Reliable Application Layer Multicast Over Combined Wired and Wireless Networks

Masahiro Kobayashi, *Member, IEEE*, Hidehisa Nakayama, *Member, IEEE*, Nirwan Ansari, *Fellow, IEEE*, and Nei Kato, *Senior Member, IEEE*

***Abstract*—During the last several years, the Internet has evolved from a wired infrastructure to a hybrid of wired and wireless domains by spreading worldwide interoperability for microwave access (WiMAX), Wi-Fi, and cellular networks. Therefore, there is a growing need to facilitate reliable content delivery over such heterogeneous networks. On the other hand, application layer multicast (ALM) has become a promising approach for streaming media content from a server to a large number of interested nodes. ALM nodes construct a multicast tree and deliver the stream through this tree. However, if a node leaves, it cannot deliver the stream to its descendant nodes. In this case, quality-of-service (QoS) is compromised dramatically. Especially, this problem is exacerbated in wireless networks because of packet errors and handovers. In order to cope with this problem, multiple-tree multicasts have been proposed. However, existing methods fail to deliver contents reliably in combined wired and wireless networks. In this paper, we propose a method to ensure the robustness of node departure, while meeting various bandwidth constraints by using layered multiple description coding (LMDC). Finally, we evaluate the proposed method via extensive simulations by using the network simulator (ns-2). By comparing our proposed method with the existing ones, we demonstrate that our method provides better performance in terms of total throughput, relative delay penalty (RDP), and relative delay variation (RDV). The results indicate that our approach is a more reliable content delivery system when compared with contemporary methods in the context of heterogeneous networks containing wired and wireless environments.**

***Index Terms*—Application layer multicast, heterogeneous networks, layered multiple description coding, wired/wireless networks.**

I. Introduction

With the widespread availability of inexpensive broadband Internet connections for home users, many content delivery applications have now become practical. The choice of broadband Internet access is usually restricted to T1, digital subscriber line (DSL), cable-modem, or passive optical network (PON)-based wired connections. On the other hand, worldwide interoperability for microwave access (WiMAX) [1], Wi-Fi [2], and cellular broadband wireless access [3], [4] are constituting the next generation wireless systems. These technologies provide high throughput broadband connections over long distance, and are expected to be the last mile wireless broadband access as an alternative to the wired connection. So, there is a growing need to facilitate efficient content delivery over combined wired and wireless networks. For this reason, recently, multimedia streaming services, such as Internet Protocol Television (IPTV) [5], distance learning, video conferencing, and news broadcasting, have been a focus of constant attention. To tackle the scalability issue of the unicast-based media streaming architectures, tree-based solutions have been proposed, such as IP multicast [6] and application layer multicast (ALM) [7]. In IP multicast, the server and the end-nodes act as the root and as the leaf nodes, respectively. The intermediate nodes are routers that provide point-to-multipoint transmission through packet replication. The IP multicast approach has not been readily deployed because it requires routers with special capability. On the other hand, in ALM systems, the multicast tree is rooted at the media server, and participating nodes join the tree as interior and leaf nodes. An interior node is responsible for forwarding data from its parent node to its children through unicast. Additionally, although IP multicast requires special routers, ALM does not. In this paper, we focus on designing an ALM protocol for heterogeneous networks containing wired and wireless environments. However, duplication and relay of packets performed by the end-nodes are generally less reliable than those performed by routers. Therefore, ALM needs to address the following issues in combined wired and wireless networks.

First, since nodes are free to join and leave the service at any time, the number of “currently active” nodes is unpredictable. The departure of interior nodes in the multicast tree severely affects the descendant nodes, and thus a multicast service is greatly susceptible to node dynamics. Therefore, if the nodes often leave, quality-of-service (QoS) of the stream is degraded. This problem is especially severe in wireless environments, in which streaming packets and control messages often fail because of packet errors and handovers. If a handover occurs, the wireless node cannot communicate to other nodes for a few seconds. Therefore, wireless nodes are not as reliable as wired nodes. In ALM, it is indeed critical to improve “the robustness of node departure”. Here, the robustness of node departure implies the ability of a node to continue receiving data streams in spite of simultaneous departure of several of its descendant nodes.

Manuscript received March 12, 2009; revised August 14, 2009. First published September 22, 2009; current version published November 18, 2009. This work was supported in part through the strategic international cooperative program between the Japan Science and Technology Agency (JST) and the National Science Foundation Cyber Trust (NSF) under grant no. 0726549. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Zhihai (Henry) He.

M. Kobayashi was with the Graduate School of Information Sciences, Tohoku University, Sendai 980-579, Japan. He is now with the NTT Service Integration Laboratories, Tokyo 180-8585, Japan (e-mail: kobayashi.masahiro@lab.ntt.co.jp).

H. Nakayama is with the Department of Electronics and Intelligent Systems Faculty of Engineering, Tohoku Institute of Technology, Sendai 982-8577, Japan (e-mail: hidehisa@m.ieice.org).

N. Ansari is with the Advanced Networking Laboratory, Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102 USA (e-mail: Nirwan.Ansari@njit.edu).

N. Kato is with the Graduate School of Information Sciences, Tohoku University, Sendai 980-8579, Japan (e-mail: kato@it.ecei.tohoku.ac.jp).

Digital Object Identifier 10.1109/TMM.2009.2032692

TABLE I
COMPARISON OF NETWORK TECHNOLOGIES

Type of Network	Technology	Maximum Bandwidth	Packet Error Rate
Wired Networks	T1, DSL, cable-modem, PON	100 Mbps	Low
Wireless Local Area Networks (WLANs)	Wi-Fi (IEEE 802.11)	54 Mbps	High
Wireless Metropolitan Area Networks (WMANs)	WiMAX (IEEE 802.16)	75 Mbps	High
Wireless Wide Area Networks (WWANs)	Cellular Networks (3GPP, 3GPP2)	3 Mbps	Very High

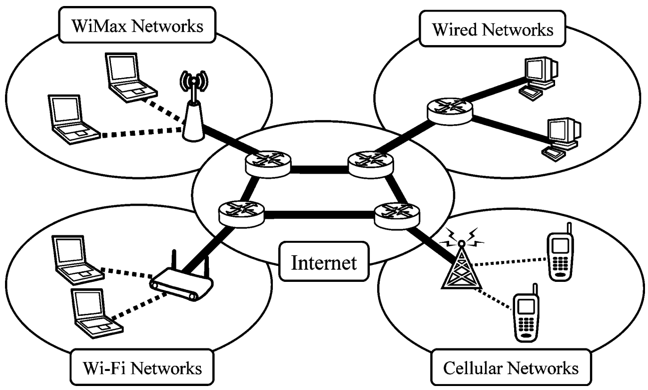

Fig. 1. Heterogeneous networks containing wired and wireless environments.

Second, the propagation delay from the source node to the destination node may be excessive because the data are forwarded by a number of interior nodes along the multicast tree. Since end-nodes in ALM do not have the routing information available to the routers, the multicast trees built in ALM suffer from the increase of propagation delays and the inefficient usage of bandwidth as compared to IP multicast. Third, in the Internet, each user's available bandwidth is highly heterogeneous because its last mile connection is mixed with wired and wireless portions. Therefore, ALM systems need to adapt to various bandwidth constraints.

In order to improve the robustness of node departure, multiple-tree multicast was proposed [8]–[12]. This method splits the original data stream into several descriptions with multiple description coding (MDC) [13], [14], and delivers the descriptions by using multiple multicast trees in parallel. In MDC, we can playback the contents by receiving one of the descriptions. Higher quality can be achieved by obtaining more descriptions. However, existing methods are difficult to deliver contents reliably in combined wired and wireless networks.

In order to address the above issues, we propose a method to ensure the robustness of node departure, while meeting bandwidth constraints by using layered multiple description coding (LMDC) [15]. LMDC has been proposed as a means of combining MDC and layered coding [16], [17] for emerging multicast and overlay audio/video streaming applications. To exploit the benefits of both MDC and layered coding, LMDC splits the descriptions which are divided by MDC into several layers. When two layers are used, the low bandwidth nodes receive only a base layer, while the high bandwidth nodes additionally receive an enhancement layer. In the proposed method, we construct multiple-tree based on arrangement graph (AG) [18], [19] like topology-aware hierarchical arrangement graph (THAG) [11] and network-aware hierarchical arrangement graph (NHAG) [12]. Each node calculates the requested layer based on the available bandwidth and searches for the joinable AG based on this information. Furthermore, each AG calculates the AG layer according to the joining nodes' requested layers. When an AG's AG layer is l, its parent-AG transfers the l layers of descriptions to the AG. By doing so, each node can receive the appropriate number of layers of descriptions without degrading the robustness of node departure.

