Vol. 45 No. 2 IPSJ Journal Feb. 2004

Regular Paper

Source Mobility Support Multicast (SMM)

Katsuhiko Sato,†,††,††† Michiaki Katsumoto††
and Tetsuya Miki†††

A number of multicast and mobile routing techniques have been developed for IP networks. However, multicasting that supports the mobility of the source host has not received sufficient attention. This paper proposes a source mobility support multicast (SMM) technique that guarantees consecutive communication in high-speed mobile environments and minimizes the overhead in reconstructing multicast trees which caused by a source movement. SMM is based on a new concept of constructing multicast trees over Cellular IP [13] based networks, in which a source point tree and a rendezvous point tree are combined into one multicast tree. It also provides wide operability by partitioning the network and confining protocol uniqueness and topological constraints within the local access networks. This paper gives an overview of the SMM technique and protocol descriptions, and presents the results of numerical analyses to minimize the tree reconstruction overhead due to a source movement.

1. Introduction

The broadband infrastructure for the distribution of information has been growing intensively on the basis of the Internet technologies. To enable high-quality video distribution and worldwide mobile communication, a number of advanced IP routing techniques have been developed. Multicasting presents additional advantages for optimal network resource utilization when the same data are distributed to multiple points.

We have been studying multicast techniques for high-quality video distribution in high-speed mobile environment, which is aimed at intelligent transport system (ITS) based on the broadband Internet. Previous research on multicasting that supports host mobility, which is represented by MoM (Mobile Multicast) [20], has been abundant, however, all MoM based techniques have not focused on the source host mobility and guarantees of consecutive communication since MoM is based on Mobile IP [10]. We propose a new concept of multicasting over Cellular IP [13] to perform consecutive communication and technique that combines a source point tree and a rendezvous point tree into one multicast tree to minimize the tree reconstruction overhead due to a source movement. We also provide wide operability by partitioning the network and confining protocol uniqueness and topological constraints within the local access networks. We name these ideas Source Mobility Support Multicast (SMM). This paper gives an overview of the SMM scheme, detail protocol descriptions and numerical analyses for SMM performance. The results of analyses indicate SMM extremely reduces the tree reconstruction overhead —— the number of nodes updating the multicast forwarding cache table —— comparing with conventional multicast techniques.

This paper is organized as follows. Section 2 reviews recent studies that operates both multicast and mobile routing, and proposes new multicast concept that supports source mobility and consecutive communication. Section 3 shows an architecture of SMM and protocol overview and gives a detailed description of the SMM protocol. Section 4 shows numerical analyses performed to minimize the tree reconstruction overhead.

2. Related Works and Proposal of New Source Mobility Support Multicasting

2.1 Reviewing Current Multicast and Mobile Routing

Multicast routing minimizes consumption of network bandwidth resources, in which a single flow sent from a source is shared by many receiving hosts, which are called multicast group members. Switch nodes copy the flow and forward it to the networks with active members. As a result, a distribution tree is formed, which is called a multicast tree. A number of multicast routing protocols have been de-

† Japan Radio Co., Ltd
†† Communication Research Laboratory
††† The University of Electro Communication

veloped in IETF (Internet Engineering Task Force). These protocols can be classified into two types: implicit protocols [7),9)] and explicit protocols [1)~3),6)]. In implicit protocols, to construct a multicast tree, multicast data are periodically broadcast throughout the network, and the links with no active members are pruned. Because unnecessary traffic is still periodically sent to links with no active members until the pruning has been done, this model is effective only when the network has abundant bandwidth resources or when the density of group members is high.

In explicit protocols, to construct a multicast tree, explicit messages are sent from the members to a source point or a rendezvous point on the tree. In the case of the source point, the formed multicast tree is called a source point tree (SPT), while in the case of the rendezvous point, it is called a rendezvous point tree (RPT). Although this model needs a mechanism that broadcasts the source or rendezvous point's address to all members or network nodes, it does not transmit unnecessary traffic, which is in contrast to the implicit model. Therefore, this model can be considered more optimal when we operate multicasting on a wide scale.

Mobile routing aims at maintaining communication without changing the host's address whenever the host changes their location. Mobile routing mechanisms can be classified into two groups: macro-mobility for wide area and micro-mobility for local area. A representative macro-mobility mechanism in IP networks is Mobile IP [10)]. In this mechanism, a certain node (Home Agent: HA) managing the mobile host's location forwards, through tunneling, data packets with care-of address to another node (Foreign Agent: FA) to which the mobile host attaches. This scheme is known to suffer from the triangle route problem, or route redundancy, when all data packets are necessarily forwarded via an HA, and there is a time overhead needed to establish a tunnel between an HA and an FA, although many improvements (e.g., route optimization and a fast handover mechanism, etc.) [11),12)] have been proposed and discussed in IETF.

Cellular IP [13)~15)] is a representative micro-mobility mechanism. In Cellular IP, switch nodes physically organize the tree topology. All of them have the routing caches to forward data packets. A data-forwarding path is dynamically established every time data packets originate from a mobile host. Some of switch nodes near root of the tree have the paging caches to manage the host's location. To page a mobile host, the Page message is broadcasted in a paging area according to the paging caches. The paging caches are updated by the Paging Update message from a mobile host whenever it changes the paging area. As one feature, Cellular IP supports the diversity handover where multiple data-forwarding paths are established simultaneously while mobile host is moving between two radio base stations. This guarantees consecutive communication.

Hence, Mobile IP, which is based on general IP unicast routing suite, is simple and can thus be operated for wide area although it has some route redundancy. Meanwhile, Cellular IP enables guaranteeing consecutive communication, but it can be operated only local area since this unique routing scheme is completely independent of the general IP unicast routing. Therefore, Mobile IP is used for backbone of Cellular IP networks.

There have been a number of studies that addressed both multicast and mobile routing. The primary subject of mobile routing involves address translation depending on the location. Multicast routing, on the other hand, provides a mechanism for location-independent addressing. Mysore and Bharghavan [17)], and Helmy [18),19)] have proposed a micro-mobility routing architecture based on the multicast infrastructure, which enables fast handover. However, these studies have not focused on mobility support multicasting.

To provide multicasting with mobility functions, several techniques have been proposed on the basis of Mobile IP, where an HA sets up a tunnel to the FA to which the mobile host attaches and forwards multicast data to the FA. Harrison, Williamson, and colleagues proposed the MoM (Mobile Multicast) [20)] technique as an early mobility support multicast technique. The MoM technique solves the tunnel convergence problem, which occurs as a result of multiple tunnels from different HAs terminating at a common FA. In this protocol, an FA performs a selection to appoint one HA as a designated multicast service provider (DMSP). However, if a mobile member whose HA has been selected as the DMSP is moved out of the FA, DMSP handover must occur, and it affects the remaining members' communication. To solve this problem, Wang and Chen developed a multicast agent (MA) that establishes a tunnel from itself to the FA instead of the HA[21]. However, as the distance between the FA and MA increases, so does the tunnel length, which causes the redundant route problem. Lin and Wang proposed a range-based MoM technique[22], which defines the length over which the MA can provide a tunnel to the FA. This scheme reduces the length of the tunnel and solves the redundant route problem. Rolland, Luis, and colleagues showed through numerical analyses the advantages of retrenching a redundant tunnel route[24].

All these techniques are based of Mobile IP. Therefore, they cannot completely eliminate redundant routes in tunneling. In addition, the above studies did not focus on the guarantee of consecutive communication, and did not consider the situation when the multicast source is moving at a high speed. In particular, no attempt has been made to develop a diversity handover mechanism and minimize the overhead in reconstructing a multicast tree, which caused by the source movement. We focus on these points and discuss a new multicast technique that supports effective source mobility and consecutive communication.

2.2 Considerations for Source Mobility and Consecutive Communication

This paper focuses on explicit multicasting from the point of view of optimal network resource utilization. The followings are discussions of explicit multicasting for supporting source mobility and consecutive communication.

