

Delay Analysis of Feedback-Synchronization Signaling for Multicast Flow Control

Xi Zhang, *Senior Member, IEEE*, and Kang G. Shin, *Fellow, IEEE*

Abstract—Feedback signaling plays a key role in flow control because the traffic source relies on the signaling information to make *correct* and *timely* flow-control decisions. Design of an efficient signaling algorithm is a challenging task since the signaling messages can tolerate neither error nor latency. Multicast flow-control signaling imposes two additional challenges: *scalability* and *feedback synchronization*. Previous research on multicast feedback-synchronization signaling has mainly focused on the algorithm design and implementation. However, the delay properties of these algorithms are, despite their vital importance, neither well understood nor thoroughly studied. In this paper, we develop both *deterministic* and *statistical* binary-tree models to study the delay performance of the multicast signaling algorithms. The deterministic model is used to derive the expressions of each path's feedback roundtrip time in a multicast tree, while the statistical model is employed to derive the general probability distributions of each path becoming the multicast-tree bottleneck. Using these models, we analyze and contrast the signaling delay scalability of two representative multicast signaling protocols—the *Soft-Synchronization Protocol* (SSP) and the *Hop-By-Hop* (HBH) scheme—by deriving the first and second moments of multicast signaling delays. Also derived is the optimal flow-control update interval for SSP to minimize the multicast signaling delay.

Index Terms—Feedback roundtrip time (RTT), feedback-synchronization signaling, multicast flow control, soft-synchronization protocol (SSP).

I. INTRODUCTION

A FLOW-CONTROL algorithm consists of two basic components: rate control and flow-control feedback signaling. These two components are conceptually separate from the flow-control theory's standpoint, but are often blended together in most flow-control algorithms. Rate control adapts the source rate to the dynamic variation of available network bandwidth. Flow-control signaling delivers the information related to congestion and rate control between the source and network/receivers. Consequently, this signaling is critically important to flow control because the source relies on the signaling information in making *correct* and *timely* flow-control decisions. Designing an efficient flow-control signaling protocol is difficult because the signaling messages—unlike data

or audio/video traffic—tolerate neither error nor a large latency. A signaling message could be useless or even harmful if it is not timely and accurate. For instance, in ATM ABR service, flow-control signaling employs the resource management (RM) cells to convey the rate-control and congestion information among the source rate controller, network switches, and the receivers.

Signaling for multicast flow control introduces two additional problems: *scalability* and *feedback synchronization*. These two problems are closely related in the signaling protocol for multicast flow control. First, simultaneous feedback arrivals from all downstream branches can cause a *feedback implosion* [1] at the source or at a branch point, especially when the multicast tree is large. Hence, it is important for each branch point to consolidate the congestion information carried by the feedbacks from its downstream branches and send only the consolidated feedback to its upstream node. Second, we need a feedback-synchronization signaling algorithm for consolidating feedbacks at each branch point, because different downstream branches' feedbacks may arrive at significantly different times.

The first-generation feedback consolidation algorithms [2]–[5] for multicast ABR flow control employ a simple hop-by-hop (HBH) mechanism to deal with the feedback-implosion problem. The HBH scheme works as follows. On receipt of one forward RM cell at each branch node, only *one* consolidated feedback RM cell is propagated upward by a *single* hop. While HBH ensures that at each multicast branch point, the ratio of feedback RM cells to the forward RM cells is no larger than one, its multicast signaling delay is proportional to the multicast-tree height. Thus, HBH scheme does not scale well in terms of multicast signaling delay. Moreover, since the feedback RM cells from downstream nodes are *randomly* consolidated without strict synchronization (or *freely synchronized*) at branch points, the source may be misled by this incomplete feedback information, causing *consolidation noise* [6], [7]. So, HBH performs poorly in terms of signaling accuracy.

To reduce the RM-cell feedback roundtrip time (RTT) and improve the multicast signaling accuracy, the authors of [6] and [7] proposed feedback synchronization by accumulating feedback from *all* branches. The main drawback of this scheme is its slow transient response, as the feedback from the congested branch may have to needlessly wait for the feedback from longer paths, which may not be congested at all. The authors of [8] proposed an algorithm to speed up the transient response by sending fast congestion feedback without awaiting all branches' feedbacks in the transient phase.

One critical deficiency of the schemes described above is that they do not detect and remove nonresponsive branches

Manuscript received August 14, 2000; revised December 11, 2001; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor K.-Y. Siu. This work was supported in part by the U.S. Office of Naval Research under Grant N00014-99-1-0465. A subset of the materials in this paper was presented at the IEEE GLOBECOM'99, Rio de Janeiro, Brazil, 1999.

X. Zhang is with the Networking and Information Systems Laboratory, Department of Electrical Engineering, Texas A&M University, College Station, TX 77843 USA (e-mail: xizhang@ee.tamu.edu).

K. G. Shin is with the Real-Time Computing Laboratory, Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: kgshin@eecs.umich.edu).

Digital Object Identifier 10.1109/TNET.2003.813035

during feedback synchronization. One or more nonresponsive branches may detrimentally impact signaling accuracy and timeliness by providing either stale congestion information, or by stalling the entire multicast connection. In [9] and [10], we proposed a novel feedback-synchronization signaling algorithm, called the *Soft-Synchronization Protocol* (SSP), which derives a single consolidated RM cell at each branch point from feedback RM cells of different downstream branches that are not necessarily responses to the same forward RM cell in each synchronization cycle. The SSP not only scales well with the multicast-tree topology, but also can readily detect and remove nonresponsive branches. SSP scales better than HBH because the feedback delay of SSP is not subject to the multicast-tree height, but only depends on the length of the most congested path within a multicast tree. In addition, unlike HBH, at each branch point SSP synchronizes the feedbacks from all connected downstream branches and thus, always extracts the feedback congestion information from the most congested path within the multicast tree, which ensures the feedback signaling accuracy as far as the source's flow control is concerned.

All of the above-referenced work only focused on the design and implementation of multicast signaling algorithms. However, the delay properties of these algorithms are, despite their vital importance, neither well understood nor thoroughly studied. In this paper, we develop the deterministic and statistical models to investigate the delay performance of multicast signaling algorithms. The benefits of developing these modeling techniques are twofold. First, the derived models enable the direct quantitative comparison, which deterministically and statistically shows that SSP outperforms HBH in terms of multicast signaling delay. Our analytical results also show that SSP can not only support efficient multicast signaling, but also make the effective multicast signaling delay virtually independent of the multicast-tree's height and path lengths. Second, the proposed modeling technique establishes a general signaling-delay evaluation framework for different feedback-synchronization protocols. The modeling technique based on ABR multicast can be applied to the signaling-delay analysis for any other multicast flow-control algorithms.

The paper is organized as follows. Section II details SSP. In Section III, we introduce the binary-tree model and apply it to derive the feedback signaling delay of each path for SSP and HBH, respectively. Section IV investigates the statistical properties of multicast signaling protocols in terms of average multicast-tree RM-cell RTTs and delay variations for both SSP and HBH. Section V describes the simulation experiments, which verify the analytical results. In Section VI, we derive the optimal RM-cell interval for SSP to minimize the RM-cell RTTs for a given multicast tree. The paper concludes with Section VII.

II. DESCRIPTION OF SSP

The RM-cell flows for multicast-feedback signaling are illustrated by Fig. 1(a) and (b) with an example of one multicast source connected to four receivers R_1, \dots, R_4 through an unbalanced binary multicast tree. The source periodically sends the forward RM cells, which are replicated and multicast to multiple downstream nodes at each branch point/switch while they traverse along every path through the entire multicast tree, forming the forward RM cell flows as shown in Fig. 1(a). Upon

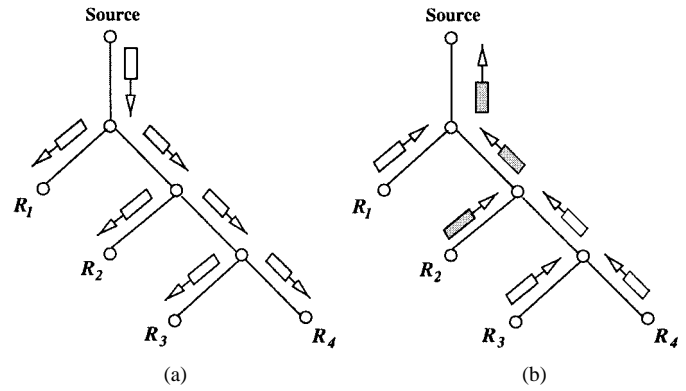


Fig. 1. Forward and backward RM-cell flows in an unbalanced multicast tree. (a) Forward RM-cell flows. (b) Backward RM-cell flows.

the arrival of the forward RM cell at their receivers R_1, \dots, R_4 , each multicast receiver returns a feedback RM cell with its congestion indication (CI) bit marked, or unmarked, depending on whether the receiver is, or not, congested. The backward RM cells return to the source, as shown in Fig. 1(b), where the shaded, or blank, rectangular boxes denote the marked, or unmarked, feedback RM cell, respectively. At each branch point, the consolidated feedback RM cell is marked, or kept intact (determined by the local branch switch's congestion status), if at least one, or none, of downstream branches connected to this branch point is congested/marked.

Based on the above RM-cell flow descriptions for multicast signaling given in Fig. 1(a) and (b), we now present an overview of SSP, especially the switch feedback-synchronization algorithm [9], [10]. At the heart of SSP is a pair of connection-update vectors: 1) *conn_patt_vec*, the connection pattern vector where $conn_patt_vec(i) = 0$ (1) indicates the i th output port of the switch is (not) a downstream branch of the multicast connection. Thus, $conn_patt_vec(i) = 0$ (1) implies that a data copy should (not) be sent to the i th downstream branch and a feedback RM cell is (not) expected from the i th downstream branch,¹ and 2) *resp_branch_vec*, the responsive branch vector is initialized to $\underline{0}$ (an all 0's vector) and reset to $\underline{0}$ whenever a consolidated RM cell is sent upward from the switch. $resp_branch_vec(i)$ is set to 1 if a feedback RM cell is received from the i th downstream branch. The connection pattern specified in *conn_patt_vec* is updated by *resp_branch_vec* each time when the nonresponsive branch is detected or a new connection request is received from a downstream branch.

A simplified pseudocode of the switch feedback RM-cell processing algorithm is given in Fig. 2. On receipt of a feedback RM cell from a connected downstream branch, the switch first marks its corresponding bit in *resp_branch_vec* and then performs RM-cell consolidation operations. If the modulo-2 addition (the soft-synchronization operation), $conn_patt_vec \oplus resp_branch_vec = \underline{1}$, an all 1's vector, indicating all feedback RM cells are synchronized, then a fully consolidated feedback RM cell is generated and sent upward. But if the modulo-2 addition is not equal to $\underline{1}$, the switch needs to await other feedback RM cells for synchronization. Notice that since the synchronization algorithm allows feedback RM cells corresponding to different forward RM cells to be consolidated with

¹Note that the negative logic is used for convenience of implementation.

```

00. On receipt of a feedback RM cell from the  $i$ -th branch:
01. if ( $conn\_patt\_vec(i) \neq 1$ ) { ! Only process connected branches;
02.    $resp\_branch\_vec(i) := 1$ ; ! Mark connected and responsive branch;
03.    $MCI := MCI \vee CI$ ; ! Bandwidth-congestion indicator processing;
04.    $MER := \min\{MER, ER\}$ ; ! ER information processing;
05.   if ( $conn\_patt\_vec \oplus resp\_branch\_vec = \underline{1}$ ) { ! Soft Synchronization;
06.     send RM cell ( $dir := back, ER := MER, CI := MCI$ );
07.      $no\_resp\_timer := N_{nrt}$ ; ! Reset non-responsive timer;
08.      $resp\_branch\_vec := \underline{0}$ ; ! Reset responsive branch vector;
09.      $MCI := 0; MER := ER$ ; ! Reset RM-cell control variables.

```

Fig. 2. Pseudocode for switch feedback-synchronization algorithm.

each other, the feedback RM cells are “softly synchronized” or “loosely synchronized” at branch nodes.

We now summarize what makes SSP different from, better scalable than, and more accurate than, HBH, as follows.

- S1. SSP can complete a feedback loop with a single forward RM cell while HBH needs to send multiple forward RM cells for each returned backward RM cell, thus making SSP scale better than HBH since the resulting HBH’s feedback delay is proportional to the tree height while SSP is not.
- S2. SSP employs the soft synchronization, which derives a consolidated RM cell at each branch point from feedback RM cells of different downstream branches that are not necessarily responses to the same forward RM cells in each synchronization cycle, further shortening RTT delay.
- S3. SSP forward a consolidated-feedback RM cell at each branch point if and only if (*iff*) the branch point has received at least one feedback RM cell from each connected downstream branch, ensuring the accuracy of feedback signaling. But HBH forwards a consolidated feedback RM cell at each branch point *iff* the branch point received a forward RM cell from the upstream without synchronizing feedbacks, which can cause consolidation noise.
- S4. SSP can detect and remove the nonresponsive branches or subtrees while HBH cannot.