Finally, our simulation results by using network simulator (ns-2) [20] have demonstrated that our approach provides better performance in terms of total throughput, relative delay penalty (RDP), and relative delay variation (RDV) than those of existing approaches [10]–[12]. The results indicate that our approach is more reliable in combined wired and wireless networks.

The rest of the paper is organized as follows. In Section II, we provide an overview of existing access network technologies and the conventional ALM tree construction methods. Section III describes THAG and NHAG. Our proposed ALM method is described in Section IV. In Section V, we present our simulation results and performance comparisons. Concluding remarks are given in Section VI.

II. RELATED WORK

In this section, we provide an overview of existing access network technologies (Section II-A) which use the current Internet environments, and review the conventional ALM protocols (Section II-B).

A. Current Access Network Technologies

During the last several years, the Internet has evolved from a wired infrastructure to a hybrid of wired and wireless domains, as shown in Fig. 1 where dashed and solid lines indicate wireless and wired links, respectively. The wired access networks consist of T1, DSL, cable-modem, and PON-based wired connections, and the wireless access networks comprise WiMAX (IEEE 802.16) [1], Wi-Fi (IEEE 802.11) [2], and cellular broadband wireless access (3 GPP: 3rd Generation Partnership Project [3], 3 GPP2 [4]). The properties of each network technology are summarized in Table I.

The wired connections based on T1, DSL, cable-modem, or PON have high bandwidth and low packet error rates. Many broadband users have asymmetric connections (whereby download rate $\gg$ upload rate). However, ALM systems generally need more upload bandwidth capacity than that of the download. Therefore, most DSL nodes would easily be able to receive but not forward all the received descriptions. In an academic or business environments, symmetric connections are more common. Such nodes can often receive and forward several times more descriptions than those in asymmetric connections. Thus, in the

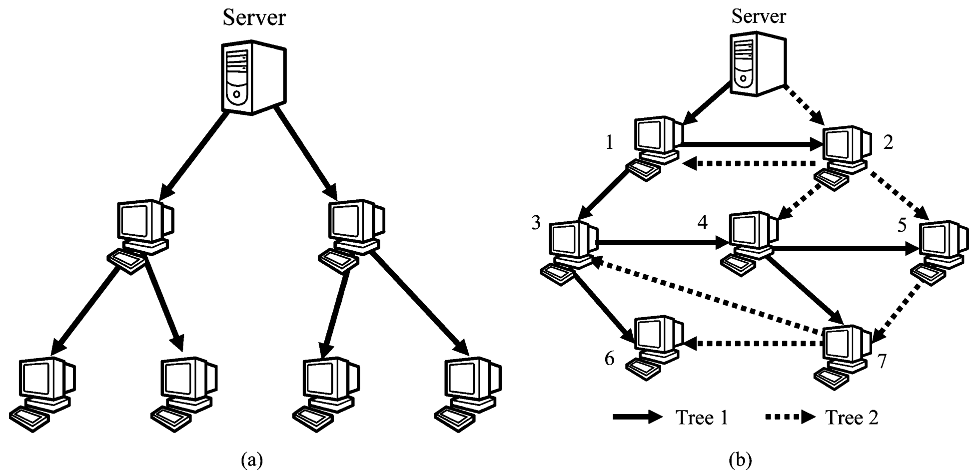

Fig. 2. Tree-based ALM. (a) Single-tree multicast. (b) Multiple-tree multicast.

wired networks, we consider only the upload bandwidth constraints. On the other hand, for wireless technologies, WiMAX can deliver a theoretical maximum upload and download data rate of 75 Mbps on a single channel. 3G cellular networks' maximum data rate is about 3 Mbps, and Wi-Fi's is 54 Mbps for the currently adopted IEEE 802.11g version. However, the upload/download bandwidth attenuates over distance. Therefore, we consider not only the upload bandwidth constraints but also the download bandwidth constraints. Furthermore, packet error rates of those wireless technologies are much higher than those of the wired technologies, and packet errors multiply over distance.

As described above, in the Internet, each user's available upload and download bandwidths are highly heterogeneous. Therefore, we need to propose a content delivery system which is tailored for combined wired and wireless networks.

B. Application Layer Multicast (ALM)

In ALM, an overlay network is constructed at the application layer independently from the network layer by allowing each end-node to forward the streaming data. A multicast tree is created by having the end-nodes (which are responsible for duplication of the received media stream) acting as a branch. The stream delivery by ALM flows through the multicast tree, and the root acts as the source node, which owns and disseminates the media data. A significant amount of research efforts has been directed toward QoS improvement. Existing works can be roughly classified into single-tree and multiple-tree multicast schemes.

Many of the existing works have advocated on building a single-data distribution tree rooted at the media data originator (the sender), as shown in Fig. 2(a). Therefore, each receiver has only one path from the sender along the tree. So far, Yoid [21], SpreadIt [22], ALMI [23], HBM [24], NICE [25], ZIGZAG [26], and Scribe [27] have been proposed. These methods use only one multicast tree to deliver a stream. Therefore, if the stream is not delivered due to node leaving, QoS of the stream will degrade dramatically.

To utilize path diversity for improving reliability and QoS of streaming, multiple-tree multicast schemes have been proposed. Multiple-tree multicast constructs multiple paths between the root and each receiver, as shown in Fig. 2(b), and delivers descriptions by using MDC [13], [14]. MDC is able to split the original streaming media into several descriptions. We can playback the contents by receiving one of the split descriptions, and higher quality can be achieved by obtaining more descriptions. So far, CoopNet [8], [9], SplitStream [10], THAG [11], and NHAG [12] have been proposed.

CoopNet proposes a centralized algorithm to facilitate deployment of multiple-multicast trees from different sources, and does not have explicit mechanisms to maximize bandwidth. In contrast, SplitStream has proposed a decentralized algorithm to construct a forest of multicast trees from a single source. SplitStream is based on Scribe, a tree-based multicast algorithm based on structured overlay networks. Both CoopNet and SplitStream cannot ensure the construction of node-disjoint multicast trees, implying that a node can be an interior node in several multicast trees and its departure will prevent the descendant nodes from receiving descriptions. In THAG and NHAG, node-disjoint multicast tree construction is ensured. Construction of node-disjoint multicast trees guarantees that the departure of any node will only affect data delivery in at most one multicast tree. In these methods, all the participating nodes are divided into a number of arrangement graphs, and several node-disjoint multicast trees are embedded in each AG. However, in THAG, it is difficult to deliver descriptions to meet various bandwidth constraints imposed by the network. In order to cope with this problem, we previously proposed NHAG to ensure QoS of the received stream by dynamically changing the AG size according to the available bandwidth and by preventing nodes from disabling participation in forwarding the stream. However, in NHAG, many nodes cannot receive all the descriptions delivered from the source because each node is delivered the number of descriptions based on its available bandwidth. Therefore, NHAG has the issue of degraded robustness of node departure. This problem is more serious in wireless environments.

Existing methods can hardly deliver contents reliably in combined wired and wireless networks.

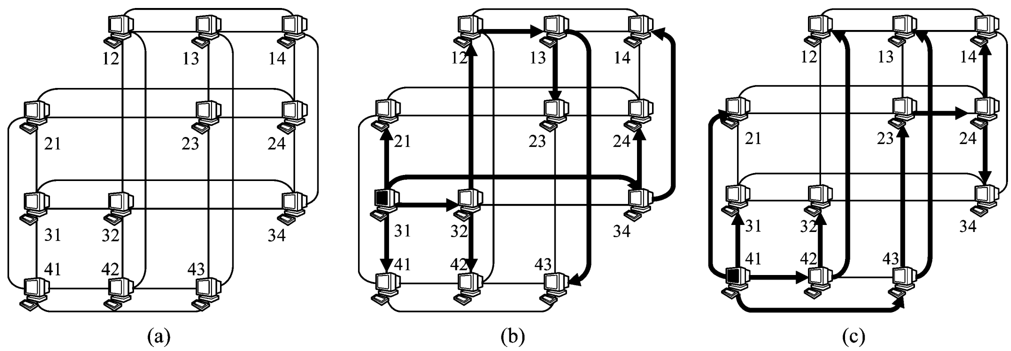

Fig. 3. Tree structure based on arrangement graph. (a) Arrangement graph with size of 4. (b) Multicast tree rooted at node 31. (c) Multicast tree rooted at node 41.

