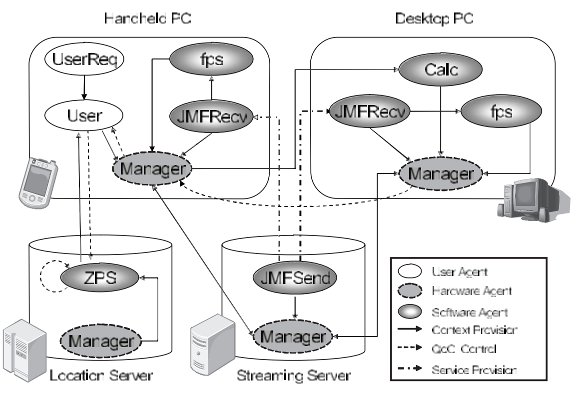


Figure 5: Agent organizations in experimental system

- *User*: This agent manages CI about a user entity. It manages the user's profile, request of QoS, location information, etc.; it sends the user-related information to other agents.
- *UserReq*: This agent controls U/I software entities that have functions to acquire user request. It informs the user request obtained through this U/I to the *User* agent.
- *ZPS*: This agent has function of acquiring and providing the location information of a specific entity with its tag using an ultrasonic positioning sensor system [14]. This agent can control the up-to-dateness of location information dynamically based on requests (QoC change request) from other agents; this agent informs the CI to other agents.
- *Manager*: This agent manages all the agents on each terminal PC; *Manager* can communicate with the other *Manager*. It judges timing and devices for transmitting and receiving the video service to be switched.
- *JMFSend, JMFRecv*: These agents control Java Media Framework (JMF) [15] which is an entity of multimedia communication software component. *JMFSend* agent transmits streaming video and *JMFRecv* agent receives it based on the request of *Manager*. These agents can adjust the quality of video such as encoding quality, data rate, and frame rate.
- *Fps*: This agent monitors the frame rate of streaming video which *JMFRecv* receives. It passes on a warning to *Manager* when it detects requirement violation about the frame rate.
- *Calc*: This agent calculates the distance between the user and the target PC, and the speed of the moving user using the location information; this agent derives value of QoC by using the mapping function described in Section 3.5.

Figure 6 shows the experimental environment. We employed DASH [11] and IDEA [13] for the software infrastructure. DASH is a rule-based multi-agent framework, while IDEA is a development-support environment of the DASH agents. Figure 7 shows a snapshot of a handheld PC and an ultrasonic positioning sensor system (ZPS). We used ZPS as a sensing device for the location information. In this system, a ZPS tag is carried by a user. Thus the location information of the tag is mapped onto the location information of the user entity.

We used C++ and java as program languages of the system build on Windows XP operating system. As for the transport layer, we used TCP in the control part and UDP for the video transmission. We constructed the environment with two kinds of networks: wired (Ethernet 100 Mbps) and wireless (IEEE802.11g, 54 Mbps) networks. In addition, we used a Web camera to capture the video and PC display to receive and play the video delivered from the Web camera.

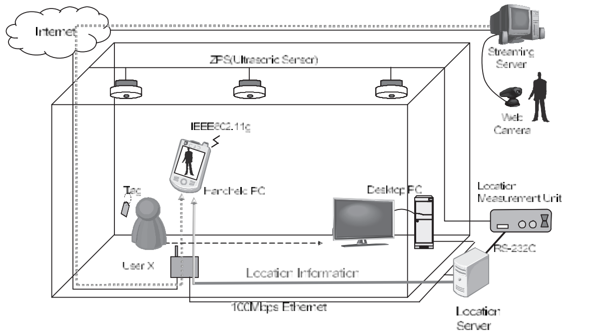


Figure 6: Experimental environment

As for the default setting, a handheld PC which is always carried by a user displays the video form a Web Camera in Figure 6. Moreover, the destination of output of the video migrates from the handheld PC to the display of the desktop PC when the user approaches a desktop PC. *User* and *UserReq* reside in the handheld PC. *User* manages the user's location information receiving from *ZPS*.

5 EXPERIMENTS AND EVALUATION

5.1 Experimental Method

We performed experiments to confirm effectiveness of the proposed scheme. In these experiments, we evaluated that the system can satisfy QoS requirement of user by QoC control which is adapted to the user's location and moving speed. We use a prototype system described in Section 4.2.