First, conventional explicit multicasting requires a broadcast mechanism that transmits connection point information (a set of source or rendezvous point addresses corresponding to a multicast group address) in advance to all group members or switch nodes to which the members are attached. In PIM-SM (Protocol Independent Multicast - Sparse Mode)[1] especially, a certain switch node (called bootstrap router) periodically transmits connection point information to the entire network. This mechanism is called the bootstrap mechanism. However, such periodical transmitting is unacceptable in high-speed mobile environments, because it cannot provide correct connection point information in real time as tracking the location of the moving source point.

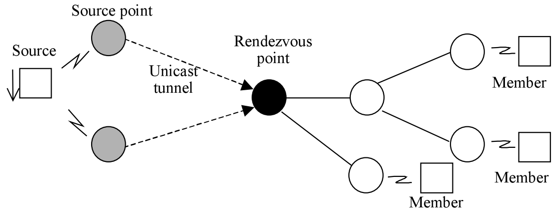

Fig. 1 Diversity handover for the moving source with RPT.

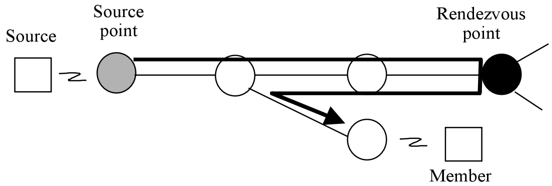

Fig. 2 Backtrack route problem with RPT.

Second, we consider diversity handover while the source moving and the overhead in reconstructing a multicast tree. With RPT, diversity handover for the moving source can be easily performed by establishing two paths from both the previous and new source points to the rendezvous point during transition. In PIM-SM, the two paths can be established by unicast tunneling as shown in **Fig. 1**. With SPT, diversity handover for the moving source requires the construction of another tree. In PIM-SM, every multicast forwarding cache entry that the switch nodes organizing the tree maintain must be changed, since it contains the source point address. Hence, the source handover with SPT introduces a considerable overhead, compared with that with RPT. However, RPT has a backtrack route problem as shown in **Fig. 2**, which occurs when the route to the receiving members is branched from the intermediate nodes between the source and the rendezvous point[4]. In reality, PIM-SM eliminates this redundancy of backtrack route by switching from RPT to SPT. Hence, RPT has an advantage of the free from the overhead in reconstructing multicast tree due to the source movement but it has disadvantage of the route redundancy.

Based on the above considerations, we discard the bootstrap mechanism; in this study, the connection point is determined statically and every node can thereby send an explicit-join message to the connection point without knowing the connection point address by periodical bootstrap massages. In addition, the network is physically organized a tree topology and every node can thereby forward the

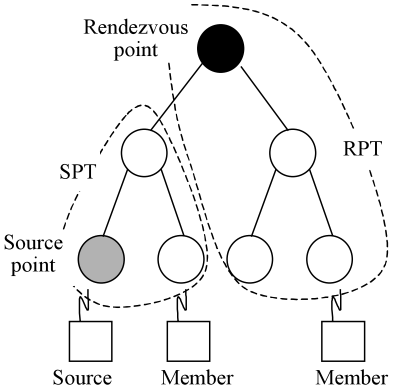

Fig. 3 Combination of RPT and SPT over tree topology.

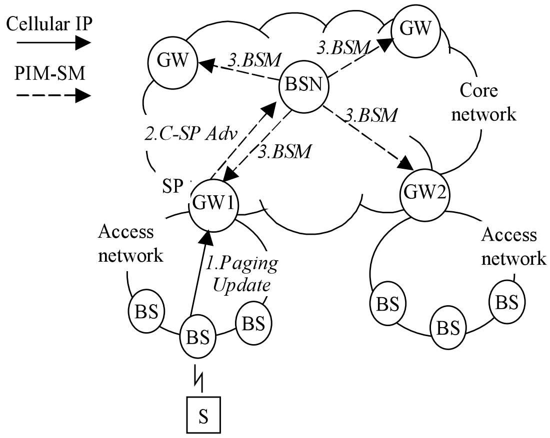

Fig. 4 Overview of SMM Proc. 1: The S transmits its multicast group and location to GW1 using a *Paging Update* message. GW1 becomes the source point for the group and sends an *C-SP Adv* message to BSN. The BSN broadcast *BSM* messages to all GWs.

explicit-join message without knowing a route to the connection point. Organizing such tree topology, Cellular IP can establish the forwarding path without general unicast routing protocols.

And we use both tree; an RPT to minimize the overhead in tree reconstruction caused by source movement and an SPT to eliminate redundant route for the members between the source and rendezvous points. We remove the unicast tunneling from the source to the rendezvous point, which is performed by PIM-SM, and construct an SPT only for the members whose distance from the source point is shorter than the distance from the source to the rendezvous point. In this way, the range of reconstructing tree is limited to around the moving source, and the backtrack route between the source and the rendezvous points is eliminated. **Figure 3** depicts the combination of RPT and SPT on a physical tree topology.

However, such topological constraints and protocol uniqueness obviously affect the network scalability. To solve this problem, the network should be partitioned into core and access networks, and functions should be deployed according to its scale and requirements. Both core and access networks are hierarchically connected, and the protocol uniqueness and topological constraints are confined in the access networks. The access networks organize a physical tree topology and form both RPT and SPT as shown in Fig. 3. Since the handover is to occur frequently in the access networks, consecutive communication is enabled there at the cost of protocol generality and network scalability. On the other hand, we allow the core network to be universal and scalable by using the conventional unicast and multicast routing schemes. The core network constructs an SPT by using PIM-SM and performs tunneling between the access networks, while the multicast source is crossing over the access networks. Since the handover is to occur less frequently in the core network, we allow for a redundant route by tunneling there.

3. SMM Descriptions

3.1 SMM Architecture and Protocol Overview

In this section, we present the basic scheme of our source mobility support multicast (SMM) technique. As shown in **Fig. 4**, the network consists of one core network and several access networks that are connected in a star-like fashion with gateways (GWs) on the boundary.

The core network allows the switch nodes to form any topology. The core network constructs an SPT based on PIM-SM where GWs become the source points (SPs). A bootstrap node that transmits the SPs' addresses throughout the core network is deployed in the core network. In the access networks, the switch nodes form a tree topology. The GWs are the root of the trees, and the radio base stations (BSs) are the leaves of the tree.

SMM operates different protocols on each network. **Figure 7** depicts the protocol stack where the three entities are defined as SMM-C (SMM for the core network), SMM-A (SMM for the access networks), and SMM-O (SMM

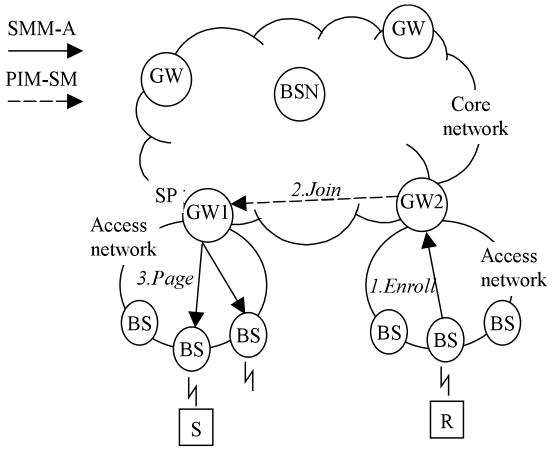

Fig. 5 Overview of SMM Proc. 2: The R sends an *Enroll* message for the multicast group to GW2 to create forwarding cache entries between the R and GW2 (SP). GW2 sends *Join* message to GW1. GW1, if it has not received multicast data, sends *Page* message to all BSs.