III. DETERMINISTIC DELAY ANALYSIS OF MULTICAST SIGNALING

A. The Deterministic Binary-Tree Model

To simplify the analysis of RM-cell RTTs, we *quantize* the network feedback delay by assuming each switch hop to have a uniform delay that includes the processing and propagation delays. This assumption can be relaxed easily because the difference in switch-processing delays and the link-propagation delays of different switch hops can be translated into different numbers of switch hops, each with the same delay. We use the hop delay τ_h , which is the link-propagation delay in each hop, as the time unit in our delay analysis in Sections III and IV. To study the worst case and enable performance comparison, we only consider two types of multicast trees: *balanced* and *unbalanced* binary trees. Since we are only concerned with a path’s RM-cell RTT, which is determined by its length, it suffices to consider binary trees. Notice that in an unbalanced binary tree, the number of paths n from the root to all leaves equals the

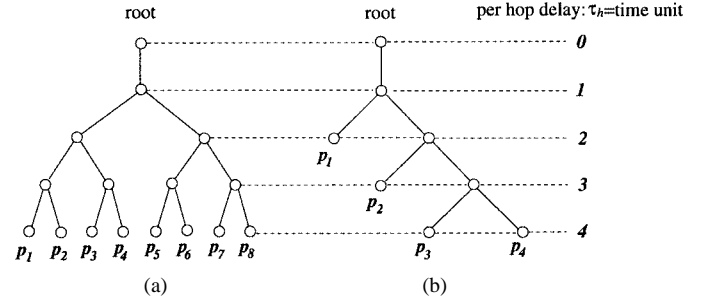


Fig. 3. Balanced and unbalanced binary multicast-tree models. (a) Balanced tree: $m = 4$. (b) Unbalanced tree: $m = 4$.

height of the tree m , while in a balanced binary tree $n = 2^{m-1}$. Fig. 3 illustrates these two types of trees with height $m = 4$.

As discussed in [8] and [10] for ABR services, only the feedback from the most congested path in a multicast tree governs the source flow-control operations. However, the RM-cell RTT of different paths in a multicast tree may vary significantly due mainly to the difference in their lengths. Thus, we need to analyze each path’s RM-cell RTT in a multicast tree. The individual path’s RTT is also affected by the feedback-synchronization algorithms used. In addition, the RM-cell RTT for a given path may vary at the beginning of the flow-control operation (in an initial state) when feedback RM cells are not yet “regularly” synchronized. The RM-cell RTT becomes stable after feedback RM cells are regularly synchronized (in a steady state).

Using the balanced and unbalanced binary-tree models shown in Fig. 3, we analyze the two representative multicast signaling protocols and derive their feedback-delay properties, which are summarized by Theorem 1 (Corollary 1) and Theorem 2 (Corollary 2) for HBH and SSP, respectively, in the following sections. The common parameters used in Theorem 1 (Corollary 1) and Theorem 2 (Corollary 2) are defined as follows:

- m multicast tree height;
- $\tau_{\max} \triangleq 2m$: maximum RTT;
- Δ RM-cell interval in the unit of hop delay τ_h .

B. Feedback-Delay Properties for the HBH Scheme

The following theorem derives the equations for all paths’ RM-cell RTTs in an unbalanced binary multicast tree with HBH.

Theorem 1: If an unbalanced multicast tree of height $m \geq 2$ is flow-controlled by HBH with $\Delta \geq 1$ (τ_h), then the RM-cell RTT, denoted by $\tau_u(j, \Delta)$, of the j th (counting from left to right) path P_j remains the same in both steady and initial states, and is determined by

$$\tau_u(j, \Delta) = 2 + j\Theta(\Delta) \quad (1)$$

where $1 \leq j \leq m - 1$ and $\Theta(\Delta)$ is defined by

$$\Theta(\Delta) \triangleq \max\{2, \Delta\} = \begin{cases} \Delta, & \text{if } 2 \leq \Delta \leq \tau_{\max} \\ 2, & \text{if } \Delta = 1 \end{cases} \quad (2)$$

where $1 \leq \Delta \leq \tau_{\max}$.²

Proof: The proof is available on-line in [11]. ■

Based on Theorem 1, Corollary 1 derives the equations for all paths’ RM-cell RTTs in a balanced binary tree with HBH.

²Theorem 1 still holds even when $\Delta \geq \tau_{\max} = 2m$. But the RM-cell update interval Δ is usually a fraction of the maximum RM-cell RTT. So, we do not consider the case of $\Delta \geq \tau_{\max} = 2m$.

Corollary 1: If a balanced multicast tree of height $m \geq 2$ is flow-controlled by HBH with $\Delta \geq 1$, then RM-cell RTTs of all paths, denoted by $\tau_b(j, \Delta)$, are the same in both steady and initial states, and are determined by

$$\begin{aligned} \tau_b(j, \Delta) &= \max_{j \in \{1, 2, \dots, m-1\}} \{\tau_u(j, \Delta)\} \\ &= \tau_{\max} + (m-1)[\Theta(\Delta) - 2] \end{aligned} \quad (3)$$

where $1 \leq j \leq 2^{m-1}$, and $\tau_u(j, \Delta)$ and $\Theta(\Delta)$ are defined by (1) and (2), respectively, for an unbalanced multicast tree of the same height.

Proof: The proof follows by letting $j = m - 1$ in (1). ■

C. Feedback-Delay Properties for the SSP Scheme

The following lemma characterizes the synchronization relationships between different paths under SSP, which lays the foundation for Lemma 2.

Lemma 1: Consider an unbalanced multicast tree of height $m > 2$. Let P_i be a relatively shorter path than another path $P_{\tilde{i}}$ such that $1 \leq i < \tilde{i} \leq m - 1$. If the multicast tree is flow-controlled by SSP with $\Delta \geq 1$, then P_i 's feedback RM cell need not wait for $P_{\tilde{i}}$'s feedback RM cell for synchronization at any branch node.

Proof: The proof is available on-line in [11]. ■

The lemma given below reveals four *iff* conditions for a path's RM-cell RTT to attain its limiting minimum, which consists of propagation delays only (i.e., no synchronization delay).

Lemma 2: If P_j is the j th path in an unbalanced binary multicast tree, which is defined in Lemma 1 for $1 \leq j \leq m - 1$ and is flow-controlled by SSP, then the following four claims are equivalent for the steady-state RM-cell RTT:

Claim 1: P_j 's feedback RM cell need not wait for a longer path $P_{\tilde{j}}$'s ($\tilde{j} > j$) feedback RM cell to achieve feedback synchronization at the first branch node from P_j 's leaf.

Claim 2: P_j 's feedback RM cell need not wait for feedback RM cells for synchronization at *any* branch node on P_j .

Claim 3: $\exists k \in \{0, 1, 2, \dots\}$ such that $2(m-j-1) - k\Delta = 0$, where $1 \leq j \leq m - 1$ and $1 \leq \Delta \leq \tau_{\max} = 2m$.

Claim 4: P_j 's steady-state RM cell RTT $\tau_u(j, \Delta)$ attains its minimum and is given by

$$\tau_u(j, \Delta) = \min_{\Delta} \{\tau_u(j, \Delta)\} = 2(j+1) \quad (4)$$

where $1 \leq j \leq m - 1$ and $1 \leq \Delta \leq \tau_{\max}$.

Proof: The proof is available on-line in [11]. ■

Using Lemmas 1 and 2, we obtain the following theorem, which derives the formulas for all paths' RM-cell RTTs in both initial and steady states in an unbalanced binary tree under SSP.

Theorem 2: Let P_j be the j th path of an unbalanced tree as defined in Lemma 1 ($1 \leq j \leq m - 1$). If the multicast tree is flow-controlled by SSP with $1 \leq \Delta \leq \tau_{\max}$,³ then the following claims hold for $j = 1, \dots, m - 1$.

Claim 1: The number of P_j 's feedback RM cells going through initial state is determined by

$$k_j^* \triangleq \max_{k \in \{0, 1, 2, \dots\}} \{k | 2(m-j-1) - k\Delta \geq 0\}. \quad (5)$$

³Theorem 2 still holds for $\Delta > \tau_{\max} = 2m$, but Δ is typically a fraction of the maximum RM-cell RTT $\tau_{\max} = 2m$.

Claim 2: P_j 's RM-cell RTT in steady state is determined by

$$\tau_u(j, \Delta) = \tau_{\max} - k_j^* \Delta. \quad (6)$$

Claim 3: The i th RM-cell's RTT during P_j 's initial state is determined by the following equations:

$$\tau_u(j, \Delta, i) = \begin{cases} \tau_{\max} - (i-1)\Delta, & \text{if } k_j^* \geq 1 \wedge 1 \leq i \leq k_j^* \\ \tau_u(j, \Delta), & \text{if } k_j^* \geq 1 \wedge i > k_j^* \\ \tau_{\max}, & \text{if } k_j^* = 0. \end{cases}$$

Proof: The proof is available on-line in [11]. ■

Based on Theorem 2, Corollary 2 derives the equations for all paths' RM-cell RTTs in a balanced binary tree under SSP.

Corollary 2: If a balanced binary-multicast-tree connection of height $m \geq 2$ is flow-controlled by SSP with $\Delta \geq 1$, then all paths' RM-cell RTTs, $\tau_b(j, \Delta)$, are the same in both steady and initial states and are determined by

$$\tau_b(j, \Delta) = \max_{j \in \{1, 2, \dots, m-1\}} \{\tau_u(j, \Delta)\} = \tau_{\max} \quad (7)$$

where $1 \leq j \leq 2^{m-1}$ and $\tau_u(j, \Delta)$ given by (6) is P_j 's RM-cell RTT for an unbalanced multicast tree of the same height.

Proof: The proof follows by letting $j = m - 1$ in (5), which leads to $k_{m-1}^* = 0$, and thus, $\tau_b(j, \Delta) = \tau_u(m-1, \Delta) = \tau_{\max}$ by (6). The detailed proof is available on-line in [11]. ■

Observations on Theorem 1 and Theorem 2: Comparing Theorem 1 and Theorem 2, we have the following observations.

- O1.** The initial-state RM-cell RTT in HBH is the same as in steady state while the SSP's initial-state RM-cell RTT, if any, is larger than, and lower-bounded by, steady-state RM-cell RTT. The initial state in SSP acts like a "warm-up" period for feedback RM cells to be synchronized at each branch, during which the initial-state RM-cell RTTs converge to their corresponding steady-state values. The warm-up periods for P_j are given by k_j^* in (5).
- O2.** For SSP, both initial- and steady-state RM-cell RTTs are upper-bounded by $\tau_{\max} = 2m$ [see **Claim 2** and **Claim 3** of Theorem 2 and (7)]. The increase rate of $\tau_u(j, \Delta)$ is $O(m)$ in the worst case. In contrast, in HBH, the RM-cell RTT $\tau_u(j, \Delta)$ is not upper-bounded by $\tau_{\max} = 2m$ [see (1) and (3)]. Also, $\tau_u(j, \Delta)$ of HBH is very sensitive to path length $j+1$ and RM-cell update interval Δ , and increases at a rate up to $O(m^2)$ in the worst case.