We measured frame rate in receiving the streaming video. This corresponds to the $Service{-}Quality$ parameter of QoS described in Section 3.1. We also measured the time that has been spent on the migration of the video streaming from the handheld PC to the desktop PC. This is regarded as the $Timeliness$ parameter of QoS. Moreover, we used frequency of location information update as the $up{-}to{-}dateness$ parameter of QoC.

This experiment uses the scenario of Figure 4. The user moves from point A to point B at a constant speed with the

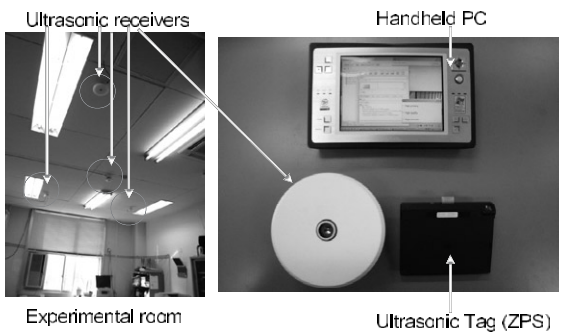
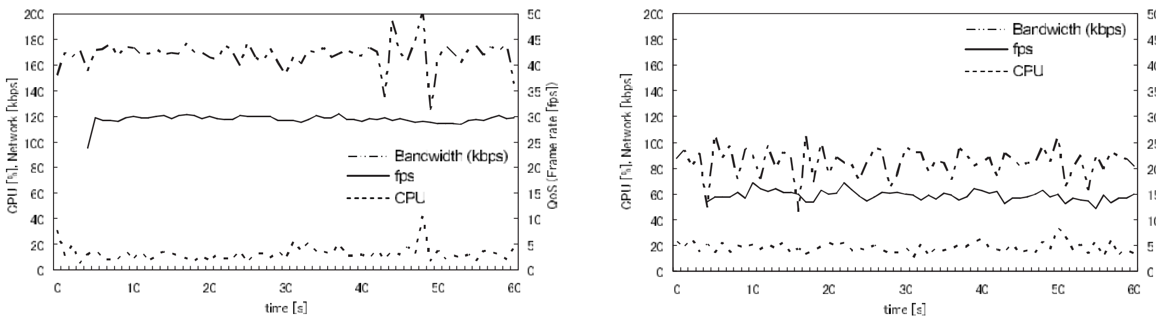


Figure 7: A snapshot of hardware configuration for experimental system



(a) In case of 30fps (b) In case of 15fps

Figure 8: Experimental results of the handheld PC's resources consumption without receiving the location information

handheld PC. The service is migrated when the user approaches to a desktop PC at point B.

We used Expr. (1) and Expr. (2) which are described in Section 3.5 to determine QoC. In this experiment, the parameters of Expr. (1), they are, X, w, p, and α, were fixed and set to the following values based on the experience of preliminary experiments.

$$X = 5550,\ w = 10^3,\ p = 0.1,\ \alpha = 1000$$

We also assumed that the QoS requirement of the user is 30 fps. In addition, the system considers this situation as QoS violation and sends QoC change request to $Manager$ agent; its request value is decided by Expr. (2) when the 5-seconds average of the frame rate decreases less than 28 fps. For the video streaming with 30 fps and 15 fps, the handheld PC's cpu usage, network useage, and fps without receiving the location information are shown in Figure 8, respectively.

This prototype system can calculate the user's speed (v) dynamically and can use it to decide QoC value. However, to show the effect of the proposed scheme clearly, we set v to the fixed value as 0.1, 0.5, 1.0, and 2.0, and compared the results from each case.

5.2 Experimental Results

Figure 9 shows the experimental results. When the user moved from point A to point B, these graphs represent temporal changes of QoS (fps) of received video and QoC (update frequency per second). In each graph, fps falls rapidly on the way and afterwards rises again. This shows that the streaming video migrates from handheld PC to desktop PC. In Figure 9, (a)∼(d) are results of each case of fixing the value of v as 0.1, 0.5, 1.0, and 2.0.