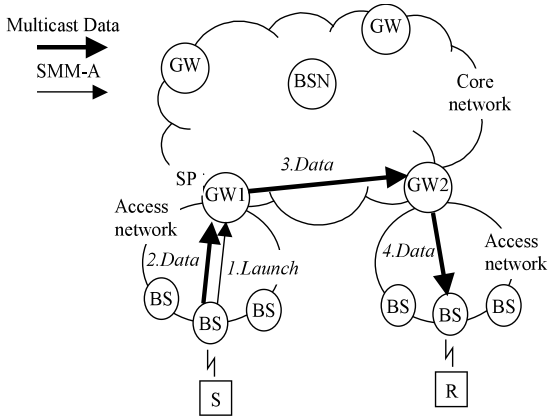

Fig. 6 Overview of SMM Proc. 3: Upon receiving the *Page* message, the S replies to GW1 with *Launch* message to create forwarding cache entries between the S and GW1. The S starts sending multicast data, and the data are forwarded to GW1, GW2 and R.

for co-ordination). Hereinafter, the multicast source and the receiving members are expressed as S and R, respectively (as shown in Fig. 7). SMM-C is located on the PIM-SM suite, and it provides tunneling function between two GWs. SMM-A is located on the Cellular IP suite, it operates both SPT and RPT, and it performs diversity handover. SMM-O provides coordination between SMM-C and SMM-A.

SMM performs the following basic procedures. First, as shown in Fig. 4, the S transmits its location to GW1 using a *Paging Update* (specified in the Cellular IP). The *Paging Update* includes the multicast group address in addition to the S's unicast address and paging area. Upon receiving the *Paging Update*, GW1 sends an *C-SP Adv* (SP candidate advertisement, specified in the PIM-SM) to the BSN, which says that GW1 has become the SP of the core network for the multicast group. The BSN sends a bootstrap message (*BSM*) to all GWs in the core network, which includes the GW1 address and the multicast group address. These procedures are performed as the bootstrap mechanism of PIM-SM.

The R sends an *Enroll* message (specified in the SMM-A) for the multicast group to GW2 (**Fig. 5**). Intermediate nodes between the R and GW2 create a cache entry (specified in the SMM-A) to forward multicast data packets for the group. After receiving the *BSM*, GW2 obtains the SP address (i.e., GW1 address) for the multicast group from the received *BSM* and sends *Join* massage to GW1. The *Join* massage and the cache entry are specified as the SPT mechanism of PIM-SM. Having received the *Join* message, GW1, if it has not received multicast data packets for the group, sends a *Page* message (originally specified in the Cellular IP and modified by the SMM-A) to all BSs in the paging area where the S is.

Upon receiving the *Page* message, the S replies to GW1 with a *Launch* message (specified in the SMM-A) as shown in **Fig. 6**. Intermediate nodes between the S and GW1 create a cache entry (specified in the SMM-A) to forward multicast data packets for the group. The S then starts sending multicast data packets, and these packets are forwarded to GW1, GW2, and R according to each multicast forwarding cache entries.

3.2 Detailed Descriptions of SMM

This section describes detailed procedures of SMM. The main features, a combination of RPT and SPT and diversity handover, are condensed within the access networks as SMM-A functions.

3.2.1 SSM-A over Cellular IP in the Access Network

As shown in **Fig. 8**, switch nodes (N1-N7) organize a tree topology. N1 is the root of the tree, and radio base stations (B1-B8) are attached as the leaves of the tree. Like in the Cellular IP, the tree is partitioned into several paging areas (PA1 and PA2). B1-B4 belong to PA1, and B5-B8 belong to PA2. N1 has a paging cache to determine in what paging

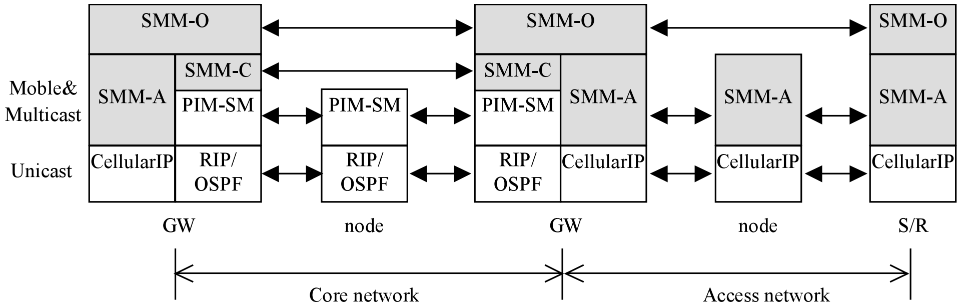

Fig. 7 Protocol Stack: SMM consists of SMM-A, SMM-C and SMM-O. SMM-A for the access network is located on the Cellular IP and SMM-C for the core network is located on the PIM-SM.

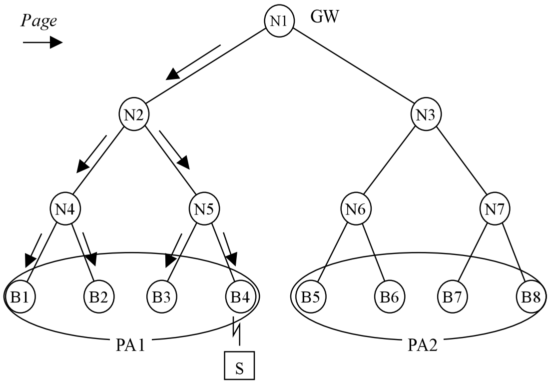

Fig. 8 SMM-A Proc. of forming RPT and SPT (1): The S is at PA1 and N1 has a paging cache. N1 sends *Page* message to N2. N2, N4 and N5 forward the *Page* to all downstream interfaces. B1-B4 broadcast the *Page* via an air interface.

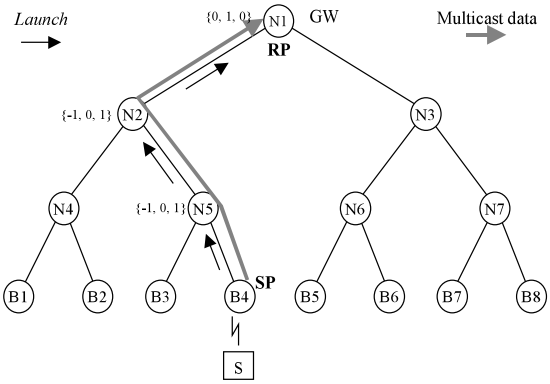

Fig. 9 SMM-A Proc. of forming RPT and SPT (2): Now, the S is at B4 cell and sends *Launch* message to N1 to set up a path from the S to RP (N1). $E_{\mathrm{G}}(\mathrm{N5}) = \{-1, 0, 1\}$, $E_{\mathrm{G}}(\mathrm{N2}) = \{-1, 0, 1\}$ and $E_{\mathrm{G}}(\mathrm{N1}) = \{0, 1, 0\}$ are created. Hence, constant data path is established between S and RP.

area the mobile host is. N2-N7 having one upstream interface and two downstream interfaces can recognize whether an interface is upstream or downstream.

All switch nodes maintain multicast forwarding cache entries (MFCEs) to determine to which interface to forward multicast data packets. Each entry has a multicast group address and the state of each interface to indicate whether multicast data packets are incoming or outgoing. Each entry is here expressed as $E_g(n) = \{X_0, X_1, \ldots, X_i\}$. The g and n refer to a multicast group address and a certain node, respectively. In Figs. 8–12, for example, $n = \mathrm{N1}, \ldots, \mathrm{N7}$. X_i shows the interface state: $X_i = 1$ is the incoming state, $X_i = -1$ is the outgoing state, and $X_i = 0$ is nothing. The i identifies the interface: $i = 0$ is an upstream interface, and $i > 0$ is a downstream interface. In Figs. 8–12, for example, $i = 1$ and $i = 2$ are, respectively, the left- and right-side downstream interfaces.