D. Numerical Comparison Between SSP and HBH

By using Theorems 1 and 2, the numerical analyses only focus on the unbalanced multicast tree to study the worst case of RM-cell RTT variations. Since P_j 's length is $j+1$ for $j = 1, \dots, m - 1$ (see the unbalanced tree in Fig. 3), $\tau_u(j, \Delta)$ is the RM-cell RTT for P_j with a length of $j+1$ in an unbalanced tree. Fig. 4(a) plots P_j 's RM-cell RTT $\tau_u(j, \Delta)$ versus P_j 's length $j+1$ and RM-cell interval Δ with tree height $m = 50$ for SSP and HBH. We observe that for both HBH and SSP, RM-cell RTTs $\tau_u(j, \Delta)$'s increase monotonically with $j+1$, Δ , and m . However, $\tau_u(j, \Delta)$ for HBH increases much faster, and is always larger, than that for SSP, and tends to blow up (as high

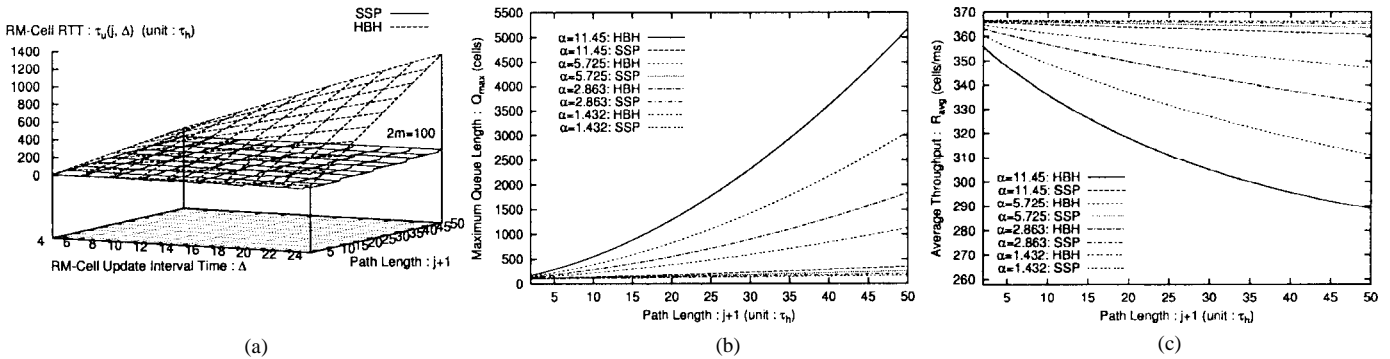


Fig. 4. Impact of P_j 's path length $j + 1$, tree height m , RM-cell interval Δ on P_j 's RM-cell RTT $\tau_u(j, \Delta)$, max. queue length Q_{max} , and average throughput \bar{R} . (a) $\tau_u(j, \Delta)$ versus $(j + 1, \Delta)$, $m = 50$. (b) Q_{max} versus $j + 1$ with $m = 50$. (c) \bar{R} versus $j + 1$ with $m = 50$.

as $1200 \tau_h$) as $j + 1, \Delta$, and m increase. In contrast, the increase of $\tau_u(j, \Delta)$ for SSP is very limited as $j + 1, \Delta$, and m get larger. Moreover, $\tau_u(j, \Delta)$ for SSP is upper-bounded by $2m = 100 = \tau_{max}$ as shown in Fig. 4(a), verifying Theorem 2. Thus, as shown in Fig. 4(a), the RM-cell RTT for SSP is virtually independent of path length and multicast-tree height, as compared to the HBH scheme. This is because: 1) the synchronization waiting time is much longer for HBH than that for SSP, and 2) the number of forward RM cells required for a feedback RM cell to return from the leaf node to the root in HBH is proportional to m , while in SSP, any single RM cell can return from the leaf node back to the root by itself.

As analyzed in [10] and [12], RM-cell RTTs, or path lengths, have a significant impact on the bottleneck maximum queue length Q_{max} and the average throughput \bar{R} . Using Q_{max} and \bar{R} as the functions [10] of RM-cell RTT and setting the multicast-tree bottleneck bandwidth $\mu = 155 \text{ Mb/s} \approx 367 \text{ cells/ms}$, $\tau_h = 0.1 \text{ ms}$, $\Delta = 4\tau_h = 0.4 \text{ ms}$, and $m = 50$, Fig. 4(b) and (c) plots Q_{max} and \bar{R} versus path length $j + 1$ while varying the rate-gain parameter α [10] for HBH and SSP. In HBH, Q_{max} is observed to increase dramatically [see Fig. 4(b)] while \bar{R} drops significantly [see Fig. 4(c)] as P_j 's path length and m increase. This undesirable trend worsens as α gets larger. In contrast, in SSP with the same parameters, both Q_{max} 's increase and \bar{R} 's drop are very small when $j + 1$ and m (even as α varies) increase. Again, Q_{max} and \bar{R} for SSP are found to be virtually independent of the path length and tree height variations. SSP is, therefore, more scalable than HBH in terms of maximum buffer requirement and average throughput when the multicast-tree topology changes.

IV. STATISTICAL DELAY ANALYSIS OF MULTICAST SIGNALING

So far, we have only focused on the deterministic properties of the feedback-signaling delay for each individual path within a multicast tree under HBH and SSP, respectively. However, in a real network the multicast tree bottleneck, defined as the most congested path (thus, dictating the source flow-control decisions) of a multicast tree [10], shifts randomly and dynamically from one path to another, depending on the network load distribution. To quantitatively characterize the delay performance of feedback-synchronization signaling for more realistic multicast scenarios, we now statistically analyze the feedback RM-cell delay performance of HBH and SSP across the *entire* multicast tree.

A. The Statistical Binary-Tree Model

Our statistical model builds on the recently proposed random early marking (REM) [13], [14] and the popular random early detection (RED) schemes [15].⁴ The REM and RED can also be extended to multicast. Moreover, unicast and multicast usually coexist in networks. In RED/REM, each router marks the packet's explicit congestion notification (ECN) [16] bit with a probability that is an exponential (REM) or linear (RED) function of the average queue length at the output link. Like in REM/RED, our statistical model also assumes that the ECN-bit markings at all links are independent. We can make such an assumption because the total number of connections in real networks is usually large, and their traffic routings/patterns and network topologies are typically heterogeneous, making the markings at different links virtually independent. However, if the total number of connections is small or link-bandwidth/buffer-size and traffic routings are almost the same, the independent-link assumption may cause the approximation error, which requires the further investigation [17]. Also, notice that this independent-link assumption does not affect the evaluation of the *relative* delay performance improvement of SSP over HBH.

Our statistical analysis will focus only on the unbalanced multicast tree since it represents the worst case, and thus, its analysis provides the lower bound of delay performance for multicast signaling. On the other hand, our statistical model captures more realistic multicast scenarios by allowing multiple concurrent bottleneck links/paths within a multicast tree. Moreover, to be able to handle any arbitrary size of the unbalanced multicast tree and make the analysis complete, the statistical model allows the multicast-tree height m to be arbitrarily large and include ∞ as its limiting case. To formulate the statistical analysis, we introduce the following definition.

Definition 1: The random-marking-based unbalanced binary multicast tree of height m consists of a set \mathcal{L} of links which satisfy the following conditions:

C1. All links in \mathcal{L} are labeled as shown in Fig. 5(a) for $m < \infty$ (e.g., $m = 4$) and Fig. 5(b) for $m \rightarrow \infty$, such that

$$L_i \in \mathcal{L} \triangleq \begin{cases} \{L_1, L_2, \dots, L_{2m-1}\}, & \text{if } m < \infty \\ \{L_1, L_2, \dots, L_\infty\}, & \text{if } m \rightarrow \infty. \end{cases} \quad (8)$$

⁴The analytical technique developed in this paper is also applicable to cases where a link's random congestion state is caused by other flow-control schemes than REM and RED.

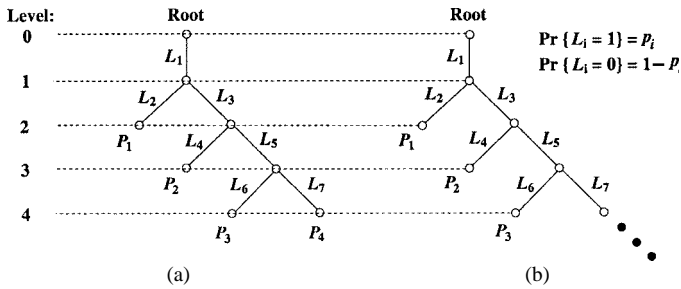


Fig. 5. Random-marking based unbalanced binary multicast-tree model. (a) Unbalanced tree: height $m = 4$. (b) Unbalanced tree: height $m = \infty$.

C2. $\forall L_i \in \mathcal{L}$, the probability $p_i \in (0, 1)$ that L_i is marked as a bottleneck link (with the ECN-bit set) is specified by

$$p_i = \begin{cases} 1 - \phi^{-\gamma \bar{q}_i}, & \text{if REM is used} \\ \left[\frac{\bar{q}_i - \min_{\text{th}}}{\max_{\text{th}} - \min_{\text{th}}} \right] p_{\max}, & \text{if RED is used} \end{cases} \quad (9)$$

where \bar{q}_i is L_i 's average queue size, $\gamma > 0$ is step size, $\phi > 1$ for REM, p_{\max} is the maximum marking probability, and $\max_{\text{th}}(\min_{\text{th}})$ is the high (low) queue size threshold of RED.

C3. Markings at all links in \mathcal{L} are independent [13]–[15]. ■

In unicast ABR service, the source rate is regulated by the feedback from the most congested link/switch having the minimum *available residual bandwidth* on the path from source to destination. A natural extension of this strategy to multicast ABR is to adjust the source rate to the available bandwidth share that can be supported by the *most congested path*, which contains a link/switch having the minimum *available residual bandwidth* in a multicast tree. This is the key for lossless transmission. Thus, the feedback from the most congested path governs the source rate. Consequently, all multicast signaling algorithms use an OR rule (e.g., Fig. 2) at a branch point to consolidate the feedbacks from the different downstream branches. Moreover, since there can be multiple bottleneck paths (links) simultaneously in networks, we need to identify the path that dictates the dynamics of the entire multicast tree. Clearly, based on the OR rule, the *shortest* bottleneck path in a multicast tree dominates the source's flow-control actions and feedback RTT. To explicitly model this feature, we introduce the following definition.

Definition 2: Among all concurrent bottleneck paths in a multicast tree, the bottleneck path of minimum length is called the *dominant bottleneck path*, and its RM-cell RTT is called the *multicast-tree bottleneck RTT*. ■

B. Statistical Properties of Feedback Signaling Delays

By using Definitions 1 and 2, Theorem 3 derives the probability of each path becoming the dominant bottleneck path.

Theorem 3: If an unbalanced multicast tree of height m as defined by Definition 1 is flow-controlled by SSP and HBH, respectively, then the following claims hold:

Claim 1: If $m \rightarrow \infty$, then there exists *one and only one* dominant bottleneck path, and the probability $\psi(P_k, \infty)$ that path P_k becomes the dominant bottleneck path, is given by

$$\psi(P_k, \infty) = (p_{2k-1} + p_{2k} - p_{2k-1}p_{2k}) \prod_{i=1}^{2(k-1)} (1 - p_i) \quad (10)$$

where $k = 1, \dots, \infty$ and $0 < p_i < 1$ is given by (9).

Claim 2: There is *one and only one* dominant bottleneck path and $\psi(P_k, \infty)$ given by (10) satisfies

$$\begin{cases} \lim_{k \rightarrow \infty} \psi(P_k, \infty) = 0 \\ \lim_{m \rightarrow \infty} \sum_{k=1}^m \psi(P_k, \infty) = 1. \end{cases} \quad (11)$$

Claim 3: If $m < \infty$, then there is *at most one* dominant bottleneck path, and the probability $\psi(P_k, m)$ that path P_k becomes the dominant bottleneck path is determined by

$$\psi(P_k, m) = \begin{cases} p_1 + p_2 - p_1 p_2, & \text{if } k=1 \\ (p_{2k-1} + p_{2k} - p_{2k-1} p_{2k}) \prod_{i=1}^{2(k-1)} (1 - p_i), & \text{if } k \leq m-1 \\ p_{2m-1} \prod_{i=1}^{2(m-1)} (1 - p_i), & \text{if } k=m \end{cases} \quad (12)$$

where p_i is given by (9), and $\psi(P_k, m)$, given by (12), satisfies

$$\sum_{k=1}^m \psi(P_k, m) \leq 1. \quad (13)$$

Proof: The proof is available on-line in [11]. ■

Remarks on Theorem 3: The first part of (11) is expected, as a longer bottleneck path is always dominated by a coexisting shorter bottleneck path. Thus, when $k \rightarrow \infty$ as $m \rightarrow \infty$, P_∞ is always dominated by a shorter bottleneck path since $0 < p_i < 1$, $i = 1, \dots, \infty$, where p_i statistically represents the traffic load level at link L_i . The second part of (11) also makes sense because as tree height $m \rightarrow \infty$ and $0 < p_i < 1$, there always is (with probability 1) one and only one dominant bottleneck path in a multicast tree. Moreover, the second part of (11) shows that $\psi(P_k, \infty)$ satisfies the Probability Mass Function (PMF) normalization condition, validating that $\psi(P_k, \infty)$ given by (10) is a valid PMF. On the other hand, (13) implies the possibility of nonexistence of any dominant bottleneck path in the multicast tree if $m < \infty$, which is also expected because of link marking probability $p_i \in (0, 1)$.

Using Theorem 3 and assuming $p_i = p \forall i$,⁵ Theorem 4 gives the probabilities of the dominant bottleneck path, analyzes their dynamics, and derives the first and second moments of the multicast-signaling delay under HBH and SSP for the homogeneous case: $p_i = p \forall i$ —a special case which suffices to assess the relative delay-performance improvement of SSP over HBH.

Theorem 4: Let a multicast tree defined by Definition 1 be flow-controlled by SSP and HBH, respectively, with the RM-cell update interval Δ . If $m < \infty$ and $0 < p_i = p < 1 \forall i \in \{1, \dots, 2m-1\}$, then the following claims hold:

⁵The analytical results derived from this special case can be easily extended to a more general case where p_i 's are different for $i = 1, \dots, 2m-1$. The derivations for the generalized case remain almost the same as derived here.