III. Multiple-Tree Multicasts With Arrangement Graphs

In this section, we simply provide an overview of THAG [11] and NHAG [12] in Sections III-A and III-B, respectively. These methods are proposed to construct multiple node-disjoint multicast trees. The node-disjoint trees can be constructed by making a node which is a parent node in a specific tree into a leaf node of all other trees, as shown in Fig. 2(b). For example, node 1 is an interior node in tree 1, but is a leaf node in tree 2. By doing so, even when a node cannot receive the description due to the departure of a node in its upper position, the descendant node can still receive the descriptions from other trees. In THAG and NHAG, participating nodes are grouped into a number of AGs [18], [19]. In each AG, several node-disjoint multicast trees are embedded. Once embedded, they will assemble the AGs into a tree-like hierarchical structure.

A. THAG

THAG uses an AG to construct node-disjoint trees. An AG is an undirected graph and has desired properties for overlay topology, such as symmetric vertex, symmetric edge, strong resilience, and maximal fault-tolerance [19]. An AG is denoted by $A_{s,k}$, and specified by integers s and k $(1 \leq k \leq s)$. Denote $\langle s \rangle = \{1, 2, \ldots, s\}$. There are k symbols denoted as $X = x_1 x_2 \ldots x_k$; x_i refers to the ith element of X. $A_{s,k}$, introduced in [18], is an undirected graph (V, E) defined in the equation at the bottom of the page. From this definition, X and Y differ in one position only. Therefore, an edge of $A_{s,k}$ connects neighboring nodes, which differ in exactly one of their k positions from each other.

THAG constructs $(s-2)$ node-disjoint trees from $A_{s,2}$. In this paper, we refer to s as the AG size. Generally, in an AG with size s, $(s\,(s-1))$ number of nodes can participate in the AG.Fig. 3(a) shows an example when the AG size is 4, while Fig. 3(b) and (c) shows examples of trees based on a size 4 AG. In these figures, the root of each tree is node $i1$ $(3 \leq i \leq s)$, and the two trees which have the root nodes 31 and 41, respectively, have been constructed. In these trees, we can see that the node, which is the parent node in one tree, is the leaf node in another tree. Therefore, these two trees are node-disjoint. Node 21 is a leaf node in all the multicast trees causing it to be selected as the AG entrance, and it will maintain the current states of all its AG members.

Furthermore, more nodes can join the AG in a hierarchical manner. When the number of nodes participating in an AG reaches the limitation, that AG is made into a parent-AG, which can spawn new child-AGs. As shown in Fig. 4, the parent-AG derives the child-AGs after it is filled. Nodes 32 and 42 in the parent-AG serve as the source nodes, which forward corresponding descriptions to child-AG 1. Similarly, nodes 13 and 43 in the parent-AG serve as the source nodes for child-AG 2. Suppose that a node acts as an interior node of one tree in a parent-AG, then it will also act as the source node for a child-AG. In this way, the node-disjointness of multicast trees is preserved. In general, a column of nodes $\{ij \mid i = 1, \ldots, s,\ i \neq j,\ i \neq (j+1) \bmod 2 + 1\}(1 < j < s)$ deliver descriptions to child-AGs. A column of nodes in the parent-AG providing data to its child-AG is referred to as the AG source. The descriptions which are delivered to the parent-AG are also delivered to child-AG 1 and child-AG 2 as well. In other words, since the delivery of descriptions is performed based on the delivery tree constructed from the AG, we can easily achieve a large-scale delivery network in a hierarchical manner.

B. NHAG

In THAG, we can create several node-disjoint multicast trees from AGs. The descriptions are delivered by using these created multicast trees in parallel. However, to deliver the descriptions stably, the node at the AG source can have no more than $(2\,(s-2))$ child nodes at the maximum, and other nodes in the

$$
\begin{aligned}
V &= \{X = x_1 x_2 \ldots x_k | x_i \in \langle s \rangle \quad \text{and} \quad x_i \neq x_j \quad \text{for} \quad i \neq j\} \\
E &= \{(X, Y) | X, Y \in V \quad \text{and for some} \quad i \in \langle k \rangle \\
&\qquad x_i \neq y_i \quad \text{and} \quad x_j = y_j \quad \text{for} \quad j \neq i\}
\end{aligned}
$$

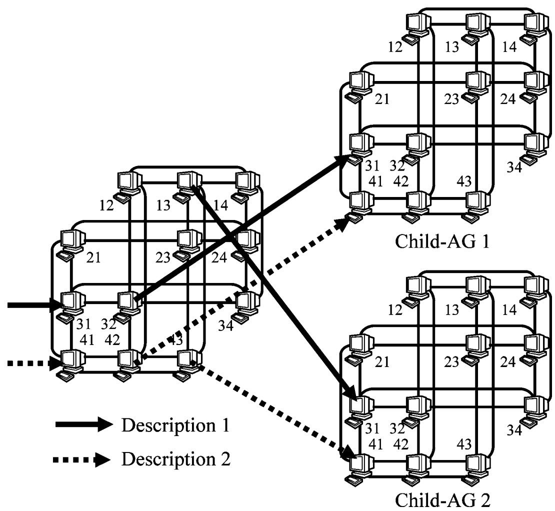

Fig. 4. Hierarchical AG in THAG.

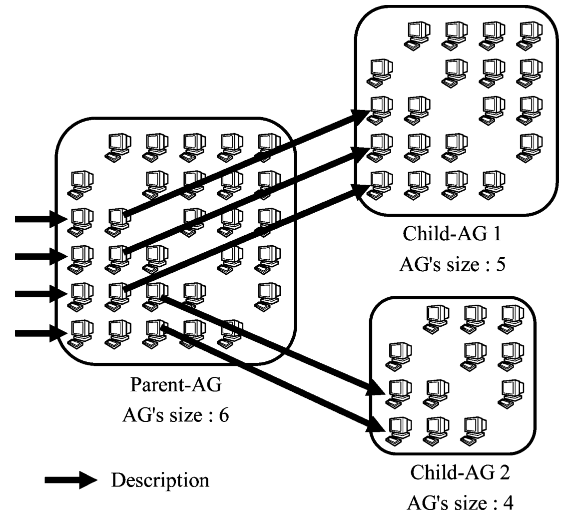

Fig. 5. Hierarchical AG in NHAG.

AG have to deliver descriptions to child nodes in the AG and the root nodes in child-AGs. Furthermore, there is a chance that every node may become an AG source. Therefore, the minimum upload bandwidth needed for streaming delivery is determined based on the AG size s and each description's rate r, which is $(2\,(s-2)\,r)$. Nodes, which connect to a link with bandwidth less than this amount, may not be able to send all the descriptions. In a conventional THAG, the required bandwidth is not taken into account because it is assumed that the AG size is fixed and all the descriptions consume the same amount of bandwidth in each link. However, in a real network, each node's available bandwidth is highly heterogeneous due to different link technologies and varying willingness to contribute (as described in Section II-A). Thus, it is difficult for THAG to deliver descriptions to meet various bandwidth constraints imposed by the network.

To cope with this problem, we proposed NHAG that can fit the various bandwidth constraints by changing the size of an AG dynamically. In NHAG, we do so for each AG, and hence the AG size is different between a parent-AG and a child-AG. If the AG size is s, the maximum number of descriptions that the AG can forward is $(s-2)$. If the AG size s becomes small, the number of descriptions which can be delivered in the AG decreases. Therefore, in this case, the parent-AG transfers $(s-2)$ descriptions to its child-AG with AG size s, as shown in Fig. 5. For this reason, NHAG should locate the AG that has the largest size in the highest position and the AG that has the smallest size in the lowest position to improve stream delivery efficiency. To realize this AG structure, each node calculates the requested size based on the available bandwidth, and searches for the AG with the appropriate size based on this information.

By following these procedures, each node can receive the number of descriptions based on its upload bandwidth.

IV. Proposed Method

In NHAG, each node receives the number of descriptions based on the available upload bandwidth. However, the robustness of node departure are degraded because many nodes cannot receive all the descriptions delivered from the source. For example, we assume that the number of trees is 4, and node 1 joins trees 1 and 2. Node 1 is the descendant node of nodes 2 and 3 in trees 1 and 2, respectively. When nodes 2 and 3 leave at the same time, node 1 cannot receive any description. If node 1 joins all the trees, the node can receive the descriptions from trees 3 and 4. Therefore, if the nodes often leave, QoS of the stream is degraded. Therefore, NHAG has an issue that the robustness of node departure is degraded. This problem is more serious in wireless environments (e.g., WiMAX, Wi-Fi, and cellular networks). Therefore, THAG and NHAG are difficult to deliver contents reliably in combined wired and wireless networks.