3.2.1.1 Constructing RPT and SPT

SMM-A constructs both RPT and SPT in one multicast tree, where the rendezvous point (RP) is at N1, and the source point (SP) is at one of the radio base stations. First, GW (N1) pages the source of multicast (S) and asks it to start sending multicast data. Here, the S is at PA1 as shown in Fig. 8. N1 pages the S in PA1. It sends a *Page* message to N2 according to the paging cache. N2, N4, and N5 forward the *Page* to all downstream interfaces, and B1-B4 broadcast the *Page* via an air interface. The *Page* contains the multicast group address (g = G) that specifies the source of the multicast group.

Now, the S is in the B4 cell. It sends a *Launch* message to N1 as a reply to the *Page* to set up a path from the S to the RP (**Fig. 9**). B4, N5, and N2 receiving the *Launch* forward it to their upstream interface. The *Launch* includes multicast address G used for the multicast data

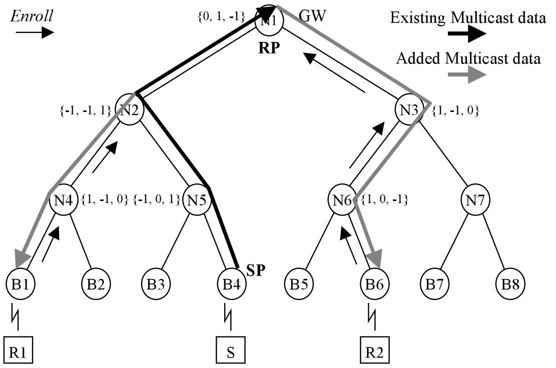

Fig. 10 SMM-A Proc. of forming RPT and SPT (3): R1 and R2 are at B1 and B6 cells, respectively. They send *Enroll* message to N1. N4, N6 and N3 have no MFCE for G, then, they create $E_{\mathrm{G}}(\mathrm{N4}) = \{1, -1, 0\}$, $E_{\mathrm{G}}(\mathrm{N6}) = \{1, 0, -1\}$ and $E_{\mathrm{G}}(\mathrm{N3}) = \{1, -1, 0\}$. N2 already has MFCE for G, then, it update to $E_{\mathrm{G}}(\mathrm{N2}) = \{-1, -1, 1\}$.

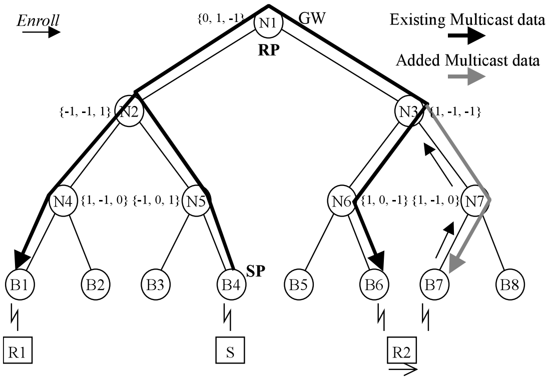

Fig. 11 SMM-A Proc. of handover of receiving member: R2 is moving to B7 cell. R2 sends *Enroll* message to N3 via N7. N7 has no MFCE for G, then, it creates $E_{\mathrm{G}}(\mathrm{N7}) = \{1, -1, 0\}$. N3 already has MFCE for G, then, it update to $E_{\mathrm{G}}(\mathrm{N3}) = \{1, -1, -1\}$. $E_{\mathrm{G}}(\mathrm{N6})$ is removed by timer or *Leave* message.

packets. N5 and N2 create an MFCE for the G and set their interface state. The upstream interface ($i = 0$) is set to the outgoing state ($X_0 = -1$), and the downstream interface at which the *Launch* message is received ($i = \mathrm{R}$) is set to the incoming state ($X_{\mathrm{R}} = 1$). Specifically, N5 creates $E_{\mathrm{G}}(\mathrm{N5}) = \{-1, 0, 1\}$, and N2 creates $E_{\mathrm{G}}(\mathrm{N2}) = \{-1, 0, 1\}$. The S then starts sending multicast data, and the data are forwarded to N1 (i.e., RP) by following the E_{G} (N5) and E_{G} (N2). A *Launch* message is periodically sent by the S and forwarded to N1. Intermediate nodes, N5 and N2, refresh their MFCE's interface state (if they do not receive a *Launch* message from the S during a certain predetermined period of time, they remove the MFCE for the G).

Hence, a constant data path between the S and RP has been established in preparation for the construction of RPT and SPT.

The receiving members of the G (R1 and R2) appear in the B1 and B6 cells, respectively. R1 being close to the S constructs an SPT with a source point at B4, and R2 being far from the S constructs an RPT with a rendezvous point at N1 (as shown in **Fig. 10**).

Each of receiving members sends an *Enroll* message to N1, which contains the multicast address, G. Radio base stations and switch nodes receiving the *Enroll* message forward it to their upstream interface, and the switch nodes see the multicast address in the *Enroll* message and look it up in their MFCE lists. There is no MFCE for the G at N4, N6, and N3, a new MFCE is created at these nodes, and the *Enroll* message is forwarded to the upstream interface. The upstream interface ($i = 0$) is set to the incoming state ($X_0 = 1$), and the downstream interface at which the *Enroll* is received ($i = \mathrm{R}$) is set to the outgoing state ($X_{\mathrm{R}} = -1$). Specifically, N4 creates $E_{\mathrm{G}}(\mathrm{N4}) = \{1, -1, 0\}$, N6 creates $E_{\mathrm{G}}(\mathrm{N6}) = \{1, 0, -1\}$, and N3 creates $E_{\mathrm{G}}(\mathrm{N3}) = \{1, -1, 0\}$. Meanwhile, MFCE for the G is already at N2, the MFCE is updated, and the *Enroll* message is not forwarded to the upstream interface. The downstream interface at which the *Enroll* is received ($i = \mathrm{R}$) is set to the outgoing state ($X_{\mathrm{R}} = -1$). Specifically, N2 updates the MFCE to $E_{\mathrm{G}}(\mathrm{N2}) = \{-1, -1, 1\}$.

R1 and R2 periodically send an *Enroll* message to N2 and N1, respectively. The MFCEs at nodes along the path can be refreshed. If no *Enroll* messages have been received at the interface ($i = \mathrm{R}$) for a predetermined period of time, the state of the interface is changed to zero ($X_{\mathrm{R}} = 0$).

Hence, an RPT whose RP is at N1 has been formed by R2, and an SPT whose SP is at B4 has been formed by R1.

3.2.1.2 Handover Procedures for Receiving Member's Movement

R2 moves from the B6 cell to the B7 cell (**Fig. 11**). Upon receiving the beacon signal from B7, R2 sends an *Enroll* message to B7. The *Enroll* includes multicast address G with which R2 is receiving multicast data. B7 and N7 forward the *Enroll* to their upstream interface. N7 does not have an MFCE for the same multicast address G, and it creates a new MFCE, $E_{\mathrm{G}}(\mathrm{N7}) = \{1, -1, 0\}$. In contrast, N3 receiving the *Enroll* already has an MFCE, $E_{\mathrm{G}}(\mathrm{N3}) = \{1, -1, 0\}$, so it updates the MFCE by setting the interface at which the *Enroll* is received ($i = 2$) to the outgoing state ($X_2 = -1$). Specifically, N3 updates the MFCE to $E_{\mathrm{G}}(\mathrm{N3}) = \{1, -1, -1\}$. N3 does not forward the *Enroll* to its upstream interface. Hence, the multicast data of the G are branched to N7 at N3. R2 receives multicast data from both B6 and B7. After receiving the multicast data from B7, R2 may send a *Leave* message to B6 to cut the unnecessary traffic. B6 forwards the *Leave* to its upstream interface. N6 receiving the *Leave* at the interface ($i = 2$) sets its interface state to $X_2 = 0$, i.e., it updates the MFCE to $E_{\mathrm{G}}(\mathrm{N6}) = \{1, 0, 0\}$. N6 has no outgoing interface for the G, so it removes the MFCE and forwards the *Leave* to its upstream interface. N3 receiving the *Leave* at the interface ($i = 1$) sets its interface state to $X_1 = 0$, *i.e.*, it updates the MFCE to $E_{\mathrm{G}}(\mathrm{N3}) = \{1, 0, -1\}$.