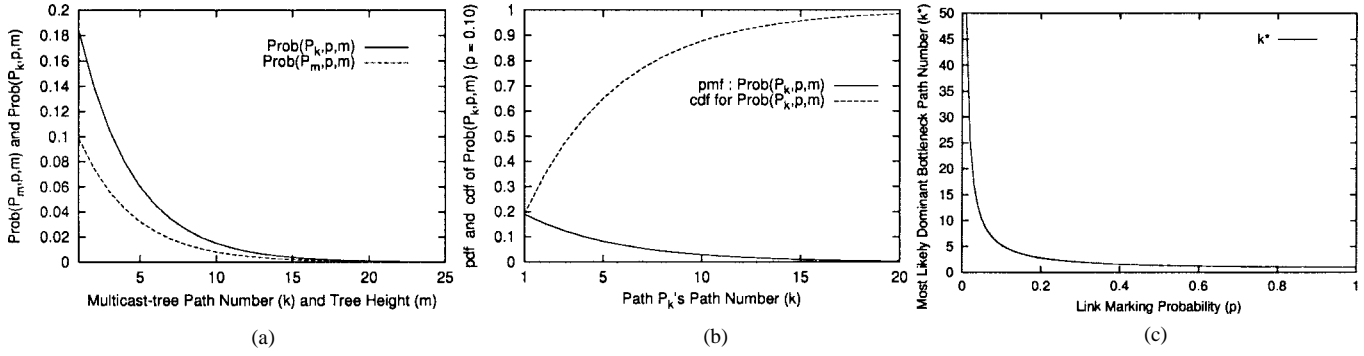


Fig. 6. Probability distributions of dominant bottleneck path. (a) $\psi(P_k, p, m)$ versus k and m . (b) $\psi(P_k, p, m)$ and its CDF versus k . (c) k^* versus p .

Claim 1: The probability $\psi(P_k, p, m)$ that path P_k becomes the dominant bottleneck path is determined by

$$\psi(P_k, p, m) = \begin{cases} p(2-p)(1-p)^{2(k-1)}, & \text{if } k \leq m-1 \\ p(1-p)^{2(m-1)}, & \text{if } k = m. \end{cases} \quad (14)$$

Claim 2: $\psi(P_k, p, m)$ attains the *unique* maximum given by the following equations:

$$\psi^*(P_k, p^*, m) = \begin{cases} \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-1}, & \text{if } k \leq m-1 \\ \frac{1}{2m-1} \left(\frac{2m-2}{2m-1}\right)^{2(m-1)}, & \text{if } k = m \end{cases} \quad (15)$$

where the *unique* marking probability maximizer p^* is given by

$$p^* \triangleq \arg \max_{0 < p < 1} \psi(P_k, p, m) = \begin{cases} 1 - \sqrt{\frac{k-1}{k}}, & \text{if } k \leq m-1 \\ \frac{1}{2m-1}, & \text{if } k = m. \end{cases} \quad (16)$$

Claim 3: The means of multicast-tree RM-cell RTT, denoted by $\bar{\tau}_{\text{SSP}}(m)$ and $\bar{\tau}_{\text{HBH}}(m)$ for SSP and HBH, respectively, are determined by the following equations:

$$\bar{\tau}_{\text{SSP}}(m) = 2m \left[1 - (1-p)^{2(m-1)} \right] - \Delta(2p-p^2) \cdot \sum_{k=1}^{m-1} \left[\frac{2(m-k-1)}{\Delta} \right] (1-p)^{2(k-1)} + 2mp(1-p)^{2(m-1)} \quad (17)$$

$$\bar{\tau}_{\text{HBH}}(m) = 2 \left[1 - (1-p)^{2(m-1)} \right] + \frac{\Theta(\Delta)}{2p-p^2} \left[(m-1) \cdot (1-p)^{2m} - m(1-p)^{2(m-1)} + 1 \right] + p(1-p)^{2(m-1)} [2 + (m-1)\Theta(\Delta)] \quad (18)$$

where $\Theta(\Delta)$ is defined by (2).

Claim 4: The variances of multicast-tree RM-cell RTT, denoted by $\sigma_{\text{SSP}}^2(m)$ and $\sigma_{\text{HBH}}^2(m)$ for SSP and HBH, respectively, are determined by the following equations:

$$\begin{aligned} \sigma_{\text{SSP}}^2(m) &= 4m^2 \left[1 - (1-p)^{2(m-1)} \right] - (2p-p^2) \\ &\quad \cdot \left\{ 4m\Delta \sum_{k=1}^{m-1} \left[\frac{2(m-k-1)}{\Delta} \right] (1-p)^{2(k-1)} \right. \\ &\quad \left. - \Delta^2 \sum_{k=1}^{m-1} \left[\frac{2(m-k-1)}{\Delta} \right]^2 (1-p)^{2(k-1)} \right\} \\ &\quad - \left\{ 2m \left[1 - (1-p)^{2(m-1)} \right] - \Delta(2p-p^2) \right. \\ &\quad \cdot \sum_{k=1}^{m-1} \left[\frac{2(m-k-1)}{\Delta} \right] (1-p)^{2(k-1)} + 2mp \\ &\quad \left. \cdot (1-p)^{2(m-1)} \right\}^2 + 4m^2 p(1-p)^{2(m-1)} \quad (19) \end{aligned}$$

$$\begin{aligned} \sigma_{\text{HBH}}^2(m) &= 4 \left[1 - (1-p)^{2(m-1)} \right] + \frac{4\Theta(\Delta)}{2p-p^2} \\ &\quad \cdot \left[(m-1)(1-p)^{2m} - m(1-p)^{2(m-1)} + 1 \right] \\ &\quad + \frac{\Theta^2(\Delta)}{(2p-p^2)^2} \left\{ (2p-p^2) \left[1 - m^2(1-p)^{2(m-1)} \right. \right. \\ &\quad \left. \left. + (m^2-1)(1-p)^{2m} \right] + 2 \left[(1-p)^2 - m(1-p)^{2m} \right. \right. \\ &\quad \left. \left. + (m-1)(1-p)^{2(m+1)} \right] \right\} + p(1-p)^{2(m-1)} \\ &\quad \cdot [2 + (m-1)\Theta(\Delta)]^2 - \left\{ \frac{\Theta(\Delta)}{2p-p^2} \left[(m-1) \right. \right. \\ &\quad \left. \left. \cdot (1-p)^{2m} - m(1-p)^{2(m-1)} + 1 \right] \right. \\ &\quad \left. + 2 \left[1 - (1-p)^{2(m-1)} \right] + (1-p)^{2(m-1)} p \right. \\ &\quad \left. \cdot [2 + (m-1)\Theta(\Delta)] \right\}^2 \quad (20) \end{aligned}$$

where $\Theta(\Delta)$ is defined by (2).

Proof: The proof is available on-line in [11]. ■

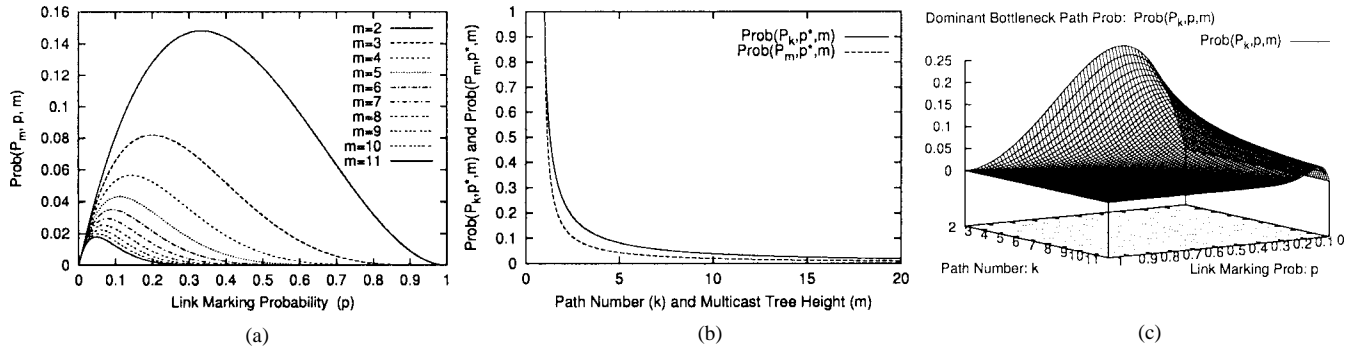


Fig. 7. Properties of dominant bottleneck path probability-distribution functions. (a) $\psi(P_m, p, m)$ versus p . (b) $\psi^*(P_k, p^*, m)$ versus k and m . (c) $\psi(P_k, p, m)$ versus (p, k) .

Remarks on Theorem 4: Assuming $p = p_i \forall i$, and observing (14), $\psi(P_k, p, m)$ is found to be a strictly monotonic decreasing function of path length k and the multicast-tree height m . Fig. 6(a) plots the PMF $\psi(P_k, p, m)$ versus k and m , also confirming the above observations. This is not surprising because a longer bottleneck path is more likely to be dominated by a shorter one. In fact, this is a desired feature of SSP because the SSP's effective multicast-tree RM-cell RTT is upper-bounded by the maximum RM-cell RTT ($2m$) and is virtually independent of the multicast-tree height. Fig. 6(b) plots the Cumulative Distribution Function (CDF) for $\psi(P_k, p, m)$, which converges to 1 as $k, m \rightarrow \infty$, confirming that $\psi(P_k, p, m)$ is a valid PMF.

Equation (14) also indicates that path P_k 's $\psi(P_k, p, m)$ is not a monotonic function of p . $\psi(P_k, p, m)$ attains the maximum $\psi^*(P_k, p^*, m)$ given by (15), as a function of p , which statistically reflects the traffic load. Solving the first part of (16) for k with a given p , we get $k^* \triangleq \lfloor 1/(p(2-p)) \rfloor$, where P_{k^*} is most likely to be the dominant bottleneck path for a given p . When p departs from p^* , $\psi(P_k, p, m)$ converges to zero as either $p \rightarrow 0$ or $p \rightarrow 1$, as shown in (14). This is expected because a small p implies that the traffic load is low, making the longest path the most likely dominant bottleneck path P_{k^*} , while a large p , implying a heavy traffic load, makes the shortest path the most likely dominant bottleneck path P_{k^*} . If the path P_k has a length somewhere between the longest and shortest paths, $\psi(P_k, p, m)$ converges to zero as $p \rightarrow 0$ or 1.

Also, p^* and $\psi^*(P_k, p^*, m)$ are both the monotonic decreasing functions of k , because $\psi(P_k, p, m)$ is a strictly decreasing function of k and, as k increases, p must decrease to ensure a longer path to be the most-likely dominant bottleneck path, which, in turn, reduces p^* . Fig. 6(c) plots the path number k^* of the most likely dominant bottleneck path versus p based on $k^* (= \lfloor 1/(p(2-p)) \rfloor)$, and shows that k^* decreases as p increases. That is, the higher the traffic load, the shorter the most-likely dominant bottleneck path. This makes SSP based on REM or RED very suitable for multicast flow control, since the multicast-tree RM-cell RTT statistically adapts to network traffic load variations. Fig. 7(b) plots $\psi^*(P_k, p^*, m)$ versus k and m . We observe that $\psi^*(P_k, p^*, m)$ drops very quickly when k and m increase, making the longer path have a relatively smaller probability to become the most-likely dominant bottleneck path as compared with the shorter path [see also Fig. 7(c)].

By applying (14), Fig. 7(a) shows how the network traffic load and multicast-tree height affect the dominant bottleneck

path probability by plotting $\psi(P_m, p, m)$ versus p with m varying. Fig. 7(a) shows that there exists a unique maximum $\psi^*(P_m, p^*, m)$ at $p = p^*$ for each given m . As m increases, the entire $\psi(P_m, p, m)$ decreases, confirming the above observations. In Fig. 7(c), $\psi(P_k, p, m)$ is plotted as a function of two independent variables, p and k . We observe that for each given path P_k , $\psi(P_k, p, m)$ attains its unique maximum at p^* while both $\psi^*(P_k, p^*, m)$ and p^* are monotonic decreasing functions of k . This also confirms our analytical findings.

$\bar{\tau}_{\text{HBH}}(m)$ and $\bar{\tau}_{\text{SSP}}(m)$ are important performance metrics for multicast signaling since they represent the average RM-cell RTT of a multicast tree. Clearly, small $\bar{\tau}_{\text{HBH}}(m)$ and $\bar{\tau}_{\text{SSP}}(m)$ are desired because a small feedback delay can improve feedback accuracy and system responsiveness. Equations (17) and (18) indicate that both $\bar{\tau}_{\text{SSP}}(m)$ and $\bar{\tau}_{\text{HBH}}(m)$ are functions of Δ and m . So, the selection of Δ affects the average multicast signaling delay. On the other hand, $\sigma_{\text{HBH}}^2(m)$ and $\sigma_{\text{SSP}}^2(m)$ represent the variation amplitudes around the average multicast signaling delay for HBH and SSP, respectively. Also, small $\sigma_{\text{HBH}}^2(m)$ and $\sigma_{\text{SSP}}^2(m)$ are desired because they impact the stability and transient performance of flow control. Likewise, (19) and (20) indicate that $\sigma_{\text{SSP}}^2(m)$ and $\sigma_{\text{HBH}}^2(m)$ are functions of Δ and m . So, Δ also affects the variation of average multicast signaling delay.