We propose a method to ensure the robustness of node departure, while meeting the various bandwidth constraints by using LMDC [15] (its properties are described in Section IV-A). LMDC splits the descriptions which are divided by MDC into several layers. In the proposed method, each node calculates the requested layer based on the available upload and download bandwidths. In Section IV-B, we discuss the required layer. In addition, each AG calculates the AG layer based on joining nodes' requested layers based on its available bandwidth. The AG layer is the maximum layer of descriptions required for all nodes which participated in the AG to stably receive and deliver. The procedure to calculate the AG layer will be described in Section IV-E. In the proposed method, the size of each AG remains constant, and each AG delivers all the number of descriptions to its child-AGs.

Moreover, when an AG's AG layer is l, its parent-AG transfers the l layers of descriptions to the AG, as shown in Fig. 6. By doing so, each node can receive the appropriate number of layers of descriptions without decreasing the number of descriptions. That is to say, the low bandwidth nodes receive only a base layer, while the high bandwidth nodes additionally receive enhancement layers based on their available bandwidth. Therefore, the proposed method improves the robustness of node departure as compared to that of NHAG. To realize this structure, we modify/tailor the node joining and leaving procedures based on the requested layer that will be described in Sections IV-C and IV-D, respectively.

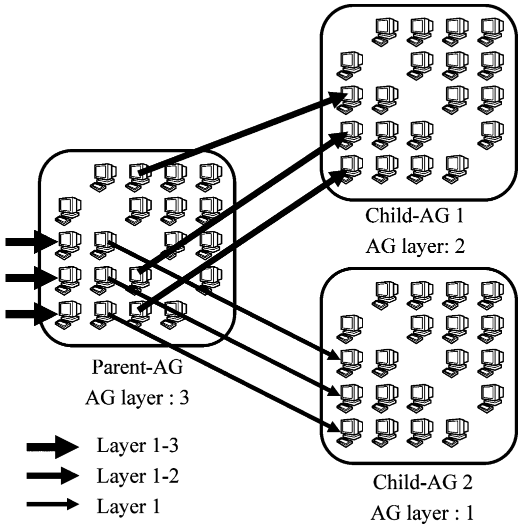

Fig. 6. Hierarchical AG in NHAG+.

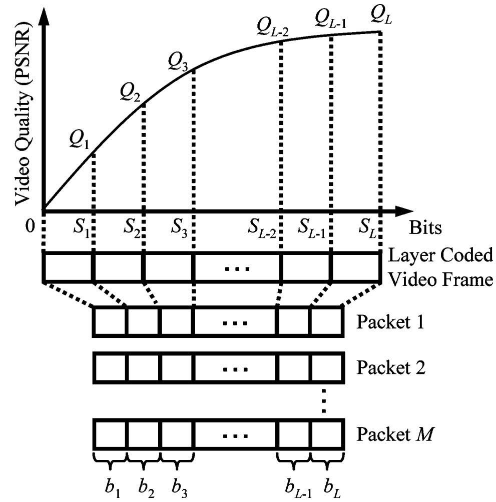

Fig. 7. Layered multiple description coding [28].

We call the proposed method the network-aware hierarchical arrangement graph plus (NHAG+). Furthermore, NHAG+ does not increase control overhead as compared to THAG and NHAG.

A. *Layered Multiple Description Coding*

LMDC [15] has been proposed as a means of combining MDC [13], [14] and layered coding [16], [17] for emerging multicast and overlay audio/video streaming applications. More specifically, multiple descriptions are spread across multiple packets via MDC, and transmitted to a collection of nodes, thereby reducing the packet errors due to network congestion or the departure of unreliable nodes. Moreover, by using layered coding, multimedia data can be encoded into different quality levels, so that nodes can play the best possible video/audio quality level according to their capabilities, such as screen resolution and link bandwidth. By combining MDC and layered coding, the LMDC scheme spreads the layered data across multiple packets with multiple descriptions. Then, nodes can play the layered data as long as the required number of descriptions are received successfully. Of course, the more descriptions a node receives, the better the reconstructed data quality will be. In practice, the LMDC scheme is usually implemented in conjunction with unequal erasure protection (UEP) [16], which provides different levels of erasure protection to the LMDC blocks by adding different amounts of redundancy.

Fig. 7 illustrates the LMDC scheme for video transfer applications. From the figure, we observe that the quality of a layered video frame improves as the size of the collected video bit stream increases. More specifically, if the size of the layered video frames is S bytes, one can split it into L equal-sized pieces and reconstruct it into Q_l quality levels by using the first l out of the L pieces [i.e., the required bit stream size for reconstructing Q_l level frame is $(S_l = l \times S/L)$]. Each layered video frame is then split among M description packets $(M \leq L)$ with unequal erasure protection on each piece of frame. In this paper, we assume that each layered video frame is split into equal-sized descriptions. Therefore, the size of the lth coded frame piece b_l is

$$b_l = \frac{S_l - S_{l-1}}{M} = \frac{S}{L \times M}. \quad (1)$$

From (1), the rate of one layer of each description, R_l, is calculated as follows:

$$R_l = \frac{R}{L \times M} \quad (2)$$

where R is the rate of the media data that is delivered by ALM. Note that for playback of a video using LMDC, descriptions can be received in a random order but layers have to be received in the ascending order. In the proposed method, we assign each description to a singular tree, i.e., the number of descriptions is equivalent to the number of trees. For distinguishing different layers in a description, the number of layers is marked in the packet.

B. *Requested Layer*

In NHAG+, each node calculates the requested layer rl which is the largest layer required to stably receive and deliver descriptions. In NHAG, each node calculates the requested size based on the node's available upload bandwidth. However, NHAG does not consider the download bandwidth constraints because it assumes wired network environments, where the download bandwidth is far larger than the upload bandwidth. On the other hand, in wireless networks, the upload and download bandwidths attenuate over distance. So, it is difficult for NHAG to meet the various bandwidth constraints in combined wired and wireless networks. In addition, if join/leave events are sparse with highly fluctuating network dynamics, the nodes that initially (at join time) had large bandwidth and were assigned a large number of child nodes may struggle when congestion occurs in their links [12]. Therefore, NHAG+ calculates the requested layer rl by using the available upload and download bandwidths, which are estimated in real time as described below.

First, we focus on upload bandwidth constraints. The node, which places the nth column in the AG, has a maximum of C_n child-nodes. C_n is calculated as follows [11]:

$$C_n = \begin{cases} 2(s-2), & \text{for} \quad n = 1 \\ s - 2 + C_{\mathrm{AG}}, & \text{for} \quad 2 \leq n \leq (s-1) \\ s-2, & \text{for} \quad n = s \end{cases} \tag{3}$$

where s is the AG size and C_{AG} is the number of child-AGs to which the node delivers the descriptions. In NHAG+, every AG size s is constant.

If the node transfers l layers of descriptions, the minimum upload bandwidth needed for streaming delivery is $(C_n \times R_l \times l = C_n \times R/L \times M \times l)$. Here, R, L, and M are the rate of media data, the number of layers, and the number of descriptions, respectively. Therefore, the requested layer rl_{down} satisfies (4):

$$C_n \times \frac{R}{L \times M} \times rl_{\mathrm{up}} = \mathrm{BW}_{\mathrm{up}}. \tag{4}$$

Here, $\mathrm{BW}_{\mathrm{up}}$ indicates the available upload bandwidth for each node. By solving this equation for rl_{up}, we have the following:

$$rl_{\mathrm{up}} = \left\lfloor \frac{\mathrm{BW}_{\mathrm{up}} \times L \times M}{C_n \times R} \right\rfloor. \tag{5}$$

Next, we focus on download bandwidth constraints. If the node receives l layers of descriptions, the requested download bandwidth is $(R/L \times M \times (s-2) \times l)$. Here, the number of descriptions M equals $(s-2)$. So, the requested layer rl_{down} satisfies (6):

$$\begin{aligned} \mathrm{BW}_{\mathrm{down}} &= \frac{R}{L \times M} \times (s-2) \times rl_{\mathrm{down}} \\ &= \frac{R}{L} \times rl_{\mathrm{down}} \end{aligned} \tag{6}$$

where $\mathrm{BW}_{\mathrm{down}}$ indicates the available download bandwidth for each node. By solving this equation for rl_{down}, we have the following:

$$rl_{\mathrm{down}} = \frac{\mathrm{BW}_{\mathrm{down}} \times L}{R}. \tag{7}$$

We can use bandwidth estimation technologies, such as initial gap increasing (IGI) [29], self-loading periodic streams (SLoPS) [30], and JitterPath [31], to obtain the corresponding values of $\mathrm{BW}_{\mathrm{up}}$ and $\mathrm{BW}_{\mathrm{down}}$.