In any case, when the user approached point B from point A, QoC was observed to increase. The case of $v = 0.1$ (Figure 9(a)), for example, after the user left from point A at once, QoC was once per 3 or 4 seconds, and then, QoC was updated to once per 2 seconds at about 30 seconds. After that, QoC was updated 1 time/s at about 50 seconds and the service migrated to desktop PC smoothly. In addition, the average frame rate was 29.1 fps and our proposed scheme did not affect to the QoS requirement of the user.

From a viewpoint of difference of moving speed of the user, it was observed that the more the speed increased, the higher QoC became. In case of $v = 0.5$ (Figure 9(b)), for example, after the user left from point A at once, QoC was set 1 time/s, and updated to 2 times/s at about 10 seconds. After that, QoC was updated 3 times/s at about 40 seconds. Compared with the case of $v = 0.1$ (Figure 9(a)), it is found that, the more the user approached the desktop PC, the more the update frequency of QoC increased.

Moreover, in cases of $v = 1.0$ and 2.0, we also observed QoC control behavior based on QoS monitoring. In case of $v = 1.0$ (Figure 9(c)), when 30 seconds passed after the user left from point A, QoC increased to about 3.5 times/s, but when at the point of 40 seconds, it decreased to about 2 times/s, and it increased to about 4.5 times/s again at the 45 seconds point. We can analyze this behavior as follows: fps had decreased at about 35 seconds; subsequently, *Manager* agent on the handheld PC sent the request to decrease QoC to decrease QoS (fps); and then *Manager* agent sent the request to increase QoC when fps was recovered. We can also see the same behavior occurred during 60 seconds to 90 seconds. In case of $v = 2.0$ (Figure 9(d)), the control behavior was observed within the range of 15 seconds to 22 seconds; QoC decreased to 2 times/s. From these results of QoC control based on fps monitoring, the service migration process was successfully executed at about 57 seconds.

We also show a result of the experiment in which the *fps* agent was inactivated. In this case, QoC control based on change of QoS is disabled. The result in case of $v = 2.0$ is shown in Figure 10.

Compared with the situation where fps agent alives (Figure 9(d)), QoC increased gradually and it did not decreased. Moreover, QoC started to decrease from 70 seconds. This is because QoC became 8 times/s at 40 seconds and this ex-