Corollary 3, given below, following directly from Theorem 4 by letting $m \rightarrow \infty$ for the (homogeneous) case of $p_i = p \forall i$, gives the probability of the dominant bottleneck path, characterizes its dynamics, and derives the first and second moments of multicast signaling delay for HBH and SSP, respectively.

Corollary 3: Let a multicast tree defined by Definition 1 be flow-controlled by HBH and SSP, respectively, with the RM-cell update interval Δ . If the tree height $m \rightarrow \infty$ and link-marking probability $0 < p_i = p < 1 \forall i \in \{1, \dots, \infty\}$, then the following claims hold:

Claim 1: The probability $\psi(P_k, p, \infty)$ that P_k becomes the dominant bottleneck path is determined by

$$\psi(P_k, p, \infty) = p(2-p)(1-p)^{(2k-1)} \quad (21)$$

where $k = 1, \dots, \infty$. Also, $\psi(P_k, p, \infty)$ given in (21) satisfies the following normalization condition:

$$\lim_{m \rightarrow \infty} \sum_{k=1}^m \psi(P_k, p, \infty) = 1. \quad (22)$$

Claim 2: $\psi(P_k, p, \infty)$ attains the unique maximum given by

$$\psi^*(P_k, p^*, \infty) = \begin{cases} \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-1}, & \text{if } k < \infty \\ 0, & \text{if } k \rightarrow \infty \end{cases} \quad (23)$$

where the marking probability maximizer p^* is given by

$$p^* \triangleq \arg \max_{0 < p < 1} \psi(P_k, p, \infty) = \begin{cases} 1 - \sqrt{\frac{k-1}{k}}, & \text{if } k < \infty \\ 0, & \text{if } k \rightarrow \infty \end{cases} \quad (24)$$

where $k = 1, \dots, \infty$.

Claim 3: The means of multicast-tree RM-cell RTT, denoted by $\bar{\tau}_{\text{HBH}}(\infty)$ and $\bar{\tau}_{\text{SSP}}(\infty)$ for the HBH and SSP schemes, respectively, exist and are determined as follows:

$$\bar{\tau}_{\text{HBH}}(\infty) = \frac{4p - 2p^2 + \Theta(\Delta)}{2p - p^2} \quad (25)$$

$$\bar{\tau}_{\text{SSP}}(\infty) = \lim_{m \rightarrow \infty} \left\{ 2m - \frac{\Delta(2p - p^2)}{(1-p)^2} \cdot \sum_{k=1}^{m-1} \left\lfloor \frac{2(m-k-1)}{\Delta} \right\rfloor (1-p)^{2k} \right\} \quad (26)$$

where $\Theta(\Delta)$ is defined by (2).

Claim 4: The variances of multicast-tree RM-cell RTT, denoted by $\sigma_{\text{HBH}}^2(\infty)$ and $\sigma_{\text{SSP}}^2(\infty)$ for the HBH and SSP schemes, respectively, exist and are determined as follows:

$$\sigma_{\text{HBH}}^2(\infty) = \frac{(1-p)^2 \Theta^2(\Delta)}{(2-p)^2 p^2} \quad (27)$$

$$\sigma_{\text{SSP}}^2(\infty) = \lim_{m \rightarrow \infty} \left\{ 4m^2 - \frac{(2p - p^2)}{(1-p)^2} \left\{ 4m\Delta \cdot \sum_{k=1}^{m-1} \left\lfloor \frac{2(m-k-1)}{\Delta} \right\rfloor (1-p)^{2k} - \Delta^2 \cdot \sum_{k=1}^{m-1} \left\lfloor \frac{2(m-k-1)}{\Delta} \right\rfloor^2 (1-p)^{2k} \right\} - \left\{ 2m - \frac{\Delta(2p - p^2)}{(1-p)^2} \cdot \sum_{k=1}^{m-1} \left\lfloor \frac{2(m-k-1)}{\Delta} \right\rfloor (1-p)^{2k} \right\}^2 \right\} \quad (28)$$

where $\Theta(\Delta)$ is defined by (2).

Proof: The proof is available on-line in [11]. ■

Remarks on Corollary 3: While m does not attain ∞ in real networks, Corollary 3 ensures the existence and convergence of finite means and variances for the dominant bottleneck-path probability distribution derived in Theorem 3. This makes our statistical analysis complete and meaningful. This corollary also states the trend of means and variances of SSP and HBH when

m is large. For instance, by (25) we observe that $\bar{\tau}_{\text{HBH}}(\infty)$ is proportional to the RM-cell interval Δ (or $\Theta(\Delta)$) for a given p . In contrast, by (26), we observe that $\bar{\tau}_{\text{SSP}}(\infty)$ is upper-bounded by the maximum RM-cell RTT, $2m$, regardless of p and Δ . Likewise, (27) indicates that $\sigma_{\text{HBH}}^2(\infty)$ is proportional to $\Theta^2(\Delta)$ or Δ^2 while $\sigma_{\text{SSP}}^2(\infty)$ is also upper-bounded as shown in (28).

C. Numerical Evaluation of Delay Statistics for SSP and HBH

Using the analytical results derived in Section IV-B, we numerically compare the statistical delay properties for HBH and SSP. The plots of $\bar{\tau}_{\text{HBH}}(m)$ and $\bar{\tau}_{\text{SSP}}(m)$ versus tree height m with Δ varying, in Fig. 8(a), show that $\bar{\tau}_{\text{HBH}}(m)$ is much larger (≈ 6 times), and increases much faster, than $\bar{\tau}_{\text{SSP}}(m)$. Moreover, $\bar{\tau}_{\text{HBH}}(m)$ is more sensitive to Δ than $\bar{\tau}_{\text{SSP}}(m)$. Unlike $\bar{\tau}_{\text{HBH}}(m)$, Fig. 8(a) shows that $\bar{\tau}_{\text{SSP}}(m)$ is virtually independent of m . Likewise, we have the similar observations from Fig. 8(b) that plots delay variations $\sigma_{\text{HBH}}(m)$ and $\sigma_{\text{SSP}}(m)$ versus m with Δ varying. Thus, the multicast signaling delay and delay variations for SSP scales much better than those for HBH in terms of the multicast-tree height and structure.

With $m = 25$ and $\Delta = 20$, Fig. 8(c) plots $\bar{\tau}_{\text{HBH}}(m)$ and $\bar{\tau}_{\text{SSP}}(m)$ versus link marking probability p of the network traffic load. We observe that both $\bar{\tau}_{\text{HBH}}(m)$ and $\bar{\tau}_{\text{SSP}}(m)$ have their respective unique maximum. This is expected since $\psi(P_k, p, m)$ has the unique maximum with respect to p . Moreover, the maximum of $\bar{\tau}_{\text{HBH}}(m)$ is found to be about eight times larger than that of $\bar{\tau}_{\text{SSP}}(m)$ while the maximizer ($p = 0.037$) of $\bar{\tau}_{\text{HBH}}(m)$ is slightly smaller than that ($p = 0.044$) of $\bar{\tau}_{\text{SSP}}(m)$. So, the SSP significantly outperforms the HBH in terms of the first moment of multicast-signaling delays. When p increases beyond the maximizer, both $\bar{\tau}_{\text{HBH}}(m)$ and $\bar{\tau}_{\text{SSP}}(m)$ decrease quickly since the dominant bottleneck path tends to be short as p increases. This implies that the multicast-signaling delay of SSP can statistically adapt itself to the network traffic-load dynamic variations. Likewise, we have the similar observations from Fig. 9(a) that plots $\sigma_{\text{HBH}}(m)$ and $\sigma_{\text{SSP}}(m)$ versus p . This implies that the multicast signaling delay variation of SSP is statistically much more stable than that for HBH.

To examine the asymptotic properties of $\bar{\tau}_{\text{HBH}}(m)$, $\bar{\tau}_{\text{SSP}}(m)$, $\sigma_{\text{HBH}}(m)$, and $\sigma_{\text{SSP}}(m)$ as tree height m gets large, their numerical solutions as the function of p (with $\Delta = 20$, for $m = 25 < \infty$ and $m \rightarrow \infty$) are plotted in Fig. 9(b) and (c), respectively, showing the convergence of $\bar{\tau}_{\text{HBH}}(m) \rightarrow \bar{\tau}_{\text{HBH}}(\infty)$, $\bar{\tau}_{\text{SSP}}(m) \rightarrow \bar{\tau}_{\text{SSP}}(\infty)$, $\sigma_{\text{HBH}}(m) \rightarrow \sigma_{\text{HBH}}(\infty)$, and $\sigma_{\text{SSP}}(m) \rightarrow \sigma_{\text{SSP}}(\infty)$, asymptotically as $m \rightarrow \infty$. We obtain the following observations.

O1. As $m \rightarrow \infty$, the extreme points of p for $\bar{\tau}_{\text{HBH}}(\infty)$, $\bar{\tau}_{\text{SSP}}(\infty)$, $\sigma_{\text{HBH}}(\infty)$, and $\sigma_{\text{SSP}}(\infty)$ disappear, and they become monotonic decreasing functions of p . This is expected since $p^* = 0$ and $\psi^*(P_k, p^*, \infty) = 0$ as $k, m \rightarrow \infty$ as shown in (24) and (23).

O2. $\bar{\tau}_{\text{HBH}}(m)$ and $\bar{\tau}_{\text{SSP}}(m)$ converge to $\bar{\tau}_{\text{HBH}}(\infty)$ and $\bar{\tau}_{\text{SSP}}(\infty)$, respectively, and both are lower-bounded by the same bound for each scheme, as $p \rightarrow 1$. This also confirms our analytical results, because $\lim_{p \rightarrow 1} \bar{\tau}_{\text{HBH}}(m) = \lim_{p \rightarrow 1} \bar{\tau}_{\text{HBH}}(\infty) = 2 + \Theta(\Delta)$ by (25), which is the lower bound of HBH's RM-cell RTT given by (1) for path P_1 , and $\lim_{p \rightarrow 1} \bar{\tau}_{\text{SSP}}(m) = \lim_{p \rightarrow 1} \bar{\tau}_{\text{SSP}}(\infty) = 2m - \Delta \lfloor 2(m-2)/\Delta \rfloor$ by (26),

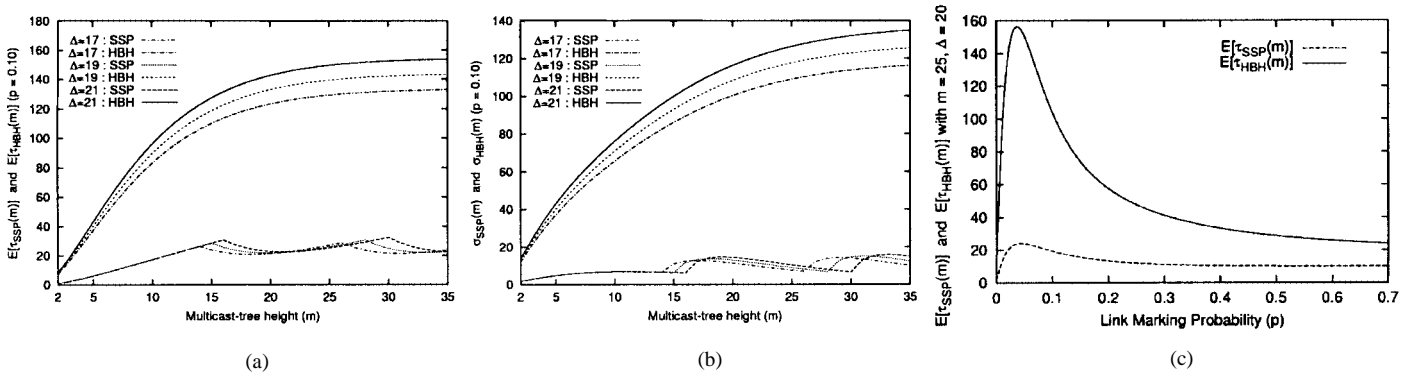


Fig. 8. Comparison of means and variances of multicast-tree RTT between HBH and SSP schemes as the multicast-tree height $m < \infty$. (a) $\bar{\tau}_{SSP}(m)$, $\bar{\tau}_{HBH}(m)$ versus m . (b) $\sigma_{SSP}(m)$, $\sigma_{HBH}(m)$ versus m . (c) $\bar{\tau}_{SSP}(m)$, $\bar{\tau}_{HBH}(m)$ versus p .