Finally, the nodes calculate the requested layer rl as follows:

$$rl = \min(rl_{\mathrm{up}},\ rl_{\mathrm{down}}). \tag{8}$$

In the proposed NHAG+ method, the node joining and leaving processes and AG maintenance operate based on this metric.

C. Node Joining Procedure

First, we explain the joining procedure in THAG. In THAG, at the beginning, the node, which wishes to join, will first send a join message to the highest AG that it can enter. If the AG is not completely filled, the node will join that AG. Otherwise, if the AG is completely filled, for each AG member e which already joins the AG, we compute the function $\mathrm{G}_{\mathrm{THAG}}(e)$ which is the ratio of the sum of distances between the node e and the AG sources to the sum of distances between the joining node and AG sources as follows:

$$\mathrm{G}_{\mathrm{THAG}}(e) = \frac{\sum_i \mathrm{Dist}(e, s_i)}{\sum_i \mathrm{Dist}(j, s_i)}. \tag{9}$$

Here, j denotes the joining node, s_i denotes the ith AG source of each description, and $\mathrm{Dist}(a,b)$ is the distance from node a to node b defined in the network coordination technology, such as Global Network Positioning (GNP) [32] and Vivaldi [33]. Network coordinates are obtained by mapping complex Internet topologies into simple geometric space. Note that $(\mathrm{G}_{\mathrm{THAG}}(e) > 1)$ implies that the joining node is closer to the AG sources than node e. Therefore, node e with maximum $\mathrm{G}_{\mathrm{THAG}}(e)$ is replaced by the joining node j. Node e that is replaced will try to find a new child-AG. On the other hand, for all the member nodes e, no replacement is performed if $(\mathrm{G}_{\mathrm{THAG}}(e) \leq 1)$. In this case, j will try to find a new child-AG. Next, the AG entrance that received the join message finds the closest AG member h to the joining node. The joining node contacts h and retrieves the information about all its child-AGs. If h has less child-AGs than it can serve, the joining node creates a new child-AG and joins the AG entrance. Otherwise, the joining node contacts all the child-AG's entrances, and selects and joins the AG that has the smallest average distance between the joining node and the AG members. By repeating these procedures, the joining node eventually joins the closest child-AG, as shown in Fig. 8.

In the joining procedure of NHAG+, we change the node replacing metric and the selection of the child-AG to which the joining node sends the next request message. NHAG+ uses the requested layer as the replacement metric instead of the distance metric as described in the following:

$$\mathrm{G}_{\mathrm{NHAG}}{+}(e) = \frac{\mathrm{RL}(e)}{\mathrm{RL}(j)}. \tag{10}$$

Here, $\mathrm{RL}(a)$ is the requested layer of node a. Note that $(\mathrm{G}_{\mathrm{NHAG}}{+}(e) < 1)$ implies that the joining node's requested layer is larger than that of node e. Therefore, node e with the minimum $\mathrm{G}_{\mathrm{NHAG}}{+}(e)$ is replaced by the joining node j. On the other hand, for all the member nodes e, no replacement is performed if $(\mathrm{G}_{\mathrm{NHAG}}{+}(e) \geq 1)$. If the number of nodes with the minimum $\mathrm{G}_{\mathrm{NHAG}}{+}(e)$ is numerous, we use the distance metric as THAG does. Furthermore, this procedure can promote the node with larger upload bandwidth to a higher tree position. On the other hand, NHAG+ creates trees according to the available bandwidth of nodes as opposed to THAG which

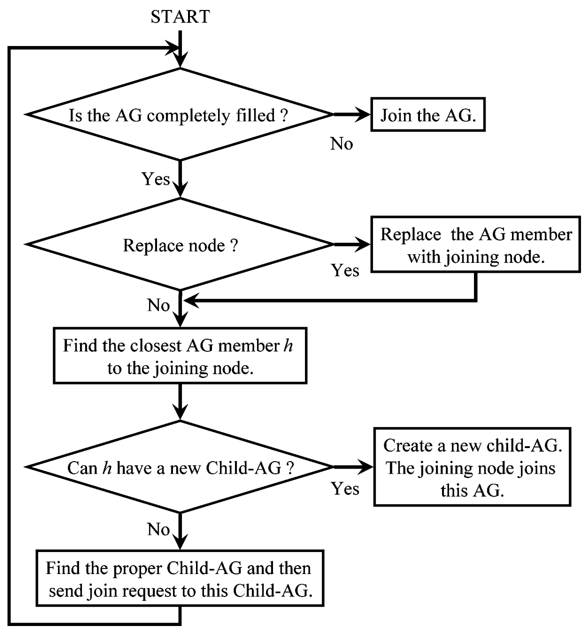

Fig. 8. Node joining procedure.

uses the distance metric to locate nodes. Consequently, the delay along the path created by NHAG+ might be, in some cases, longer than the one created by THAG. However, since the congestion rarely occurs in the upper positions of the path created by NHAG+, the overall delay becomes shorter for NHAG+ as compared to THAG. In the child-AG selection, the joining node contacts all the child-AG's entrances, and selects and joins the AG that has the closest AG layer to the joining node's requested layer. By repeating the above process, a node can join the AG which can adapt to its requested layer and receive the appropriate number of layers.

D. Node Leaving Procedure

Generally, when a node leaves, it sends the leaving messages to the AG entrance and its neighbors. However, in wireless environments, the control messages often fail because of packet errors and handovers. If a handover occurs, the wireless node cannot communicate with the other nodes for a few seconds. Therefore, the wireless nodes may suddenly leave without sending leaving messages. To cope with this problem, in THAG and NHAG, the heartbeat messages are periodically sent among the neighbors [11]. If a node does not receive the heartbeat messages from its neighbor, it assumes that the neighbor node has left.

NHAG+ handles node leaving as follows. If the leaving node is not at the AG entrance, its parent node in the same tree will undertake the position's tasks. If the leaving node is not an AG entrance, its parent node in the same tree will take over the AG entrance tasks. Moreover, if the leaving node has child-AGs, one of its child-AGs promotes a node with the maximum requested layer to replace the vacated position. To realize this procedure, each child-AG i entrance sends the maximum requested layer rl_i of the AG members to its parent-AG entrance periodically. The parent-AG entrance selects $rl_{\max}$ which is the maximum number of rl_i, and sends a notification message to the child-AG entrance to which the node with $rl_{\max}$ belongs. The child-AG, which receives the notification message, then promotes the node, which has the requested layer $rl_{\max}$, to its parent-AG. In the child-AG, similar maintenance can be performed afterward. Thus, we can promote a node from a child-AG, which has a requested layer greater than or equal to rl, even when the node is leaving.

In this way, the height of NHAG+ can be reduced and the nodes with the largest upload bandwidths are promoted to a higher position, as high as possible.

E. Renewal of AG Layer

Since nodes in an AG are frequently replaced due to their joining and leaving events, the AG layer must be dynamically renewed according to the nodes' states. Therefore, the AG layer must be recomputed whenever joining, leaving, or node replacement occurs, or after a certain elapse.

After joining the system, a node i periodically computes its requested layer rl_i based on (8) by using the most recent information of system states, and sends it to the AG entrance. The interval for renewing the AG layer is smaller than the interval for sending node's information. Keeping the renewal frequent enough enables an AG layer to quickly adapt to the dynamics of the system as a whole. The AG entrance assembles each node's requested layer from the AG members and renewed AG layer as follows.

First, the AG entrance computes the minimum requested layer $rl_{\min}$ of the N nodes joining the AG:

$$rl_{\min} = \min_{1<i<N} (rl_i) \tag{11}$$

where N is the number of nodes in the AG. We define the AG layer at time t as $\mathrm{L}(t)$. If the AG entrance sets the AG layer $\mathrm{L}(t)$ to be $rl_{\min}$, the AG overreacts to nodes and network dynamics because $rl_{\min}$ can change abruptly at any time. Therefore, we have implemented a smoothing scheme, which tracks immediate increases and decreases in $\mathrm{L}(t)$. To achieve this, we smooth $\mathrm{L}(t)$ by using the exponentially weighted moving average as follows:

$$\mathrm{L}(t) = \lfloor \alpha \times \mathrm{L}(t-1) + (1-\alpha) \times rl_{\min} \rfloor . \tag{12}$$

When the AG entrance creates a new AG, t and $\mathrm{L}(0)$ are initialized to 0 and rl_1, respectively. Here, rl_1 is the requested layer of the AG entrance.