3.2.1.3 Handover Procedures for the Moving Source

The S moves from the B4 cell to the B5 cell (**Fig. 12**). Upon receiving the beacon signal from B5, the S sends a *Move* message to B5. The *Move* includes multicast address G with which the S is sending multicast data. The *Move* is forwarded from B5 to the current SP (i.e., B4) through N6, N3, N1, N2 and N5. N6 does not have the MFCE for the G, and it creates a new MFCE. The upstream interface ($i = 0$) is set to the outgoing state ($X_0 = -1$), and the downstream interface at which the *Move* is received ($i = 1$) is set to the incoming state ($X_1 = 1$). In other words, N6 creates $E_{\mathrm{G}}(\mathrm{N6}) = \{-1, 1, 0\}$. N3, N1, N2 and N5 receiving the *Move* already have the MFCE for the G. In this case, they create another MFCE with a 'transition' flag (here, expressed as $E'_g(n)$). First, they make a copy of the existing MFCE. In the copy, they change the interface state from incoming ($X_{\mathrm{A}} = 1$) to outgoing ($X_{\mathrm{A}} = -1$) and the interface state at which the *Move* is received ($i = \mathrm{R}$) to incoming ($X_{\mathrm{R}} = 1$). Thus, $E'_{\mathrm{G}}(\mathrm{N3}) = \{-1, 1, -1\}$, $E'_{\mathrm{G}}(\mathrm{N1}) = \{0, -1, 1\}$, $E'_{\mathrm{G}}(\mathrm{N2}) = \{1, -1, -1\}$, and $E'_{\mathrm{G}}(\mathrm{N5}) = \{1, 0, -1\}$ are created. These nodes forward the *Move* to the interface whose state is incoming in the existing MFCE.

Thus, another multicast data flow is originated from B5, and N3, N1, N2, and N5 receive the same multicast data at two different interfaces. The data packets are forwarded according to the MFCE whose incoming interface is the one where the packets are received. When the S has completely moved to B5 cell, it sends a *Complete* message to B5. The *Complete* including multicast address G is forwarded from B5 to the previous SP (i.e., B4). N3, N1, N2, and N5 receiving the *Complete* have two MFCEs, they remove the older one with the 'transition' flag cleared, and they clear the 'transition' flag of the newer one.

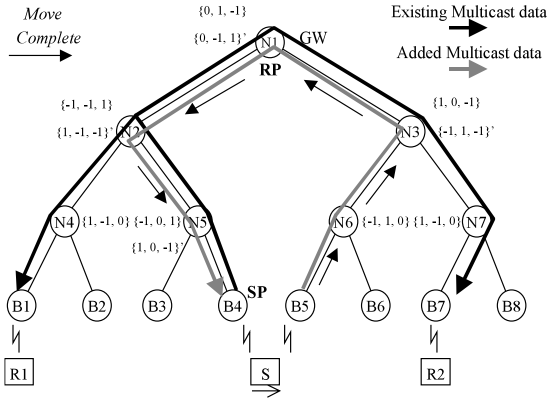

Fig. 12 SMM-A Proc. of handover of source: The S is moving to B5 cell. The S sends *Move* message to current SP (B4) by way of B5, N6, N3, N1, N2, N5 and B4. N3, N1, N2 and N5 already has MFCE for G, then, they create another MFCE with 'transition' flag; $E'_{\mathrm{G}}(\mathrm{N3}) = \{-1, 1, -1\}$, $E'_{\mathrm{G}}(\mathrm{N1}) = \{0, -1, 1\}$, $E'_{\mathrm{G}}(\mathrm{N2}) = \{1, -1, -1\}$, and $E'_{\mathrm{G}}(\mathrm{N5}) = \{1, 0, -1\}$. When the S has moved to B5 cell, it sends *Complete* message. N3, N1, N2 and N5 remove older MFCFs and clear the 'transition' flag.

Hence, the handover process for source mobility has been completed. After that, the S periodically sends a *Launch* message to N1 as long as it stays in the B5 cell.

The above SMM-A procedures can be described in a very simple manner. The following are the rules to create and update the MFCE in case of each received massage.

case: receive *Launch*
$\{X \in E_{\mathrm{G}}(n) | i = 0\} \rightarrow -1$
$\{X \in E_{\mathrm{G}}(n) | i = R\} \rightarrow 1$
if $E_{\mathrm{G}}(n)$ is new
$\{X \in E_{\mathrm{G}}(n) | i > 0, i \neq R\} \rightarrow 0$
case: receive *Enroll*
$\{X \in E_{\mathrm{G}}(n) | i = 0\} \rightarrow 1$
$\{X \in E_{\mathrm{G}}(n) | i = R\} \rightarrow -1$
if $E_{\mathrm{G}}(n)$ is new
$\{X \in E_{\mathrm{G}}(n) | i > 0, i \neq R\} \rightarrow 0$

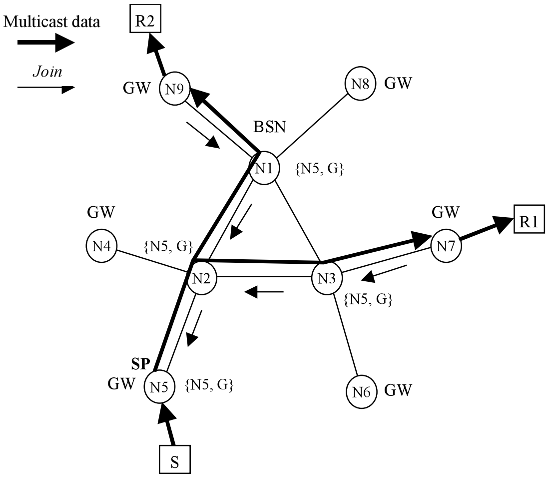

Fig. 13 SMM-C Proc. of source handover (1): The S is sending multicast data to N5. N5 periodically informs the BSN that it is the SP for G. The BSN floods *BSM* message to all GWs. N7 and N9, which R1 and R2 belong to, send a *Join* message to N5.

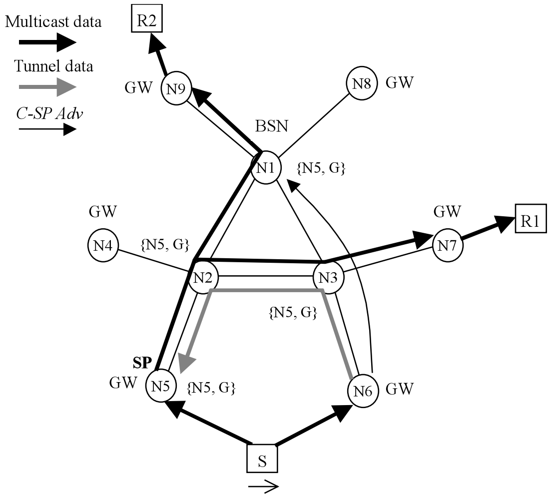

Fig. 14 SMM-C Proc. of source handover (2): The S is moving to the access network attached to N6. The S sends *Launch* message to the new GW, i.e., N6. N6 reports to the BSN that it becomes the SP for G and sets up a tunnel to N5 through which encapsulated multicast packets are forwarded by unicasting.

case: receiv *Move*
 if $E_{\mathrm{G}}(n)$ is new
  $\{X \in E_{\mathrm{G}}(n) | i = 0\} \rightarrow -1$
  $\{X \in E_{\mathrm{G}}(n) | i > 0, i \neq R\} \rightarrow 0$
 $\{X \in E_{\mathrm{G}}(n) | X_{\forall i} > 0\} \rightarrow -1$
 $\{X \in E_{\mathrm{G}}(n) | i = R\} \rightarrow 1$

3.2.2 SMM-C over PIM-SM in the Core Netowork

Here, the core network consists of switch nodes N1-N9. N1 is a bootstrap node (BSN) and N4-N9 are GWs to which the access networks are attached (**Fig. 13**). The GWs become the source points of the SPT in the core network. Now, the multicast source (S) is in the access network attached to N5, and the receiving members (R1 and R2) are in the ones attached to N7 and N9, respectively.