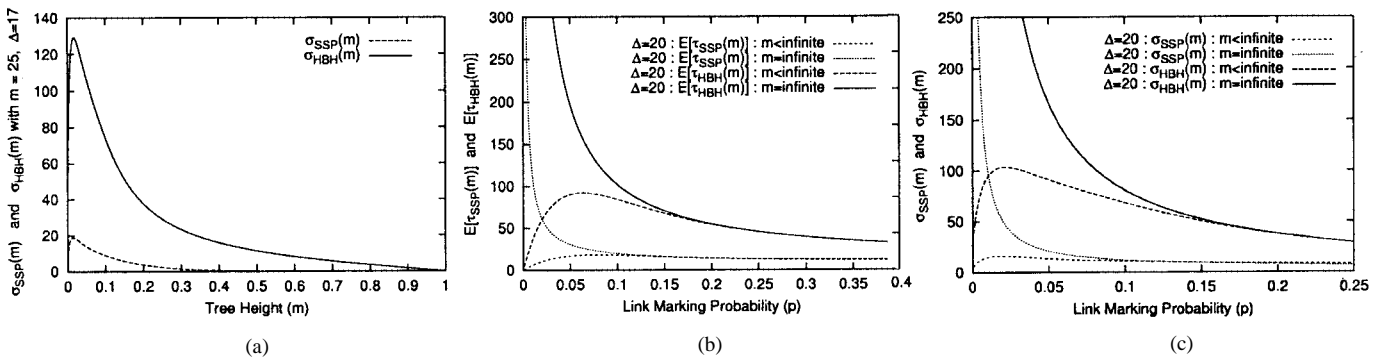


Fig. 9. Statistical properties (as $m < \infty$) and asymptotic properties of multicast-tree RTT for HBH and SSP as the multicast-tree height $m \rightarrow \infty$. (a) $\sigma_{HBH}(m)$, $\sigma_{SSP}(m)$ versus p . (b) $\bar{\tau}_{SSP}(\infty)$, $\bar{\tau}_{HBH}(\infty)$ versus p . (c) $\sigma_{SSP}(\infty)$, $\sigma_{HBH}(\infty)$ versus p .

which is the lower bound of SSP's RM-cell RTT given by (6) for path P_1 .

O3. $\sigma_{HBH}(m)$ and $\sigma_{SSP}(m)$ converge to $\sigma_{HBH}(\infty)$ and $\sigma_{SSP}(\infty)$ as $m \rightarrow \infty$, respectively, and both converge to 0, as $p \rightarrow 1$. This also confirms our analytical results, because from (20) and (27), we have $\lim_{p \rightarrow 1} \sigma_{HBH}^2(m) = \lim_{p \rightarrow 1} \sigma_{HBH}^2(\infty) = 0$, and from (19) and (28), we have $\lim_{p \rightarrow 1} \sigma_{SSP}^2(m) = \lim_{p \rightarrow 1} \sigma_{SSP}^2(\infty) = 0$.

O4. When m gets large, $\bar{\tau}_{HBH}(m)$ and $\bar{\tau}_{SSP}(m)$ drop very quickly as p increases; e.g., as $p \geq 0.1$, i.e., when the network is busy for more than 10% of the time, $\bar{\tau}_{SSP}(m)$ already converges closely to the lower bound ($\lim_{p \rightarrow 1} \bar{\tau}_{SSP}(m) = 2m - \Delta[2(m-2)/\Delta] = 10$). In contrast, $\bar{\tau}_{HBH}(m)$ does not converge closely to its lower bound ($\lim_{p \rightarrow 1} \bar{\tau}_{HBH}(m) = 2 + \Delta = 22$) until traffic load p is beyond 40%–50%. Likewise, a similar behavior is found for $\sigma_{HBH}(m)$ and $\sigma_{SSP}(m)$. Thus, SSP's multicast signaling delay and its variation converge to the lower bound much faster than the HBH's, making SSP adapt to the network traffic load much faster than HBH in terms of multicast signaling delay.

V. SIMULATION RESULTS

Using the NetSim [18], we simulate the network with concurrent multiple multicast/unicast virtual circuits (VCs) and multiple bottlenecks to study and contrast the statistical delay performance of SSP and HBH. Fig. 10 shows the simulated network model, consisting of 16 switches, $SW_1, SW_2, \dots, SW_{16}$, connected via 15 links L_1, \dots, L_{15} as an unbalanced multicast tree

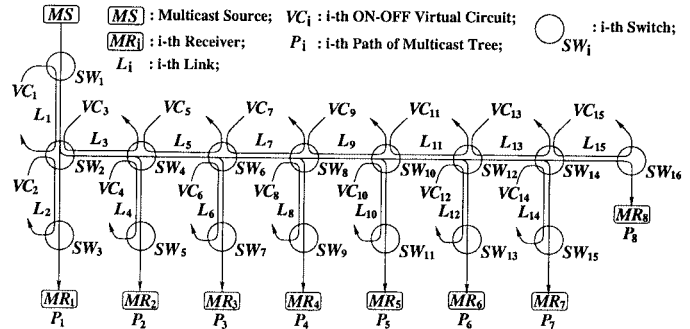


Fig. 10. Simulation model for delay analysis of bottleneck RTT with $m = 8$.

TABLE I
RTT (MILLISECONDS) OF EACH PATH FOR
THE SIMULATED NETWORK MODEL

Path Name: P_j	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8
$\tau_{ii}^{HBH}(j, \Delta)$	8	14	20	26	32	38	44	44
$\tau_{ii}^{SSP}(j, \Delta)$	4	10	10	10	16	16	16	16
$\tau_{min}^{Limit}(j)$	4	6	8	10	12	14	16	16

of height $m = 8$. The simulated network contains one multicast connection with a persistent ABR traffic source, starting from sender MS to eight receivers MR_1, \dots, MR_8 through the multicast tree and forming eight paths P_1, \dots, P_8 . We set up 15 independent *random* ON-OFF unicast VCs, each of which is denoted by VC_i going through link L_i for $i = 1, \dots, 15$. VC_i acts as independent random cross-traffic sharing L_i 's bandwidth with the multicast connection. The activity intensity of VC_i determines the congestion marking probability p_i for link L_i . When

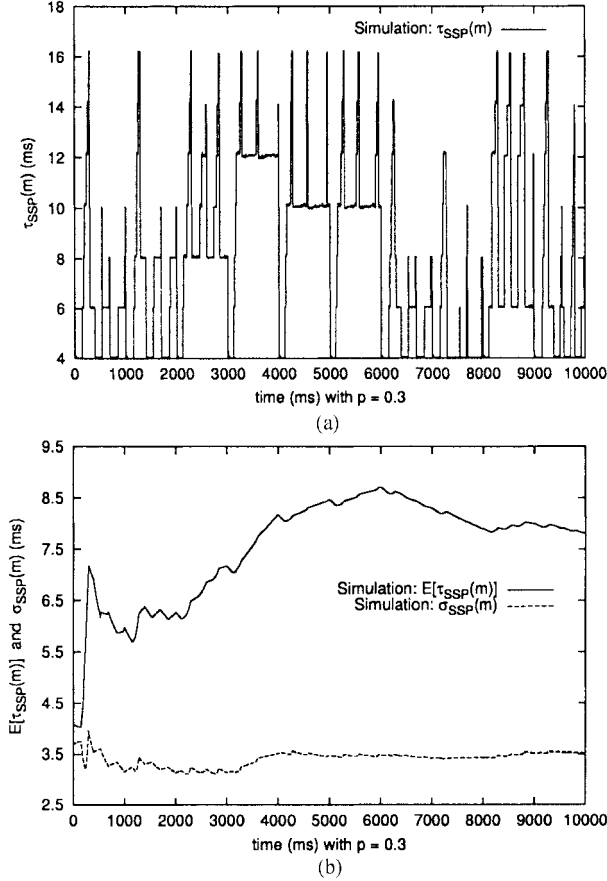


Fig. 11. Simulated multicast-tree bottleneck RM-cell RTTs and their statistics for SSP. (a) Instant multicast-tree delay $\tau_{SSP}(t)$ versus t . (b) Statistics $\bar{\tau}_{SSP}(t)$ and $\sigma_{SSP}(t)$ versus t .

these bursty VC_i 's randomly switch between ON and OFF states, the multicast-tree bottleneck changes randomly from one path to another, making the multicast-tree bottleneck RM-cell RTT vary dynamically and randomly, as shown below.

We set the bandwidth $\mu_i = \mu = 155$ Mb/s (ATM Forum standards) for each $L_i \forall i$, and link (or hop) delay $\tau_h = 1$ ms (milliseconds) to approximate the propagation delay of 300 km, a typical single-hop distance for the ATM wide area networks [19]. Thus, all paths' RTTs, $\tau_u^{HBH}(j, \Delta)$ and $\tau_u^{SSP}(j, \Delta)$, are obtained by (1) and (6), respectively, and are listed in Table I. The last row in Table I gives the physically limiting minimum for RTT, $\tau_{\min}^{\text{Limit}}(j)$, of each path. The RM-cell interval Δ is set to 6 ms as a fraction of $\tau_{\max} = 2m = 16$ ms. A small Δ yields more up-to-date feedback at expense of high signaling bandwidth overhead. However, too small a Δ may cause an oscillatory dynamics of flow control, degrading the system stability. On the other hand, too large a Δ cannot timely convey the feedback to the multicast source, thus making the feedback information stale. Thus, Δ needs to be finely tuned by making a tradeoff.

The link marking probabilities, p_i ($i = 1, \dots, 15$), are generated by 15 independent $[0, 1]$ -uniformly distributed random-number generators, R_i , which control the 15 cross-traffic VC_i 's ON-OFF states and their activity intensities. The entire simulation observation time T is divided into N repeated observation slots T_k , $k = 1, \dots, N$, and $T = \sum_{k=1}^N T_k$. At the beginning of each observation slot T_k , $k = 1, \dots, N$, VC_i , $i = 1, \dots, 15$,

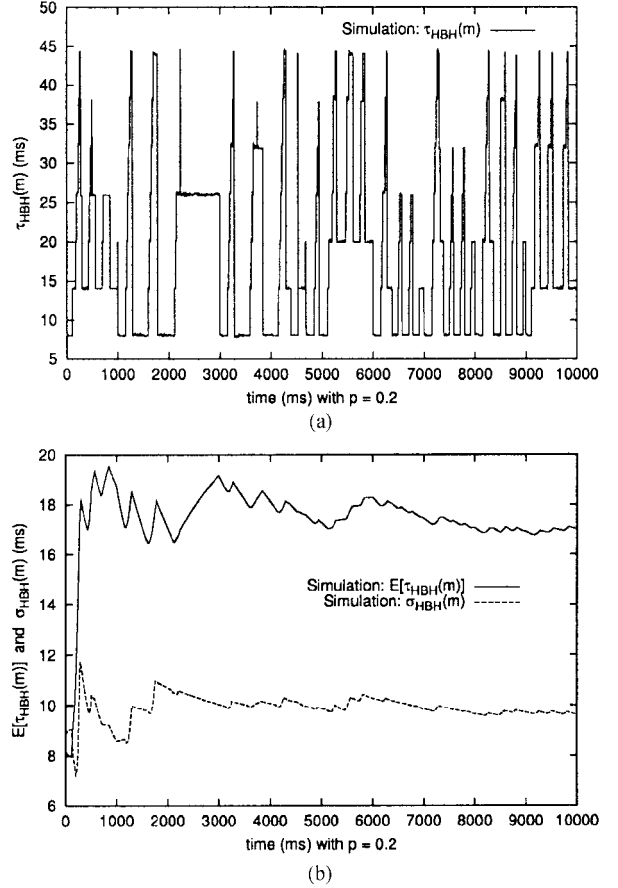


Fig. 12. Simulated multicast-tree bottleneck RM-cell RTTs and their statistics for HBH. (a) Instant multicast-tree delay $\tau_{HBH}(t)$ versus t . (b) Statistics $\bar{\tau}_{HBH}(t)$ and $\sigma_{HBH}(t)$ versus t .

enters an ON (OFF) state and stays there for a period of T_k if $R_i \leq p_i$ ($R_i > p_i$). This yields a multicast-tree bottleneck RM-cell RTT observation $\tau(t_k)$ at the end of T_k . Notice that since the 15 random cross-traffic VC_i 's independently switch between ON and OFF states at any T_k controlled by R_i , there may be multiple congested links/paths during some overlapping ON periods. Repeating the above observation in T_k 's independently for $k = 1, \dots, N$, we obtain N multicast-tree bottleneck RM-cell RTT observations $\tau(t_1), \dots, \tau(t_N)$. Then, the means $\bar{\tau}$ and standard deviations σ of the multicast-tree bottleneck RM-cell RTT can be estimated through their time-sample averages, $\hat{\tau}(T)$ and $\hat{\sigma}(T)$, respectively, over T as follows:

$$\begin{cases} \bar{\tau} \approx \hat{\tau}(T) = \frac{1}{T} \int_0^T \tau(t) dt \\ \sigma \approx \hat{\sigma}(T) = \sqrt{\frac{1}{T} \int_0^T [\tau(t) - \hat{\tau}(T)]^2 dt} \end{cases} \quad (29)$$

where $\tau(t)$ is the instant multicast bottleneck RM-cell RTT observed at time t during the simulation run.