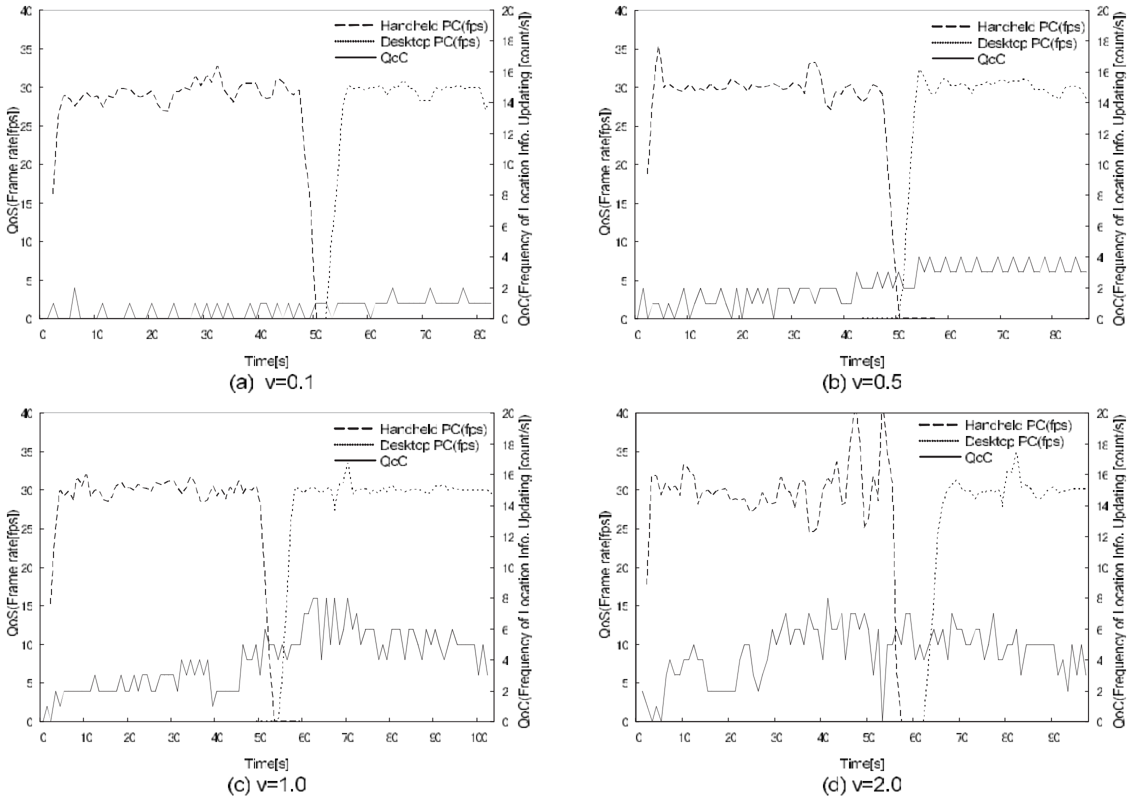


Figure 9: Experimental results of service migration experiment (v as 0.1, 0.5, 1.0, and 2.0)

cessive QoC caused the unstable frame rate. Therefore, we found that the CPU resource of handheld PC decreased and QoC processing did not catch up.

Additionally we are trying to experiment on using a data communication network of Personal Handy-phone System (PHS) as the network link of the handheld PC. Figure 11 shows the result in case of $v = 0.1$ using PHS/128kbps. In this case, we assumed that the QoS requirement of the user is 15 fps. In the same way as the case of the handheld PC used IEEE802.11g (Figure 9(a)), QoC was observed to increase when the user approach point B from point A, and our proposed scheme did not affect to the QoS requirement (15fps). This result shows that our proposed scheme can apply various network environments.

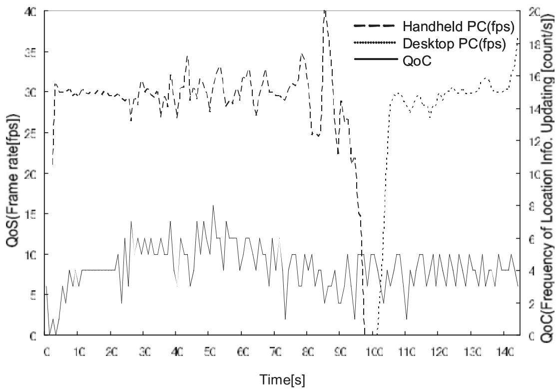


Figure 10: Experimental result in case of no fps agent available ($v = 2.0$)

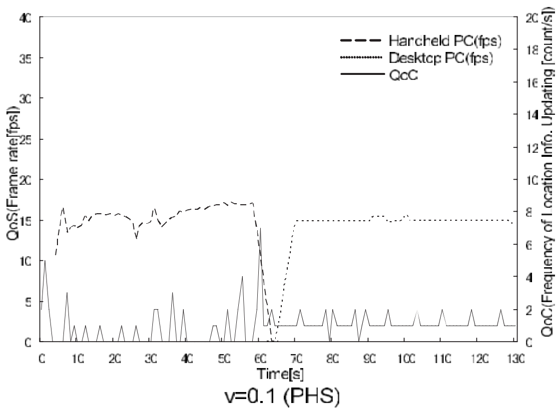


Figure 11: Experimental result in case of using PHS ($v = 0.1$)

5.3 Discussion

We show that it is possible to control QoC dynamically based on user's position and moving speed from the experimental results. We also show the problem that providing excessive QoC causes decrease in QoS depending on the user's speed. We confirmed that the system could tune QoC properly to improve QoS by the effect of our proposed QoC control scheme. In this experiment, we did not show the effect of Adaptation Function (F2); however, we give the basis of the function. Further investigation is needed to design and implement the (F2).

Here we discuss the concept of "service session". The service session means a time period when sets of service provisioning are executed in the same configuration of real space. From this classification viewpoint, the function (F1) is regarded as an adaptation in a single service provisioning; whereas (F2) is an adaptation in a single service session based on the evaluation of each single service provisioning. We have to cope with the adaptation through multiple service sessions to realize more effective QoC control. This will contribute to reduce adaptation overhead greatly.