The S is sending multicast data packets with address G to N5. N5 periodically informs the BSN that it has become the source point (SP) for the G. The BSN floods this information in a *BSM* throughout the core network. N7 and N9, which R1 and R2 belong to, send a *Join* message to N5, and as a result, an SPT (SP is at N5) is formed. Multicast forwarding cache entries $\{n, g\}$ are created in N1, N2, N3, and N5. Specifically, $\{\mathrm{N5}, \mathrm{G}\}$ are created in these nodes. For details see the specifications of PIM-SM.

Now, the S moves to the access network at N6 (**Fig. 14**). Upon receiving the beacon signal of the N6's network, the S sends a *Launch* message to a new GW, i.e., N6. The *Launch* includes the current GW information, i.e., indicating that the current network is at N5. N6 receiving such kind of *Launch* message recognizes that this movement is handover between the networks. Then, N6 reports to the BSN that it has become the SP for the G, and sets up a tunnel to N5 where the encapsulated multicast packets are transparently forwarded by unicasting. This tunnel is maintained until a new SPT with an SP at N6 has been constructed.

The BSN sends a *BSM* broadcasting that N6 is the SP for the G throughout the core network. N7 and N9 receiving the *BSM* send a *Join* message to N6 (**Fig. 15**). As a result, a new SPT (SP at N6) is formed. New multicast forwarding cache entries, $\{\mathrm{N6}, \mathrm{G}\}$, are created in N1, N3, and N6. The old ones in N1, N2, N3, and N5, $\{\mathrm{N5}, \mathrm{G}\}$, will be removed as *join*s for N5 are not received for predetermined period of time.

Thus, in the core network, SMM-C controls the tunnel between the two SPs over the PIM-SM SPT mechanism.

4. Numerical Analyses

To evaluate our SMM, we numerically analyzed the overhead in reconstructing a multicast tree due to the source movement from the

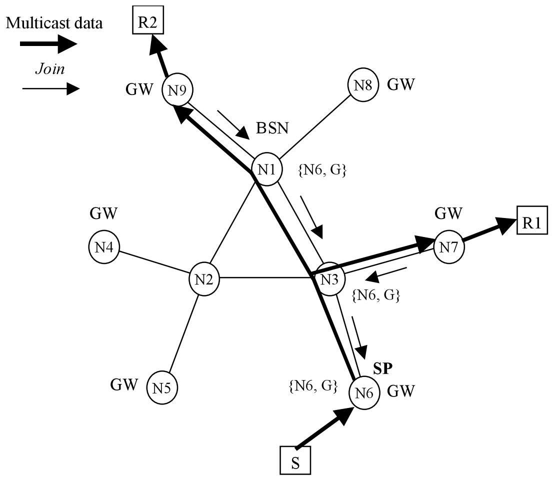

Fig. 15 SMM-C Proc. of source handover (3): The BSN floods *BSM* message to notify that N6 is the SP for G. N7 and N9 send a *Join* message to N6, thus a new SPT is constructed. N6 stops tunneling to N5.

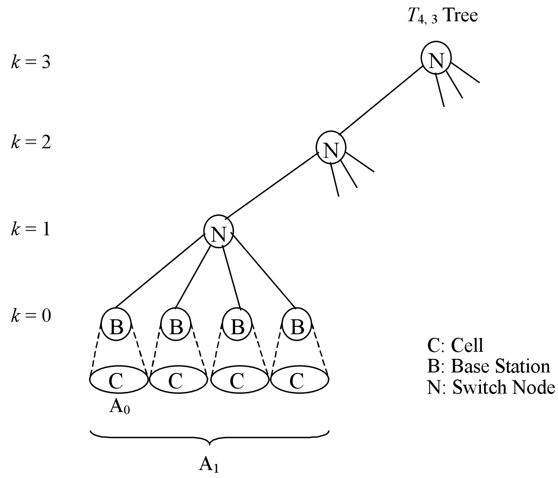

Fig. 16 Tree model: υ-ary tree of depth μ; $T_{\upsilon,\mu}$.

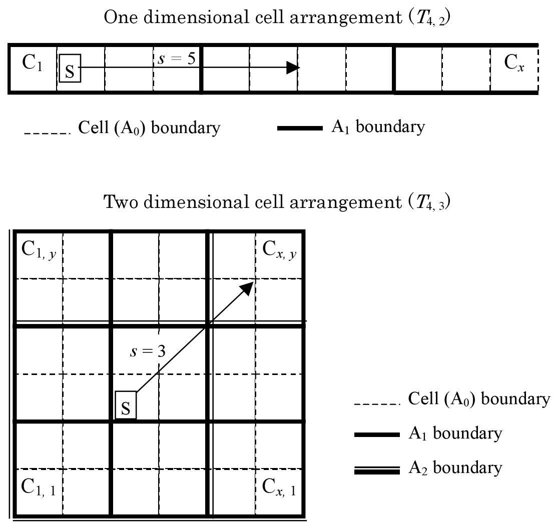

Fig. 17 Cell arrangement.

point of view of the number of nodes updating the multicast forwarding cache table. In the core network where SMM-C performs simple SPT construction, the effect of reducing the overhead in reconstructing tree is not expected. In the access network where SMM-A performs combination of SPT and RPT, the tree reconstruction overhead is greatly reduced. Here, we confirm the effectiveness in the access network by comparing SMM-A and PIM-SM.

Consider a υ-ary tree of depth μ; $T_{\upsilon,\mu}$. As shown in **Fig. 16**, we let k denote the layers of the tree (from the leaves of the tree, $k = 0, 1, \ldots, \mu$) and let A_k express the area (a set of cells) that the nodes of layer k cover (e.g., A_0 means a cell). The number of cells in A_k is υ^k.

Figure 17 depicts two ways of cell arrangement: cells are placed in-line (one-dimensional arrangement) and on a plane (two-dimensional arrangement). In the latter, we assume that the S moves to the destination along the shortest path, that is, when the S moves from $\mathrm{C}_{x,y}$ to $\mathrm{C}_{x+1,y+1}$, it does not pass $\mathrm{C}_{x+1,y}$ and $\mathrm{C}_{x,y+1}$. We express the speed of the source's movement (the number of cells that S passes per unit of time) as s. Here, we assume that the S does not move beyond the largest area A_μ in a unit of time (i.e., $s < \upsilon^\mu$ in one-dimensional arrangement and $s < \upsilon^{\mu/2}$ in two-dimensional arrangement).