Fig. 11(a) plots the simulated time sample of the multicast-tree bottleneck RTT of SSP for $p_i = p = 0.3$ and $N = 100$ over T . We set $T = 10000$ ms because $\hat{\tau}(T)$ and $\hat{\sigma}(T)$ already converge approximately to their statistical averages, $\bar{\tau}$ and σ , respectively, as $t \rightarrow T = 10000$ ms (as shown in Figs. 11(b) and 12(b), respectively). Fig. 11(b)

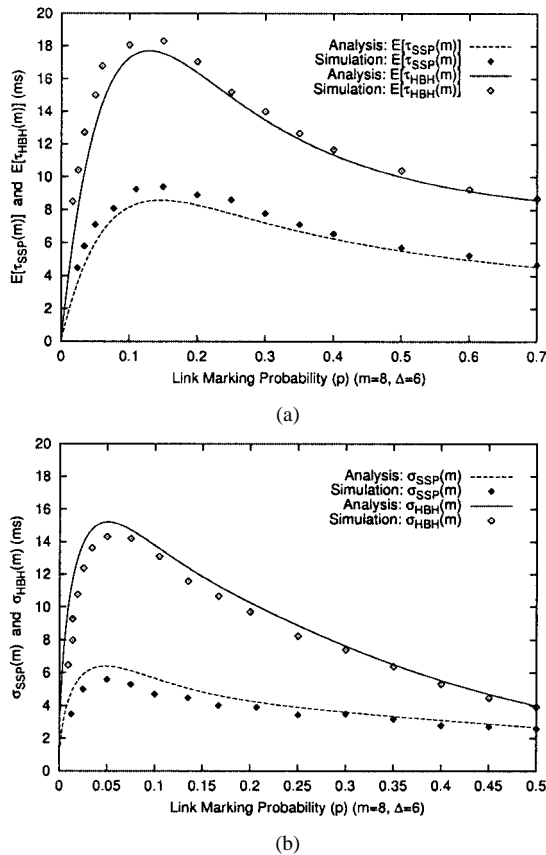


Fig. 13. Comparison of the simulated delay means and standard deviations with the analytical results. (a) $\bar{\tau}_{SSP}(m)$ and $\bar{\tau}_{HBH}(m)$ versus p . (b) $\sigma_{SSP}(m)$ and $\sigma_{HBH}(m)$ versus p .

plots the running sample-average $\hat{\tau}_{SSP}(t)$ and standard deviation $\hat{\sigma}_{SSP}(t)$ of the multicast-tree bottleneck RTT by (29). The ending values of $\hat{\tau}_{SSP}(T)$ and $\hat{\sigma}_{SSP}(T)$ over T give $\hat{\tau}_{SSP}(T) \approx \bar{\tau}_{SSP}(m)$ and $\hat{\sigma}_{SSP}(T) \approx \sigma_{SSP}(m)$, respectively, for $p = 0.3$. Fig. 11(a) shows that $\tau_{SSP}(t)$ evolves randomly, depending on the probabilities of ON-OFF cross-traffic sessions over T . However, $\tau_{SSP}(t)$ is always bounded from above by $\tau_{\max} = 2m = 16$ ms,⁶ and from below by $\tau_{\min}^{\text{Limit}}(1) = 4$ ms, confirming Theorem 2 (see also Table I). Fig. 11(b) shows that $\hat{\tau}_{SSP}(T) \rightarrow 7.78$ ms and $\hat{\sigma}_{SSP}(T) \rightarrow 3.51$ ms, approximately converging to the statistical averages $\bar{\tau}_{SSP}(m)$ and $\sigma_{SSP}(m)$ as $t \rightarrow T$ for $p = 0.3$ [see Fig. 13(a) and (b)], respectively. Likewise, we have similar observations from Fig. 12(a) and (b), which plot the corresponding simulation results with $p = 0.2$ for HBH. However, $\tau_{HBH}(t)$ is bounded from above by $\tau_{\max} = 2m = 44$ ms,⁷ and from below by $\tau_{\min}^{\text{Limit}}(1) = 8$ ms, confirming Theorem 1 (see also Table I). Fig. 12(b) shows that $\hat{\tau}_{HBH}(T) \rightarrow 17.0$ ms and $\hat{\sigma}_{HBH}(T) \rightarrow 9.7$ ms, converging approximately to the statistical averages $\bar{\tau}_{HBH}(m)$ and $\sigma_{HBH}(m)$ as $t \rightarrow T$ for $p = 0.2$ [see Fig. 13(a) and (b)], respectively.

Fig. 13(a) and (b) plot the simulated means $\bar{\tau}_{SSP}(m)$ and $\bar{\tau}_{HBH}(m)$ and standard deviations $\sigma_{SSP}(m)$ and $\sigma_{HBH}(m)$ of multicast-tree bottleneck RTTs versus p , and compare them with

⁶The slight exceed of $\tau_{SSP}(t)$ upper bound over 16 ms as shown in Fig. 11(a) is due to switching processing delays.

⁷The slight exceed of $\tau_{HBH}(t)$ upper bound over 44 ms as shown in Fig. 12(a) is due to switching processing delays.

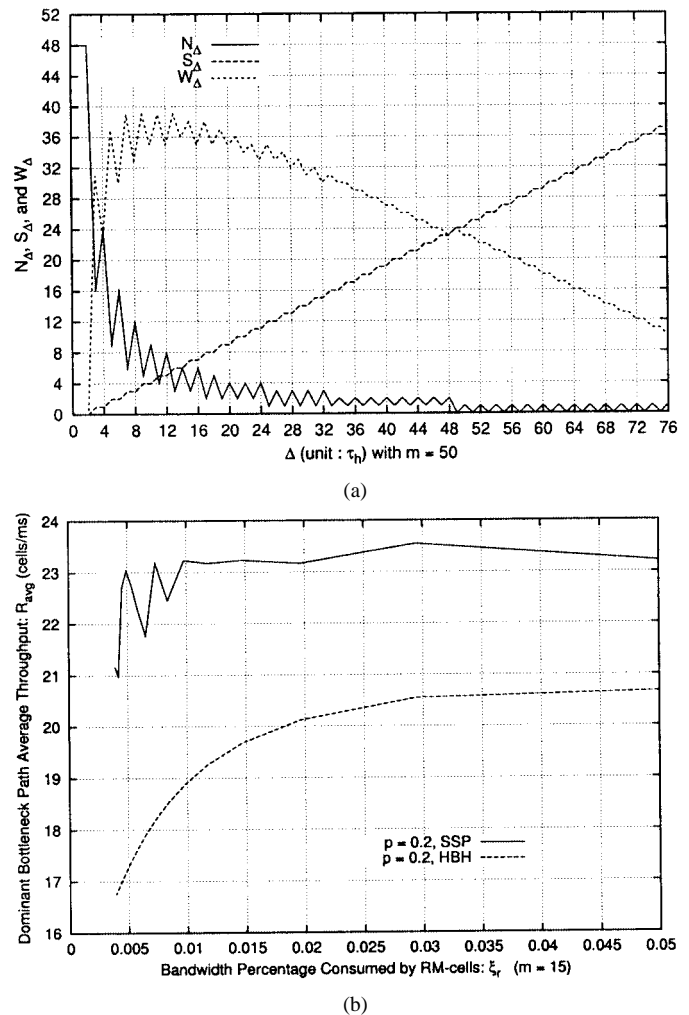


Fig. 14. Impact of Δ on multicast-tree RM-cell delays and the relation between average throughput and percentage bandwidth consumed by RM cells. (a) N_{Δ} , S_{Δ} , and W_{Δ} versus Δ with $m = 50$. (b) \bar{R} versus ξ_r ($m = 15$).

the corresponding analytical results [given by (17)–(20) in Theorem 4] derived for SSP and HBH, respectively. Comparing Fig. 13(a) and Table I, one can observe that the statistical averages of multicast-tree bottleneck RTTs for both SSP and HBH are generally smaller than the upper bound of the deterministic RTT on each path. This is because the probability $\psi(P_k, p, k)$ of the dominant bottleneck path favors the path P_{k^*} ($k^* \approx 3$) which is closer to P_1 than P_7 or P_8 . The simulation results also show the existence of the respective unique p^* maximizing $\bar{\tau}_{SSP}(m)$ and $\bar{\tau}_{HBH}(m)$, respectively, verifying that $\psi(P_k, p, k)$ has the unique maximum with respect to p for each path P_k as shown in Theorem 4.

Fig. 13(a) and (b) also shows that SSP's average multicast signaling delay over p is always smaller than the upper bound of $\tau_{\min}^{\text{Limit}}(j)$ as shown in Table I. In contrast, HBH's average multicast signaling delay can be larger than the upper bound of the physically limiting minimum $\tau_{\min}^{\text{Limit}}(j)$ for certain values of p . The statistics collected from simulations show that, on average, the signaling delay for SSP is only about one half of that for HBH as illustrated in Fig. 13(a). This advantage remains unchanged when p varies in its simulated range. Hence, the multicast flow control under SSP is more responsive, and thus more

efficient, than HBH. Moreover, Fig. 13(b) shows that the variation of multicast signaling delay for SSP is much smaller than that for HBH. So, SSP is more stable than HBH in terms of the multicast signaling delay. In addition, Fig. 13(a) and (b) also shows that the simulation results agree well with the analytical results, thus verifying the correctness of modeling and analytical results for both the deterministic analysis derived in Section III and the statistical analysis derived in Section IV.

VI. ON SELECTION OF RM-CELL UPDATE INTERVAL Δ

A. Relationships Between RM-Cell RTTs and Δ Under SSP

Unlike unicast, Δ has significant impact on multicast signaling delay. To quantify this impact, we introduce the following definition.

Definition 3: If P_j 's feedback RM cell is synchronized only with the feedback RM cells corresponding to the same forward RM cell, then P_j is said to be *strictly synchronized*. ■

P_{m-1} is always strictly synchronized since it is synchronized only with P_m . Theorem 5 describes the three *iff* conditions (as a function of Δ) to identify the strictly synchronized paths.

Theorem 5: If P_j is the j th path of an unbalanced multicast tree, which is defined in Lemma 1 ($1 \leq j \leq m-1$) and flow-controlled by SSP, then the following three claims are equivalent:

Claim 1: The number of P_j 's RM cells going through the initial state, $k_j^* = 0$, where k_j^* is defined by (5) in Theorem 2.

Claim 2: P_j is a strictly synchronized path.

Claim 3: P_j 's RM-cell RTT attains the maximum: $2m$.

Proof: The proof is available on-line in [11]. ■

Remarks on Theorem 5: The strictly synchronized path has the largest RM-cell RTT, and hence, the number of strictly synchronized paths should be minimized. As shown in (5), a larger Δ results in a larger number of strictly synchronized paths, and thus, the smaller Δ , the better.

Definition 4: Let W_j be the *net waiting time* for the P_j 's feedback RM cell to synchronize with feedback RM cells via the other paths at all consolidating branch nodes along P_j . If $W_j = 0$, then P_j is said to be *wait-free synchronized*. ■

Clearly, P_{m-1} is always wait-free synchronized according to Lemma 1. Since P_{m-1} is both strictly synchronized and wait-free synchronized, we exclude P_{m-1} from all the following theorems and treat P_{m-1} separately. Theorem 6 derives formulas for W_j and establishes the *iff* condition to identify wait-free synchronized paths, all of which are functions of Δ .

Theorem 6: If P_j is the j th path of an unbalanced multicast tree, which is defined in Lemma 1 ($1 \leq j \leq m-2$) and is flow-controlled by SSP, then the following claims hold:

Claim 1: W_j is upper-bounded by Δ , and $W_j = 2(m-j-1) - k_j^* \Delta$, where k_j^* is defined by (5) in Theorem 2.

Claim 2: If P_j is strictly synchronized, then $W_j = 2(m-j-1) > 0$.

Claim 3: P_j is a wait-free synchronized path, i.e., $W_j = 0$ iff $2(m-j-1) \bmod \Delta = 0$.

Proof: The proof is available on-line in [11]. ■

Remarks on Theorem 6:

- 1) By Lemma 2, the wait-free synchronized path has the minimum RM-cell RTT. Thus, the number of wait-free synchronized paths should be maximized.
- 2) A smaller Δ will lead to a larger number of wait-free synchronized paths. So, a small Δ is desirable.

Applying Theorems 5 and 6, we obtain the following theorem, which classifies the paths of a multicast tree into three exclusive groups, and provides explicit expressions (as functions of Δ) for the number of paths for each path group.