In terms of the improvement of (F1), the existing system keeps same QoC after the user arrives at point B, because QoC control after the service migration is not considered. In the future work, we need to cope with such case when it is unnecessary to increase QoC after service migration; for instance, by suppressing the QoC value.

6 CONCLUSION

We presented a dynamic control scheme of context information delivery based on multi-agent, to develop an effective context information managing scheme for context-aware service with appropriate QoS according to user's request and changing real space. We also designed and implemented an initial experimental system. From results of experiments by the experimental system, we confirmed the effectiveness of the proposed scheme; it can manage an up-to-dateness (QoC) of positional information according to the QoS, user's position and user's speed. We can greatly improve the adaptation ability of the ubicomp space to the real space by employing our proposed scheme.

We are planning to design and develop (F2) and evaluate it. We only use fps as the QoS parameter and the up-to-dateness of location information as the QoC parameter so far using an experimental system. Our future work includes an extension of the parameter of QoS and QoC, and evaluation considering many situations such as healthcare service using vital sensor and environmental sensor. As for QoS, we will use picture size or resolution as a QoS parameter, or accuracy or granularity as a QoC parameter. Moreover we would like to advance detail definition and modeling of relation of QoS and QoC.

ACKNOWLEDGEMENT

This work was partially supported by Sendai Intelligent Knowledge Cluster and the Ministry of Education, Culture, Sports, Science and Technology, Grants-in-Aid for Scientific Research, 19200005.

REFERENCES

[1] M. Weiser, The Computer for the Twenty-first Century, Scientific American, Vol. 265, No. 3, pp. 94–104 (1991).

[2] N. Davis, K. Cheverst, K. Mitchell and A. Efrat, Using and determining location in a context-sensitive tour guide, IEEE Comput., Vol. 34, No. 8, pp. 35–41 (2001).

[3] T. Yamazaki, Ubiquitous Home: Real-life Testbed for Home Context-Aware Service, Proc. of First International Conference on Testbeds and Research Infrastructures for the DEvelopment of NeTworks and COMmunities (TRIDENTCOM'05), pp. 54–59 (2005).

[4] E. D. Mynatt, A.-S. Melenhorst, A. D. Fisk and W. A. Rogers, Aware technologies for aging in place: understanding user needs and attitudes, IEEE Pervasive Comput., Vol. 3, No. 2, pp. 36–41 (2004).

[5] T. Nakamura, M. Matsuo and T. Itao, Context handling architecture for adaptive networking services, Proc. of IST Mobile Summit 2000, pp. 295–302 (2000).

[6] K. Fujinami, T. Yamabe and T. Nakajima, Take me with you! : A Case Study of Context-aware Application integrating Cyber and Physical Spaces, Proc. of ACM Symposium on Applied Computing (SAC) 2004, pp. 1607–1614 (2004).

[7] T. Buchholz, A. Kupper and M. Schiffers, Quality of context: what it is and why it need", Proc. of the Workshop of the HP OpenView University Association 2003 (HPOVUA2003) (2003).

[8] K. Sheikh, M. Wegdam and M. van Sinderen, Middleware Support for Quality of Context in Pervasive Context-Aware Systems, Proc of the 5th Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PerComW'07) (2007).

[9] M. C. Huebscher and J. A. McCann, An adaptive middleware framework for context–aware applications, Springer Personal and Ubiquitous Computing Journal, Vol. 10, No. 1, pp. 12–20 (2005).

[10] Y. Tokairin, K. Yamanaka, H. Takahashi, T. Suganuma and N. Shiratori, An Effective QoS Control Scheme for Ubiquitous Services based on Context Information Management, Proc. of International Workshop on NGN and their Impact on E-Commerce and Enterprise Computing(NGN-EC2) in conjunction with CEC'07 and EEE'07, pp. 619–625 (2007).

[11] S. Fujita, H. Hara, K. Sugawara, T. Kinoshita and N. Shiratori, Agent-based Design Model of Adaptive Distributed System, The International Journal of Artificial Intelligence, Neural Networks and Complex Problem-Solving Technologies, Vol. 9, No. 1, pp. 57–70 (1998).