Thus, we tailor the AG layer to the joining node's network conditions.

V. Performance Evaluation

We have evaluated the performance of the proposed method through extensive simulations by using the network simulator ns-2 [20]. The simulation setup and performance metrics are described in Section V-A. In Section V-B, we present our simulation results and performance comparisons.

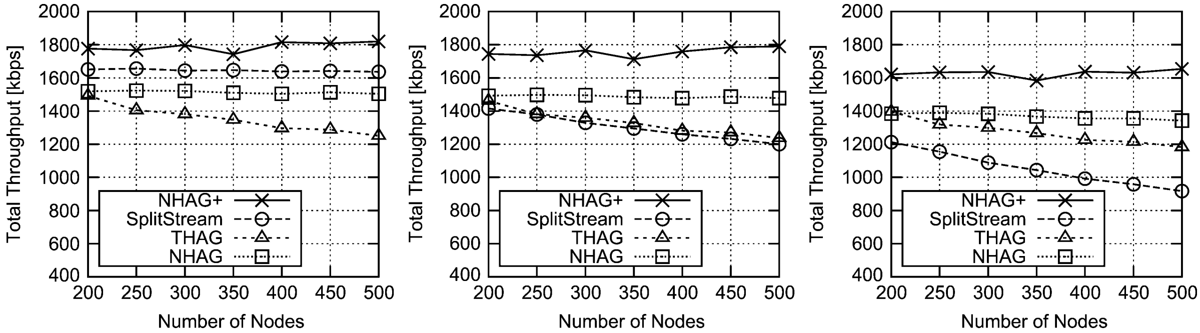

Fig. 9. Comparison of total throughput. (a) Case I. (b) Case II. (c) Case III.

A. Simulation Setup

In our simulations, the transit-stub topology created by the GT-ITM tool [34] was used as the underlying network topology. The network topology consisted of 1010 routers and about 5000 edges. The link delay was randomly set between 1 and 10 ms for each edge. We created end-nodes and randomly connected them to the routers chosen from the stub domain. Each node's upload and download bandwidths are randomly distributed between 2 and 5 Mbps. One of the end-nodes was selected as the media source. The number of end-nodes was varied from 200 to 500. Streaming is delivered in a scenario where each node joins the ALM one by one every 2 s, and after all the nodes have joined, nodes leave one by one every 2 s. When nodes with wireless link leave the trees, they are set to leave without any prior notice to any other nodes in the corresponding trees. This is attributed to the common phenomenon of the wireless environment, i.e., the sudden cutoff of the wireless link. We compare NHAG+ with three contemporary methods, namely SplitStream [10], THAG [11], and NHAG [12] in the following three cases.

In case I, all nodes are wired. In case II, half of the nodes are wired, and the remaining nodes are wireless. Case III comprises wireless nodes only. In the wired nodes, we set the upload and download links' packet error rates to be 0%. On the other hand, in the wireless nodes, we set the upload and download links' packet error rates to be 1% by using the respective error model in ns-2.

We set the rate of the source media data to be 2 Mbps. MDC divided the media into four descriptions, and each description rate was 500 kbps. In LMDC, the number of descriptions N was four, and the number of layers L was four. Therefore, the rate of a layer of each description R_l was 125 kbps. The number of multicast trees was four in all the methods. For each tree, a singular description is allocated. The four layers in each description are adaptively adjusted according to the network bandwidth. THAG and NHAG+ were restricted to the AG size of six. In NHAG, the maximum AG size of six is employed.

In order to evaluate the performance and QoS of each method, we use the following metrics: total throughput, RDP [35], and RDV [11].

1) Total Throughput: Total throughput is defined as the sum of each description's throughput:

$$\text{Total Throughput} = \sum_{i=1}^{M} \text{Throughput}(i). \qquad (13)$$

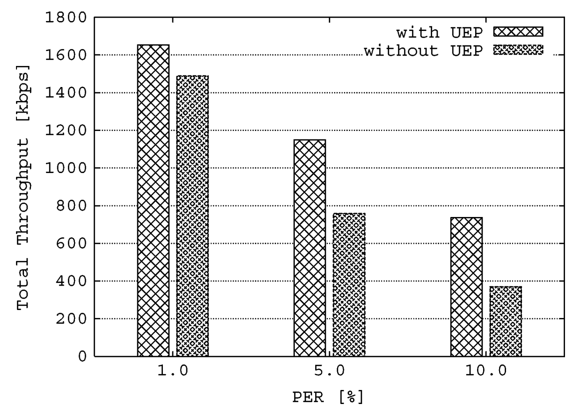

Fig. 10. Throughputs with UEP and without UEP (500 wireless nodes).

Here, $\text{Throughput}(i)$ denotes the received ith description rate and M denotes the number of descriptions.

2) Relative Delay Penalty (RDP): The average ratio of propagation delay on the paths from the source to the receiver node in ALM trees over the end-to-end unicast latency between these nodes is defined as follows:

$$\text{RDP} = \frac{1}{M} \sum_{i=1}^{M} \frac{\text{mDelay}(i)}{\text{uDelay}}. \qquad (14)$$

Here, $\text{mDelay}(i)$ and uDelay denote the multicast delay of the ith description and the unicast delay, respectively. RDP exhibits the relative increase in delay that a packet experiences in ALM as compared to IP multicast.

3) Relative Delay Variation (RDV): The normalized difference of delay on the paths from a source to a node in a different multicast tree is defined as follows:

$$\text{RDV} = \frac{D_{\max} - D_{\min}}{D_{\min}}. \qquad (15)$$

Here, $D_{\max}$ and $D_{\min}$ indicate the maximum and minimum propagation delays, respectively, that are experienced by a node when receiving descriptions from different multicast trees.

To investigate the impact of link errors on the throughput and effectiveness of UEP, we also conducted experiments with link error rates of 1%, 5%, and 10%, respectively. Owing to the constraints of the computer hardware, experiments with up to a

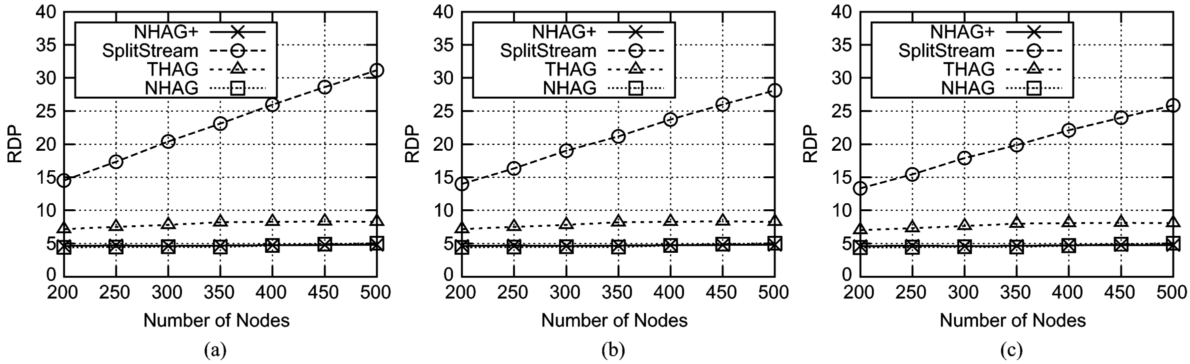

Fig. 11. Comparison of relative delay penalty (RDP). (a) Case I. (b) Case II. (c) Case III.