Theorem 7: If P_j is the j th path of an unbalanced multicast tree, which is defined in Lemma 1 ($1 \leq j \leq m-2$) and flow-controlled by SSP, then the entire path set $\mathcal{P} \triangleq \{P_1, P_2, \dots, P_{m-3}, P_{m-2}\}$ is partitioned into a strictly synchronized path subset \mathcal{P}_S , a wait-free synchronized path subset \mathcal{P}_N , and a non strictly synchronized and non-wait-free synchronized path subset \mathcal{P}_W , i.e., $\mathcal{P} = \mathcal{P}_S \oplus \mathcal{P}_N \oplus \mathcal{P}_W$, and, furthermore, for $1 \leq \Delta \leq \tau_{\max} = 2m$, the following claims hold:

Claim 1: The number of strictly synchronized paths S_Δ is determined by $S_\Delta \triangleq \|\mathcal{P}_S\| = \lceil \Delta/2 \rceil - 1$. (Here, $\|\cdot\|$ denotes the cardinality of a set.)

Claim 2: The number of wait-free synchronized paths N_Δ is determined by

$$N_\Delta \triangleq \|\mathcal{P}_N\| = \begin{cases} \lfloor 2(m-2)/\Delta \rfloor, & \text{if } \Delta = \text{even} \\ \lfloor (m-2)/\Delta \rfloor, & \text{if } \Delta = \text{odd}. \end{cases} \quad (30)$$

Claim 3: The number W_Δ of paths which are neither wait-free synchronized nor strictly synchronized is given by

$$W_\Delta \triangleq \|\mathcal{P}_W\| = \begin{cases} m - \lfloor 2(m-2)/\Delta \rfloor - \lceil \Delta/2 \rceil - 1, & \text{if } \Delta = \text{even} \\ m - \lfloor (m-2)/\Delta \rfloor - \lceil \Delta/2 \rceil - 1, & \text{if } \Delta = \text{odd}. \end{cases} \quad (31)$$

Proof: The proof is available on-line in [11]. ■

Remarks on Theorem 7:

- 1) S_Δ is proportional to Δ .
- 2) N_Δ is proportional to $1/\Delta$.
- 3) If $\Delta = 1$ or 2 , then P_j is always wait-free synchronized for all $j = 1, \dots, m-2$.
- 4) Taking $\Delta = \text{even}$ and small is preferable in increasing N_Δ .

B. Numerical Evaluations and Discussions

By Theorem 7, Fig. 14(a) plots $N_\Delta, S_\Delta, W_\Delta$ versus Δ for $m = 50$, and shows that: 1) N_Δ decreases as Δ increases, S_Δ is proportional to Δ , and W_Δ is not monotonic and reaches its maximum when $N_\Delta = S_\Delta$, and 2) when $\Delta > m-2$, N_Δ becomes very small and fluctuates between 0 and 1, while if Δ decreases from $m-2$ to 1, N_Δ increases dramatically. If τ_h is large enough, then taking $\Delta = 2$ will produce the optimal case where all paths become wait-free synchronized. Also, we observe that an even Δ is preferred since it gives a larger N_Δ than

the neighboring values of an odd Δ , which is consistent with (30). Thus, in general, Δ should be taken as an even number within $[2, m - 2]$.

While a small Δ reduces multicast signaling delay, this improvement comes at the higher bandwidth μ_r ($\propto (1/\Delta)$) cost for multicasting RM cells, introducing a tradeoff between signaling delay performance and bandwidth cost for RM cells. Using (17) and (18) and setting $m = 15$ and $p = 0.2$, Fig. 14(b) plots the average throughput \bar{R} [10] versus the percentage bandwidth $\xi_r \triangleq \mu_r/\mu_t = (\mu_t - \mu_d)/\mu_t$ consumed⁸ by the multicast signaling under SSP and HBH, respectively, with $\mu_t = 25$ cells/ms. Fig. 14(b) shows that: 1) as ξ_r is too small (near 0.5%), \bar{R} increases faster as ξ_r increases due to the more timely and accurate feedback, while as ξ_r is too large (near 5%), the increase rate of \bar{R} slows down and even gets saturated if ξ_r further increases, because the unnecessary feedback signalling starts wasting data bandwidth, and 2) SSP outperforms HBH, in terms of \bar{R} 's for any given ξ_r and the robustness to the variation of Δ , verifying Theorems 1, 2, and 4.

VII. CONCLUSION

We developed both deterministic and stochastic binary-tree models to study the delay performance of the multicast signaling algorithms. Using the deterministic model, we derived each path's feedback RTT in a multicast tree. By applying the statistical model we obtained the probabilities of each path becoming the multicast-tree bottleneck. We employed these two models to analyze and contrast the signaling delay scalability of two multicast signaling protocols, SSP and HBH, by deriving their first and second moments of multicast signaling delays. Also derived is the optimal flow-control update interval for SSP to minimize multicast signaling delay. SSP is shown, deterministically and statistically, to scale better than HBH in signaling delay, and make the effective multicast signaling delay virtually independent of, and hence, scalable with, the multicast-tree topology.

ACKNOWLEDGMENT

The authors would like to thank S. H. Low for his preprints.

REFERENCES

- [1] J. Crowcroft and K. Paliwoda, "A multicast transport protocol," in *Proc. ACM SIGCOMM*, Aug. 1988, pp. 247–256.
- [2] L. Roberts, "Rate based algorithm for point to multipoint ABR service," ATM Forum, Contribution 94-0772, Sept. 1994.
- [3] K.-Y. Siu and H.-Y. Tzeng, "Congestion control for multicast service in ATM networks," in *Proc. IEEE GLOBECOM*, Nov. 1995, pp. 310–314.
- [4] —, "On max-min fair congestion control for multicast ABR services in ATM," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 545–556, Apr. 1997.
- [5] H. Saito, K. Kawashima, H. Kitazume, A. Koike, M. Ishizuka, and A. Abe, "Performance issues in public ABR service," *IEEE Commun. Mag.*, vol. 11, pp. 40–48, Nov. 1996.

⁸ μ_t , μ_d , and μ_r are the total available bandwidth, bandwidth used by data cells, and bandwidth used by RM cells, respectively, at the dominant bottleneck link/path.

- [6] Y.-Z. Cho and M.-Y. Lee, "An efficient rate-based algorithm for point-to-multipoint ABR service," in *Proc. IEEE GLOBECOM*, Nov. 1997, pp. 790–795.
- [7] W. Ren, K.-Y. Siu, and H. Suzuki, "On the performance of congestion control algorithms for multicast ABR in ATM," in *Proc. IEEE ATM Workshop*, 1996.
- [8] S. Fahmy, R. Jain, R. Goyal, B. Vandalor, and S. Kalyanaraman, "Feedback consolidation algorithms for ABR point-to-multipoint connections in ATM networks," in *Proc. IEEE INFOCOM*, Apr. 1998, pp. 1004–1013.
- [9] X. Zhang and K. G. Shin, "Performance Analysis of feedback synchronization for multicast ABR flow control," in *Proc. IEEE GLOBECOM*, Dec. 1999, pp. 1269–1274.
- [10] X. Zhang, K. G. Shin, D. Saha, and D. Kandlur, "Scalable flow control for multicast ABR services," *IEEE/ACM Trans. Networking*, vol. 10, pp. 67–85, Feb. 2002.
- [11] X. Zhang and K. G. Shin, "Delay analysis of feedback-synchronization signaling for multicast flow control," Networking and Information Systems Lab., Dept. Elect. Eng., Texas A&M Univ., College Station, Tech. Rep. [Online.] Available: http://ece.tamu.edu/~xizhang/papers/mcast_sig.pdf, July 2002.
- [12] —, "Second-order rate-control based transport protocols," in *Proc. IEEE Int. Conf. Network Protocols (ICNP)*, Nov. 2001, pp. 342–350.
- [13] D. Lapsley and S. Low, "Random early marking: For internet congestion control," in *Proc. IEEE GLOBECOM*, Dec. 1999, pp. 1747–1752.
- [14] S. Athuraliya, D. Lapsley, and S. Low, "An enhanced random early marking algorithm for internet flow control," in *Proc. IEEE INFOCOM*, Mar. 2000, pp. 1425–1434.
- [15] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. Networking*, vol. 1, pp. 397–413, Aug. 1993.
- [16] S. Floyd, "TCP and explicit congestion notification," in *ACM SIGCOMM Comput. Commun. Rev.*, vol. 24, Oct. 1994, pp. 10–23.
- [17] X. Zhang and K. G. Shin, "Statistical analysis of feedback synchronization signaling delay for multicast flow control," in *Proc. IEEE INFOCOM*, Apr. 2001, pp. 1152–1161.
- [18] A. Heybey, "The Network Simulator," Labs. Comput. Sci., Massachusetts Inst. Technol., Cambridge, MA, Oct. 1990.
- [19] A. Tanenbaum, *Computer Networks*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1996.



Xi Zhang (S'89–SM'98) received the B.S. and M.S. degrees from Xidian University, Xi'an, China, and the M.S. degree from Lehigh University, Bethlehem, PA, all in electrical engineering and computer science. He received the Ph.D. degree in electrical engineering and computer science (Electrical Engineering–Systems) from The University of Michigan, Ann Arbor.

He is currently an Assistant Professor and the Founding Director of the Networking and Information Systems Laboratory, Department of Electrical Engineering, Texas A&M University, College Station. He was an Assistant Professor and the Founding Director of the Division of Computer Systems Engineering, Department of Electrical Engineering and Computer Science, Beijing Information Technology Engineering Institute, Beijing, China, from 1984 to 1989. He was a Research Fellow with the School of Electrical Engineering, University of Technology, Sydney, Australia, and the Department of Electrical and Computer Engineering, James Cook University, Queensland, Australia, under a fellowship from the Chinese National Commission of Education. He was a Summer Intern with the Networks and Distributed Systems Research Department, Bell Labs, Murray Hills, NJ, and at AT&T Labs Research, Florham Park, NJ, in 1997. His current research interests focus on the areas of design, modeling, and performance analysis of protocols and systems for multicast and unicast computer communications over wired and wireless networks supporting QoS guarantees, information theory, statistical communications theory, random signal processing, and distributed computer-control systems. He has published approximately 40 technical papers in the above areas.

Dr. Zhang is a Member of the Association for Computing Machinery (ACM).



Kang G. Shin (S'75–M'78–SM'83–F'92) received the B.S. degree in electronics engineering from Seoul National University, Seoul, Korea, in 1970 and the M.S. and Ph.D degrees in electrical engineering from Cornell University, Ithaca, NY, in 1976 and 1978, respectively.

He is the Kevin and Nancy O'Connor Chair Professor of Computer Science and the Founding Director of the Real-Time Computing Laboratory, Department of Electrical Engineering and Computer Science, The University of Michigan, Ann Arbor.

From 1978 to 1982, he was on the faculty of Rensselaer Polytechnic Institute, Troy NY. He has held visiting positions at the U.S. Air Force Flight Dynamics Laboratory, AT&T Bell Laboratories, the Computer Science Division, Department of Electrical Engineering and Computer Science, University of California at Berkeley, International Computer Science Institute, Berkeley, CA, IBM T. J. Watson Research Center, and Software Engineering Institute, Carnegie-Mellon University. He also chaired the Computer Science and Engineering Division, Electrical Engineering and Computer Science Department, The University of Michigan, for three years beginning in January 1991. His current research interests focus on the areas of QoS-sensitive networking and computations, distributed real-time fault-tolerant computing, and embedded systems. He has supervised the completion of 42 Ph.D. dissertations/theses, and authored/coauthored over 600 technical papers and numerous book chapters in the areas of distributed real-time computing and control, computer networking, fault-tolerant computing, and intelligent manufacturing. He has co-authored (jointly with C. M. Krishna) a textbook, *Real-Time Systems* (New York: McGraw Hill, 1997).

Dr. Shin has received numerous prestigious awards, including the 2003 IEEE Communications Society William R. Bennet Prize Award for the best original paper published in the IEEE/ACM TRANSACTIONS ON NETWORKING, the outstanding paper awards from the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, the IEEE IWQoS 2003, the 2000 UNSENIX Technical Conference, and the Best Student Paper Awards from the 1996 IEEE Real-Time Technology and Application Symposium. He was the General Chair of the 2000 IEEE Real-Time Technology and Applications Symposium, the Program Chair of the 1986 IEEE Real-Time Systems Symposium (RTSS), the General Chair of the 1987 RTSS, the Guest Editor of the 1987 August special issue of the IEEE TRANSACTIONS ON COMPUTERS on Real-Time Systems, a Program Co-Chair for the 1992 International Conference on Parallel Processing, and has served on numerous technical program committees. He also chaired the IEEE Technical Committee on Real-Time Systems from 1991 to 1993, was a Distinguished Visitor of the Computer Society of the IEEE, an Editor of the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED COMPUTING, and is currently an Editor of the *ACM Transactions on Embedded Computing Systems* and an Area Editor of the *International Journal of Time-Critical Computing Systems* and *Computer Networks*. He is a Fellow of the IEEE and of the Association for Computing Machinery (ACM).