In the one-dimensional cell arrangement, the number of areas, A_k that the S passes per unit of time, α_k, is

$$\alpha_k = \left\lceil \frac{s}{\upsilon^k} \right\rceil - 1 \tag{1}$$

The number of boundaries in each area A_k that the S crosses per unit of time, β_k, is

$$\beta_k = \alpha_k - \alpha_{k+1} = \left\lceil \frac{s}{\upsilon^k} \right\rceil - \left\lceil \frac{s}{\upsilon^{k+1}} \right\rceil \tag{2}$$

As shown in Fig. 17, for example, in case that s is 5 in one-dimensional cell arrangement ($T_{4,2}$), α_0, α_1 and α_2 are 4, 1 and 0 respectively, and β_0 and β_1 are 3 and 1 respectively. The number of all area boundaries that the S crosses per unit of time is

$$\sum_{k=0}^{\mu} \beta_k = (\alpha_0 - \alpha_1) + \ldots + (\alpha_{\mu-1} - \alpha_\mu)$$
$$= \alpha_0 - \alpha_\mu$$

$$= \left\lceil \frac{s}{\upsilon^0} \right\rceil - 1 - \left(\left\lceil \frac{s}{\upsilon^\mu} \right\rceil - 1 \right)$$
$$= \lceil s \rceil - 1 \qquad (3)$$

Then, the possibility that the boundary that the S crosses is the one of area A_k is expressed as follows.

$$P(A_k) = \frac{\beta_k}{\sum_{k=0}^{\mu} \beta_k} = \frac{\lceil s\upsilon^{-k} \rceil - \lceil s\upsilon^{-(k+1)} \rceil}{\lceil s \rceil - 1} \qquad (4)$$

The S sends a *Move* message to the previous SP every time it crosses any area boundary. When the S crosses the boundary of A_k, the number of nodes whose MFCE must be updated after receiving the *Move* is

$$\eta_{A_k} = 2k + 1 \qquad (5)$$

Then, the expected number of nodes that need updating the MFCE when the S crosses any area boundary is

$$\overline{\eta} = \sum_{k=0}^{\mu-1} \{\eta_{A_k} \times P(A_k)\}$$
$$= \sum_{k=0}^{\mu-1} \left\{ (2k+1) \times \frac{\lceil s\upsilon^{-k} \rceil - \lceil s\upsilon^{-(k+1)} \rceil}{\lceil s \rceil - 1} \right\} \qquad (6)$$

In the two-dimensional cell arrangement, the number of areas, A_k that the S passes per unit of time, α_k, and the number of boundaries in each area that the S crosses per unit of time, β_k, are

$$\alpha_k = \left\lceil \frac{s}{\sqrt{\upsilon^k}} \right\rceil - 1 \qquad (7)$$

$$\beta_k = \alpha_k - \alpha_{k+1} = \left\lceil \frac{s}{\upsilon^{k/2}} \right\rceil - \left\lceil \frac{s}{\upsilon^{(k+1)/2}} \right\rceil \qquad (8)$$

The number of area boundaries that the S crosses per unit of time is

$$\sum_{k=0}^{\mu} \beta_k = \alpha_0 - \alpha_\mu = \lceil s \rceil - 1 \qquad (9)$$

The possibility that the boundary that the S crosses is the one of area A_k is

$$P(A_k) = \frac{\beta_k}{\sum_{k=0}^{\mu} \beta_k}$$
$$= \frac{\lceil s\upsilon^{-k/2} \rceil - \lceil s\upsilon^{-(k+1)/2} \rceil}{\lceil s \rceil - 1} \qquad (10)$$

Then, the expected number of nodes updating the MFCE when the S crosses any area boundary is

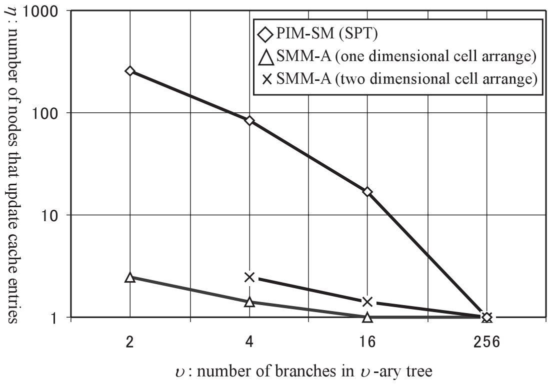

Fig. 18 Number of nodes that must update multicast forwarding cache.

$$\overline{\eta} = \sum_{k=0}^{\mu-1} \{\eta_{A_k} \times P(A_k)\}$$
$$= \sum_{k=0}^{\mu-1} \left\{ (2k+1) \times \frac{\lceil s\upsilon^{-k/2} \rceil - \lceil s\upsilon^{-(k+1)/2} \rceil}{\lceil s \rceil - 1} \right\} \qquad (11)$$

Here, we let d denote the cell size, and let D denote the size of the access network. The number of cells in the access network can be expressed as

$$D/d = \upsilon^\mu, \quad \mu = \log_\upsilon (D/d) \qquad (12)$$

Thus, Eqs. 6 and 11 are, respectively,

$$\overline{\eta} = \sum_{k=0}^{\log_\upsilon (D/d)-1} \left\{ (2k+1) \times \frac{\lceil s\upsilon^{-k} \rceil - \lceil s\upsilon^{-(k+1)} \rceil}{\lceil s \rceil - 1} \right\} \qquad (13)$$

$$\overline{\eta} = \sum_{k=0}^{\log_\upsilon (D/d)-1} \left\{ (2k+1) \times \frac{\lceil s\upsilon^{-k/2} \rceil - \lceil s\upsilon^{-(k+1)/2} \rceil}{\lceil s \rceil - 1} \right\} \qquad (14)$$

Figure 18 shows the estimated value, $\overline{\eta}$, for several values of υ in Eqs. 13 and 14 when $D/d = 256$. We compared this value with that in PIM-SM's SPT (assuming that all leaves of the tree have at least one receiving member, all switch nodes must update their forwarding cache entry when the source moves). Figure 18 shows that SMM-A significantly reduces the overhead in reconstructing the tree compared to that in PIM-SM's SPT. For example, when υ is 4, SMM-A makes the $\overline{\eta}$ decrease to 1.4% and 2.8% to PIM-SM in one-dimensional and two-dimensional cell arrangement, respectively. The value of $\overline{\eta}$ decreases as υ approaches D/d,

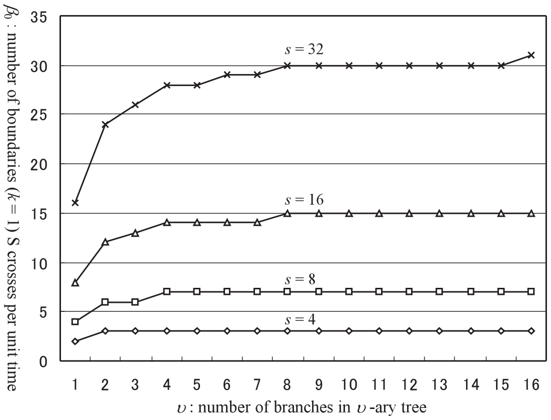

Fig. 19 Load of the node ($k = 1$).

i.e., as the depth of the tree μ becomes small. When υ is D/d, $\overline{\eta}$ is identical to 1 (there is no difference between SMM-A and PIM-SM).

However, we should note that the burden per node increases as μ becomes small. For any k, it follows that

$$\beta_k - \beta_{k+1} = \lceil s\upsilon^{-k} \rceil - \lceil s\upsilon^{-(k+2)} \rceil > 0 \quad (15)$$

For the number of area boundaries the S crosses, β_0 is the largest. That is, the load of the nodes is the largest at $k = 1$ (Eq. 16).

$$\beta_0 = \lceil s \rceil - \lceil s\upsilon^{-1} \rceil \quad (16)$$

Figure 19 shows the load of the nodes ($k = 1$). It indicates that the burden rises as the speed of the S increases, and that the burden decreases when υ is small. Hence, we should not make υ large and should design an optimal tree, $T_{\upsilon,\mu}$, considering the average speed of the S and the cell size in access network.