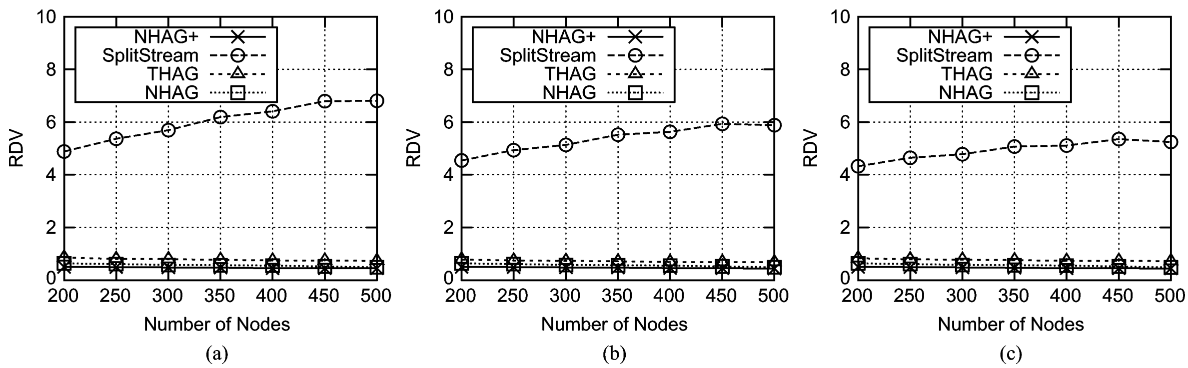

Fig. 12. Comparison of relative delay variation (RDV). (a) Case I. (b) Case II. (c) Case III.

maximum of 500 wireless nodes were conducted. For UEP, we assume that Layer n can only be recovered when at least n descriptions are recovered.

B. Simulation Results

1) Total Throughput: Fig. 9 shows the average total throughput of each method for cases I, II, and III, respectively. We can see that our proposed NHAG+ provides the highest total throughput in all cases. In addition, as the number of wireless nodes increases, the total throughput does not decrease. Furthermore, even when all the nodes have wireless links (case III), NHAG+ achieves substantially higher total throughput (about 1600 kbps). On the other hand, the total throughputs of the other methods decrease as the number of wireless nodes increases. The total throughput achieved by SplitStream is quite low because it cannot construct the node-disjoint trees. Especially, when the number of nodes is 500, SplitStream's total throughput is lower than 1000 kbps. In THAG, the robustness of node departure is high because all nodes join all the trees, but THAG cannot meet the various bandwidth constraints. Therefore, THAG's total throughput is lower than those of NHAG and NHAG+. In NHAG, the total throughput is also lower than that of NHAG+ because the robustness of node departure is lower than that of NHAG+.

Fig. 10 shows the results of the total throughput under the influences of packet error of 1%, 5%, and 10%, respectively. There are 500 nodes. From Fig. 10, we can see that the larger the packet error rate, the lower the total throughput. Note that the total throughput can be improved greatly with UEP.

2) Relative Delay Penalty (RDP): We study the average RDP in SplitStream, THAG, NHAG, and NHAG+. The results are shown in Fig. 11. In all cases, both NHAG+ and NHAG provide the lowest RDP which is about five because these methods meet the nodes' bandwidth constraints. On the other hand, RDP of THAG is about eight and higher than those of NHAG+ and NHAG. This is because, in THAG, congestion occurs in the upload link of a node, and hence the queuing delay increases. RDP achieved by SplitStream is indeed high and increases as the number of nodes increases in all three cases. This result demonstrates that NHAG+ and NHAG achieve low delay while delay of SplitStream is significantly high. Furthermore, no matter how large the number of nodes is, the RDP in NHAG+ and NHAG remains reasonably small and constant in all the considered cases. Since it is more difficult to optimize multiple multicast trees in a distributed network environment at the same time than in a single multicast tree [10], the small RDP values incurred in NHAG+ and NHAG are remarkable.

3) Relative Delay Variation (RDV): Fig. 12 indicates the average RDV achieved by each method. We can see that the proposed NHAG+ scheme, THAG, and NHAG provide low RDV values (smaller than 1) in all cases. Hence, in these methods, the difference between each description's delay is small. SplitStream achieves very high RDV which is larger than four, and the RDV increases as the number of nodes increases. Therefore, the difference between each description's delay is large. When the number of nodes increases, the variation of delay in SplitStream is more drastic. Hence, the nodes in THAG can experience relatively consistent media delivery from different trees. From the results pertaining to RDP and RDV, it can be seen that the proposed NHAG+ and NHAG provide high levels of QoS, while the SplitStream's QoS remains low. The simulation results indicate that SplitStream achieves low total throughput. Furthermore, the delay and the difference in delay for each description are quite large, and so QoS of the stream cannot be provisioned. THAG achieves low total throughput and high RDP because it cannot meet the various bandwidth constraints. NHAG adapts to available bandwidth constraints, but the robustness of node departure is poor. So, NHAG achieves equally low RDP and RDV, but the overall throughput is low as compared to that of NHAG+. However, NHAG+ provides high total throughput, and low RDP and RDV in all three cases. In summary, NHAG+ can consistently provision content delivery in combined wired and wireless networks.

VI. Conclusion

Recently, the Internet has evolved from a wired infrastructure to a hybrid of wired and wireless domains by spreading WiMAX, Wi-Fi, and cellular networks. Therefore, there is a growing need for an efficient content delivery system in combined wired and wireless networks.

In this paper, we have tailored the ALM protocol for this environment. We have studied and examined in details THAG and NHAG, which splits a stream into several descriptions with MDC and delivers each description along node-disjoint multicast trees constructed from AGs. However, in THAG, because the AG size is constant, it is difficult to deliver descriptions appropriately across a heterogeneous network. NHAG can meet the various bandwidth constraints by changing the size of an AG dynamically, but the robustness of node departure is degraded. These methods cannot deliver contents reliably in heterogeneous networks with wired and wireless portions. Therefore, we have proposed NHAG+ which adapts to bandwidth constraints without decreasing the number of joining trees by using LMDC. In the proposed NHAG+ method, each node can receive the appropriate number of layers according to its available bandwidth without decreasing the number of descriptions.

Our simulation results by using network simulator ns-2 have demonstrated the effectiveness of our proposal in terms of total throughput, RDP, and RDV. These results indicate that NHAG+ is a reliable and efficient ALM scheme for streaming media services across heterogeneous networks.

References

[1] *IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems*, ANSI/IEEE Std. 802.16-2004, 2004.

[2] *IEEE Standard for Information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, ANSI/IEEE Std. 802.11-2007, 2007.

[3] 3 GPP. [Online]. Available: http://www.3gpp.org/.

[4] 3 GPP2. [Online]. Available: http://www.3gpp2.org/.

[5] B. Alfonsi, "I want my IPTV: Internet protocol television predicted a winne," *IEEE Distrib. Syst. Online*, vol. 6, no. 2, Feb. 2005.

[6] A. M. Hamad and A. E. Kamal, "A survey of multicasting protocols for broadcast-and-select single-hop networks," *IEEE Netw.*, vol. 16, no. 4, pp. 36–48, Jul./Aug. 2002.

[7] M. Hosseini, D. T. Ahmed, S. Shirmohammadi, and N. D. Georganas, "A survey of application-layer multicast protocols," *IEEE Commun. Surveys Tuts.*, vol. 9, no. 3, pp. 58–74, 2007.

[8] V. N. Padmanabhan, H. J. Wang, P. A. Chou, and K. Sripanidkulchai, "Distributing streaming media content using cooperative networking," in *Proc. ACM Int. Workshop Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, May 2002, pp. 177–186.

[9] V. N. Padmanabhan, H. J. Wang, and P. A. Chou, "Resilient peer-to-peer streaming," in *Proc. IEEE Int. Conf. Network Protocol (ICNP)*, Nov. 2003, pp. 16–27.

[10] M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, "Splitstream: High-bandwidth content distribution in cooperative environments," in *Proc. Int. Workshop Peer-to-Peer Systems (IPTPS)*, Oct. 2003, pp. 298–313.

[11] R. Tian, Q. Zhang, Z. Xiang, Y. Xiong, X. Li, and W. Zhu, "Robust and efficient path diversity in application-layer multicast for video streaming," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 8, pp. 961–972, Aug. 2005.

[12] M. Kobayashi, H. Nakayama, N. Ansari, and N. Kato, "Robust and efficient stream delivery for application layer multicasting in heterogeneous networks," *IEEE Trans. Multimedia*, vol. 11, no. 1, pp. 166–176, Jan. 2009.

[13] M. Alasti, K. Sayrafian-Pour, A. Phremides, and N. Farvardin, "Multiple description coding in networks with congestion problem," *IEEE Trans. Inf. Theory*, vol. 47, no. 3, pp. 891–902, Mar. 2001.

[14] V. K. Goyal, "Multiple description coding: Compression meets the network," *IEEE Signal Process. Mag.*, vol. 18, no. 5, pp. 74–93, Sep. 2001.

[15] V. Stankovic, R. Hamzaoui, and Z. Xiong, "Robust layered multiple description coding of scalable media data for multicast," *IEEE Signal Process. Lett.*, vol. 12, no. 2, pp. 154–157, Feb. 2005.

[16] P. A. Chou, H. J. Wang, and V. N. Padmanabhan, "Layered multiple description coding," in *Proc. Int. Packet Video Workshop*, Apr. 2003.