[12] H. Takahashi, Y. Tokairin, T. Suganuma and N. Shi-

ratori, Design and Implementation of An Agent-based middleware for Context-aware Ubiquitous Services, Frontiers in Artificial Intelligence and Applications, New Trends in Software Methodologies, Tools and Techniques (Proc. of the 4th SoMeT2005), Vol. 129, pp. 330–350 (2005).
[13] T. Uchiya, T. Maemura, L. Xiaolu, and T. Kinoshita, Design and Implementation of Interactive Design Environment of Agent System, Proc. of the 20th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE 2007), LNCS4570, pp. 1088–1097 (2007).
[14] Furukawa CO,. LTD., ZPS (Zone Positioning System), http://www.furukawakk.jp/products/ZPS_1.html (in Japanese)
[15] Sun Microsystems, Java Media Framework(JMF), http://java.sun.com/products/java-media/jmf/.

(Received October 15, 2008)
(Revised July 16, 2009)

Hideyuki Takahashi is a research fellow of Research Institute of Electrical Communication of Tohoku University, Japan. He received his doctoral degree in Information Sciences from Tohoku University in 2008. His research interests include ubiquitous computing and agent-based computing. He is a member of IPSJ.

Yoshihisa Sato received M. S. degree in 2009 from Tohoku University, Japan. Currently, he works for the IBM Corp. His research interests include agent-based computing and context-aware service provisioning.

Takuo Suganuma is an associate professor of Research Institute of Electrical Communication of Tohoku University, Japan. He received a Dr.Eng. degree from Chiba Institute of Technology. He received UIC-07 Outstanding Paper Award in 2007, etc. His research interests include agent-based computing, flexible network, and symbiotic computing. He is a member of IEICE, IPSJ and IEEE.

Norio Shiratori received his doctoral degree from Tohoku University, Japan in 1977. Presently he is a Professor of the Research Institute of Electrical Communication, Tohoku University. He has been engaged in research related to symbiotic computing paradigms between human and information technology. He was the recipient of the IPSJ Memorial Prize Winning Paper Award in 1985, the Telecommunication Advancement Foundation Incorporation Award in 1991, the Best Paper Award of ICOIN-9 in 1994, the IPSJ Best Paper Award in 1997, the IPSJ Contribution Award in 2007, and many others. He was the vice president of IPSJ in 2002 and now is the president of IPSJ. He is also a fellow of IEEE and IEICE.

Submission Guidance

About IJIS

International Journal of Informatics Society (ISSN 1883-4566) is published in one volume of three issues a year. One should be a member of Informatics Society for the submission of the article at least. A submission article is reviewed at least two reviewer. The online version of the journal is available at the following site: http://www.infsoc.org.

Aims and Scope of Informatics Society

The evolution of informatics heralds a new information society. It provides more convenience to our life. Informatics and technologies have been integrated by various fields. For example, mathematics, linguistics, logics, engineering, and new fields will join it. Especially, we are continuing to maintain an awareness of informatics and communication convergence. Informatics Society is the organization that tries to develop informatics and technologies with this convergence. International Journal of Informatics Society (IJIS) is the journal of Informatics Society.

Areas of interest include, but are not limited to:

- Computer supported cooperative work and groupware
- Intelligent transport system
- Distributed Computing
- Multi-media communication
- Information systems
- Mobile computing
- Ubiquitous computing

Instruction to Authors

For detailed instructions please refer to the Authors Corner on our Web site, http://www.infsoc.org/.

Submission of manuscripts: There is no limitation of page count as full papers, each of which will be subject to a full review process. An electronic, PDF-based submission of papers is mandatory. Download and use the LaTeX2e or Microsoft Word sample IJIS formats.

http://www.infsoc.org/IJIS-Format.pdf

LaTeX2e

LaTeX2e files (ZIP) http://www.infsoc.org/template_IJIS.zip

Microsoft Word™

Sample document   http://www.infsoc.org/sample_IJIS.doc

Please send the PDF file of your paper to secretariat@infsoc.org with the following information:

Title, Author: Name (Affiliation), Name (Affiliation), Corresponding Author. Address, Tel, Fax, E-mail:

Publisher

Address: Informatics Laboratory 3-41, Tsujimachi, Kitaku, Nagoya 462-0032, Japan

E-mail: secretariat@infsoc.org

CONTENTS