5. Conclusion

We presented a new multicast technique to support source mobility. Main contributions in this paper are the following:

We reviewed recent studies related to mobility support multicasting. They do not focus on the multicast source mobility and they are all based on the macro mobility protocol and cannot guarantee the consecutive communication. Regarding the problem of supporting source mobility with explicit multicasting, we pointed out that the overhead in reconstructing a multicast tree with SPT is high and that there is a backtrack route problem with RPT. We proposed a new multicast technique that combines both SPT and RPT in one multicast tree over a micro-mobility supported network, Cellular IP. This technique allows us to minimize the overhead in reconstructing multicast trees and guarantee consecutive communication during source movement. In addition, we provide network scalability by organizing hierarchical network structure and confining protocol uniqueness and topological constraints into access networks. We described these schemes and the details of protocols. We showed the effectiveness of our technique from the point of view of the number of nodes updating the multicast forwarding cache while reconstructing the tree.

Motivation for this SMM studies is in a background of recent enthusiastic works on intelligent transport systems (ITS) over the broadband Internet technologies [25),26)]. Our SMM was developed for the purpose of 'future-vision delivery service', which is an ITS-based innovative information delivery service. In the future-vision delivery service, users going to the same destination are grouped together, and earlier users on the same route provide their real-time video information to subsequent users via a real-time streaming technique. For these later users, such video information can be their future vision. For example, using a global positioning system (GPS) and a route guiding function (both are represented in car-navigation systems), users can exchange their destination and location information with other users in the same group through the network and can find earlier users and obtain their real-time video information, or provide later users with real-time video information. This service is of great importance to ITS users because it enables users at further points along the route to provide visual information they are facing, e.g., weather information or traffic reports, to later users. Therefore, our SMM technique, which enables high-speed mobile hosts to distribute video to multiple points, is essential to the future vision delivery service.

In the future, we will attempt to minimize topological constraints, for example, by developing a technique for the construction of a logical tree. This will enable us to use radio base stations and switch nodes in any topology. In addition, we plan to design an application layer protocol to implement the future-vision delivery service on the SMM.

References

1) Estrin, D., Farinacci, D., et al.: Protocol Independent Multicast-Sparse Mode, RFC2362

(1998).

2) Holbrook, H. and Cain, B.: Source-Specific Multicast, IETF Internet Draft (Nov. 2001).
3) Fenner, B., Handley, M., et al.: Protocol Independent Multicast — Sparse Mode (Revised) IETF Internet Draft (Dec. 2002).
4) Matsuzawa, T. and Nakayama, M.: A Proposal of Dynamic DR Election for Generating Multicast Tree of PIM-SM, *IEICE Journal*, Vol.J85-B, No.8 (Aug. 2002).
5) Fujii, M., Funabiki, N., et al.: A Proposal of an Optimal Path Selection Algorithm for Static and Mobile Multicast Routing Problems, *IEICE Journal*, Vol.J85-A, No.3 (Mar. 2002).
6) Ballardie, A.: Core Based Trees (CBT version 2) Multicast Routing, RFC2189 (1997).
7) Adams, A., Nicholas, J. and Siadak, W.: Protocol Independent Multicast — Dense Mode, IETF Internet Draft (Feb. 2003).
8) Moy, J.: Multicast Extensions to OSPF, RFC1584 (Mar. 1994).
9) Waitzman, D., Partridge, C. and Deering, S.E.: Distance Vector Multicast Routing Protocol, RFC1075 (Nov. 1988).
10) Perkins, C., et al.: IP Mobility Support, RFC2002 (Oct. 1996).
11) Perkins, C. and Johnson, D.B.: Route optimization in Mobile IP, IETF Internet Draft (Feb. 1999).
12) Koodli, R., et al.: Fast Handovers for Mobile IPv6, IETF Internet Draft (Sep. 2002).
13) Valko, A.G.: Cellular IP — A New Approach to Internet Host Mobility, *ACM Computer Communication Review* (Jan. 1999).
14) Zach, D., Gatzounas, D., Campbell, A., et al.: Cellular IPv6, IETF Internet Draft (Jul. 2001).
15) Gatzounas, D., Malataras, A., et al.: Route optimization for Cellular IP networks, *Proc. IPNC 2001*, France (May 2001).
16) Corson, S.M.: A Vision of an IP-based Cellular Network, C802.20-03/08 Contributions to IEEE 802.20 (Jan. 2003).
17) Mysore, J. and Bharghavan, V.: A New Multicasting-based Architecture for Internet Host Mobility, *Proc. ACM/IEEE MOBICOM'97*, Budapest, Hungary (Sep. 1997).
18) Helmy, A.: A Multicast-based Protocol for IP Mobility Support, *ACM SIGCOMM 2nd Intl' Workshop on Networked Group Communication*, Palo Alto, CA (Nov. 2000).
19) Helmy, A. and Jaseemuddin, M.: Efficient Micro-Mobility using Intra-domain Multicast-based Mechanisms (M&M), *ACM SIGCOMM Computer Communication Review* (Oct. 2002).
20) Harrison, T.G., Williamson, C.L., et al.: Mobile Multicast (MoM) Protocol: Multicast Support for Mobile Hosts, *Proc. ACM/IEEE MOBICOM'97*, Budapest, Hungary (Sep. 1997).
21) Wang, Y. and Chen, W.: Supporting IP Multicast for Mobile Hosts, *ACM/Kluwer Mobile Networks and Applications, Special Issue on Wireless Internet and Intranet Access*, Vol.6, No.1 (Jan. 2001).
22) Lin, C.R. and Wang, K.M.: Mobile Multicast Support in IP Networks, *Proc. IEEE INFOCOM'00*, Tel Aviv, Israel (Mar. 2000).
23) Sung, S., Shin, Y., et al.: Mobile Multicast Method using the Habitude of Movement, *16th Intl' Conf. on Information Network*, Chez, Korea (Jan. 2002).
24) Rolland, V., Luis, C. and Serge, F.: Mobility and Multicast: Protocol Design and Analysis, *17th Intl' Symposium on Computer and Information Science*, Orlando, FL (Oct. 2002).
25) Internet ITS Project: Building Internet ITS Infrastructure. http://www.internetits.org/en/infra2.html
26) Munaka, T., Yamamoto, T., et al.: Advanced Join Mechanism for Multicast Group Management in ITS Networks and Its Evaluation, *IPSJ Journal*, Vol.42, No.7 (Jul. 2001).

(Received May 15, 2003)
(Accepted December 2, 2003)

Katsuhiko Sato received the B.E. degree from Aoyama Gakuin University in 1992, and the M.E. degree from The University of Electro Communication in 2002. He has been working for Japan Radio Co., Ltd and developed Frame Relay router, ATM switch, 3G wireless base station, Multi-layer switch etc. Since 2002 he has been a visiting researcher at Communication Research Laboratory and working toward the Ph.D. in The University of Electro Communication. He is a member of IPSJ, IEEE Computer Society.

Michiaki Katsumoto received the B.S., M.S. and Ph.D. degrees from The University of Toyo in 1991, 1993 and 1996 respectively. He is working now Communications Research Laboratory. His research interests include next generation Internet applications and high-quality multimedia contents. He is a member of IPSJ, IEICE, IEEE Computer Society and ACM.

Tetsuya Miki received a B.E. degree from the University of Electro-Communications in 1965, and M.E. and Ph.D. degrees from Tohoku University in 1967 and 1970, respectively. He is currently a Professor at the Department of Information and Communication Engineering, the University of Electro-Communications. His research involves optical and wireless access systems, photonic networks, multimedia information communications, and network architectures. He joined the Electrical Communication Laboratories of NTT in 1970, where he was engaged in the research and development of high-speed digital transmission systems using coaxial cable, optical transmission systems, optical access networks, ATM and multimedia transport networks, network operation systems and network architecture. He was the Executive Manager of the NTT Optical Network Systems Laboratories in 1992–1995. He received the IEICE Achievement Award in 1978 for his contribution to the early development of optical transmission systems. He is a Fellow of IEEE, and was Vice-President of IEEE Communications Society during 1998 and 1999. He is also a Fellow of IEICE, and is currently Vice-President of IEICE.