[17] R. Mathew and J. F. Arnold, "Layered coding using bitstream decomposition with drift correction," *IEEE Trans. Circuits Syst. Video Technol*, vol. 7, no. 12, pp. 882–891, Dec. 1997.

[18] K. Day and A. Tripathi, Characterization of Node Disjoint Paths in Arrangement Graphs, Comput. Sci. Dept., Univ. Minnesota, 1991, TR91-43.

[19] Y.-S. Chen, T.-Y. Juang, and E.-H. Tseng, "Congestion-free embedding of $2(n-k)$ spanning trees in an arrangement graph," *J. Syst. Archit.*, vol. 47, no. 1, pp. 73–86, Jan. 2001.

[20] The network simulator—ns-2. [Online]. Available: http://www.isi.edu/nsnam/ns/.

[21] P. Francis, Yoid: Extending the Internet multicast architecture, Apr. 2000. [Online]. Available: http://www.icir.org/yoid/.

[22] H. Deshpande, M. Bawa, and H. Garcia-Molina, Streaming Live Media Over a Peer-to-Peer Network, Stanford Univ., 2001, Tech. Rep. CS-2001-31.

[23] D. Pendarakis, S. Shi, D. Verma, and M. Waldvogel, "ALMI: An application level multicast infrastructure," in *Proc. USENIX Symp. Int. Technology and Systems (USITS)*, Mar. 2001, pp. 49–61.

[24] V. Roca and A. El-Sayed, "A host-based multicast (HBM) solution for group communications," in *Proc. IEEE Int. Conf. Networking (ICN)*, Jul. 2001, pp. 610–619.

[25] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, "Scalable application layer multicast," *Proc. ACM Special Int. Group on Data Communication (SIGCOMM)*, pp. 205–217, Aug. 2002.

[26] D. A. Tran, K. A. Hua, and T. T. Do, "A peer-to-peer architecture for media streaming," *IEEE J. Select. Areas Commun.*, vol. 22, no. 1, pp. 121–133, Jan. 2004.

[27] M. Castro, P. Druschel, A.-M. Kermarrec, and A. Rowstron, "Scribe: A large-scale and decentralized application-level multicast infrastructure," *IEEE J. Select. Areas Commun.*, vol. 20, no. 8, pp. 1489–1499, Oct. 2002.

[28] L.-J. Chen, C.-H. Yu, C.-L. Tseng, H. h. Chu, and C.-F. Chou, "A content-centric framework for effective data dissemination in opportunistic networks," *IEEE J. Select. Areas Commun.*, vol. 26, no. 5, pp. 761–772, Jun. 2008.

[29] N. Hu and P. Steenkiste, "Evaluation and characterization of available bandwidth probing techniques," *IEEE J. Select. Areas Commun.*, vol. 21, no. 6, pp. 879–894, Aug. 2003.

[30] M. Jain and C. Dovrolis, "End-to-end available bandwidth: Measurement methodology, dynamics, and relation with TCP throughput," *IEEE/ACM Trans. Netw.*, vol. 11, no. 4, pp. 537–549, Aug. 2003.

[31] Y.-C. Huang, C.-S. Lu, and H.-K. Wu, "Jitterpath: Probing noise resilient one-way delay jitter-based available bandwidth estimation," *IEEE Trans. Multimedia*, vol. 9, no. 4, pp. 798–812, Jun. 2007.

[32] T. S. E. Ng and H. Zhang, "Predicting Internet network distance with coordinates-based approaches," in *Proc. IEEE Conf. Computer Communications (INFOCOM)*, Jun. 2002, pp. 170–179.

[33] F. Dabek, R. Cox, F. Kaashoek, and R. Morris, "Vivaldi: A decentralized network coordinate system," *Proc. ACM Special Int. Group on Data Communications (SIGCOMM)*, pp. 15–26, Aug. 2004.

[34] E. W. Zegura, K. L. Calvert, and S. Bhattacharjee, "How to model an internetwork," in *Proc. IEEE Conf. Computer Communications (INFOCOM)*, Mar. 1996, pp. 594–602.

[35] Y.-H. Chu, S. G. Rao, S. Seshan, and H. Zhang, "A case for end system multicast," *IEEE J. Select. Areas Commun.*, vol. 20, no. 8, pp. 1456–1471, Oct. 2002.

Masahiro Kobayashi (S'08–M'09) received the B.E. and M.S. degrees in information engineering from Tohoku University, Sendai, Japan, in 2007 and 2009, respectively.

He is currently working in NTT Service Integration Laboratories, Tokyo, Japan. His research interests are in the areas of application layer multicast and overlay networks.

Hidehisa Nakayama (M'06) received the B.E., M.S., and Ph.D. degrees in information sciences from Tohoku University, Sendai, Japan, in 2000, 2002, and 2005, respectively.

He is a Senior Assistant Professor at the Tohoku Institute of Technology. He has been engaged in research on intelligent sensor technology, wireless mobile ad hoc network, computer networking, character string analysis, and pattern recognition.

Dr. Nakayama is a member of IEEE Communications Society, the Institute of Electronics, Information and Communication Engineers (IEICE), and the Information Processing Society of Japan (IPSJ). He received the Best Paper Award for Young Researcher of IPSJ Tohoku Chapter in 2000, the Best Paper of Pattern Recognition Award in SCI 2003, and the eighth Network System Award of IEICE Technical Committee on Network Systems in 2009.

Nirwan Ansari (S'78–M'83–SM'94–F'09) received the B.S.E.E. (summa cum laude) from the New Jersey Institute of Technology (NJIT), Newark, in 1982, the M.S.E.E. degree from University of Michigan, Ann Arbor, in 1983, and the Ph.D. degree from Purdue University, West Lafayette, IN, in 1988.

He joined NJIT's Department of Electrical and Computer Engineering as an Assistant Professor in 1988 and has been a Full Professor since 1997. He has also assumed various administrative positions at NJIT. He authored *Computational Intelligence for Optimization* (New York: Springer, 1997, translated into Chinese in 2000) with E.S.H. Hou and edited *Neural Networks in Telecommunications* (New York: Springer, 1994) with B. Yuhas. His current research focuses on various aspects of broadband networks and multimedia communications. He has also contributed over 300 technical papers, over one third of which are in refereed journals/magazines.

Dr. Ansari is a Senior Technical Editor of the *IEEE Communications Magazine* and also serves on the Advisory Board and Editorial Board of five other journals. He has been serving the IEEE in various capacities such as Chair of IEEE North Jersey COMSOC Chapter, Chair of IEEE North Jersey Section, Member of IEEE Region 1 Board of Governors, Cluster Chair of IEEE COMSOC Networking TC Cluster, Chair of IEEE COMSOC Technical Committee on Ad Hoc and Sensor Networks, and Chair/TPC Chair of several conferences/symposia. Some of his recent awards and recognitions include an IEEE Fellow (Communications Society), IEEE Leadership Award (2007, from Central Jersey/Princeton Section), the NJIT Excellence in Teaching in Outstanding Professional Development (2008), IEEE MGA Leadership Award (2008), the NCE Excellence in Teaching Award (2009), and designation as an IEEE Communications Society Distinguished Lecturer.

Nei Kato (A'03–M'04–SM'05) received the M.S. and Ph.D. degrees from Tohoku University, Sendai, Japan, in 1988 and 1991, respectively.

He has been working for Tohoku University since then and is currently a Full Professor at the Graduate School of Information Sciences. He has been engaged in researches on computer networking, wireless mobile communications, image processing, and neural networks. He has published more than 130 papers in journals and peer-reviewed conference proceedings.

Dr. Kato has served as a symposium co-chair at GLOBECOM'07 and ChinaCom'08, and a TPC member in a large number of IEEE international conferences, including ICC, GLOBECOM, WCNC, and HPSR. He was a co-guest-editor for JCN special issue on Broadband convergence Network (BcN) in 2005. He has been a technical editor of *IEEE Wireless Communications* (since 2006) and an editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS (since 2008). He is a co-recipient of the 2005 Distinguished Contributions to Satellite Communications Award from the IEEE Communications Society, Satellite and Space Communications Technical Committee, the co-recipient of FUNAI information Science Award, 2007, and the co-recipient of 2008 TELCOM System Technology Award from Foundation for Electrical Communications diffusion. He is serving as an expert member of Telecommunications Council, Ministry of Internal Affairs and Communications, Japan. He is a member of the Institute of Electronics, Information and Communication Engineers (IEICE).