A New Method for Integrated End-to-End Delay Analysis in ATM Networks

Joseph Kee-Yin Ng, Shibin Song, Chengzhi Li, and Wei Zhao

Abstract: **For admitting a hard real-time connection to an ATM network, it is required that the end-to-end delays of cells belonging to the connection meet their deadlines without violating the guarantees already provided to the currently active connections. There are two kinds of methods to analyze the end-to-end delay in an ATM network. A decomposed method analyzes the worst case delay for each switch and then computes the total delay as the sum of the delays at individual switches. On the other hand, an integrated method analyzes all the switches involved in an integrated manner and derives the total delay directly. In this paper, we present an efficient and effective integrated method to compute the end-to-end delay. We evaluate the network performance under different system parameters and we compare the performance of the proposed method with the conventional decomposed and other integrated methods [1], [3], [5]–[9].**

Index Terms: **Real-time communication, ATM networks, end-to-end delay analysis, performance evaluation.**

I. INTRODUCTION

ATM network is becoming increasingly popular in supporting hard real-time applications. As a connection-oriented packet-switched technology, the ATM network requires that before two hosts start to communicate, a connection must be established. In the case of real-time applications, every connection has to meet its delay requirement. Thus, we must have an *efficient* and *effective* method to derive the worst case end-to-end delays of cells belonging to the connections in the ATM networks.

For the delay analysis method to be effective, it must be able to produce delay bounds that are relatively tight. Excessively overestimating the delays may make the system admit less connections than it could, resulting in under-utilization of resources. On the other hand, a delay analysis method should also be *efficient*, i.e., it should be simple and fast in order to be used in on-line connection admission.

Manuscript received August 1, 1998; approved for publication by Bong Dae Choi, Division III Editor, June 25, 1999.

Joseph Ng is with the Department of Computer Science, Hong Kong Baptist University, Hong Kong, e-mail: jng@comp.hkbu.edu.hk.

Shibin Song is currently a visiting research scholar at the Department of Computer Science, Hong Kong Baptist University, Hong Kong. He is with the Department of Mathematics, Zhongshan University, Guangzhou, China, e-mail: ssong@comp.hkbu.edu.hk.

Chengzhi Li and Wei Zhao are with the Department of Computer Science, Texas A&M University, U.S.A., e-mail: chengzhi@cs.tamu.edu and zhao@cs.tamu.edu.

The work reported in this paper was supported in part by the Hong Kong RGC Earmarked Research Grant under RGC/97-98/54 and by the FRG of HKBU under FRG/96-97/II-103.

There are two different approaches to derive the end-to-end delay for an ATM network. A *decomposed* approach analyzes the worst case cell delay for each ATM switch involved and computes the worst case end-to-end delay as a summation of the worst case delays at individual switches. An advantage of the decomposed method is its simplicity. However, it is usually pessimistic because a cell suffers the worst case delay at one switch possibly may not suffer the worst case delay at the next and subsequence switches.

The second approach is to analyze the involved ATM switches in an *integrated* manner and derive the end-to-end delay directly. This kind of integrated methods can produce a tighter end-to-end delay bounds than the decomposed method and thus, improving the system effectiveness. However, such an integrated delay analysis may be very complicated and hence may not be suitable to be used in on-line connection admission control.

Several studies have been reported to address this problem. By introducing the concept of guarantee service curves and the operation of min-plus convolution, an end-to-end service curve for a connection can be obtained. The service curve provides a lower bound on the service a connection may receive in the given time interval. Then, the end-to-end worst case delay can be derived as in [1], [3], [9]. As we will show in the later part of this paper, this integrated method can be efficient as well if proper traffic model is utilized. However, the service curve based approach tends to underestimate the service a connection may receive. As a result, it may still overestimate the end-to-end delay. In this paper, we propose a new integrated method to deal with this problem by taking a *system equivalency* approach. In this approach, for a given ATM network with multiple switches, we will manage to derive a single switch system that has the exact same input and output behavior as the multiple switch one in terms of cell timing delay characteristics. We then compute the end-to-end delay based on this simplified system. While as efficient as its predecessors, our new method produces much tighter delay bounds than the others, hence improving the system effectiveness dramatically. Extensive simulations are performed to verify the improvement in the system performance.

Our work [22]–[24] complements the recent progress in real-time communications [2], [12]–[15], [18], [21], [29], [30]. Much of the previous work in ATM has been concentrating on obtaining the delay bounds and connection admission criteria for individual scheduling [4]–[9], [16], [19]–[20], [25]–[28], [33], [34]. Ferrari and Verma [10] and Zheng and Shin [39] studied the use of Earliest Deadline First scheduling in wide area networks. Zhang and Ferrari [36] discussed how local deterministic delay bounds can be guaranteed over an ATM link for

1229-2370/99/$10.00

bursty traffic, even when the sum of peak rates of all the connections exceeds one. Deterministic delay bounds in networks have also been studied by Yates, Kurose and Towsley [35] and by Cruz [5]–[9].

Analyzing delay in an integrated manner has been addressed in [9], [11]. While Cruz provided a general framework for integrated analysis [9], Georgiadis, Guerin, Peris and Sivarajan concentrated on networks consisting of rate-controlled service [11]. Furthermore, several research groups have also been investigating the admission control based on the delay analysis utilizing the service curves [1], [3], [31].

The remainder of this paper is organized as follows. In Section II, we give a brief review for the hard real-time connections, present our ATM network model and introduce the traffic description functions. Section III discusses the two integrated methods in analyzing the worst case end-to-end delay. In Section IV, the performance comparison among the decomposed method and the two integrated methods on the worst case end-to-end delay are presented. Section V concludes the paper.

II. SYSTEM MODELS AND DEFINITIONS

A. Networks

An ATM switch consists of input ports, switching fabric and output ports. A cell that arrives at an input port of a switch is transported by the switching fabric to an output port where the cell is then transmitted along the physical link associated with the output port. Refer to Fig. 1 for a general architecture of an ATM switch.

The scheduling policy at an output port controller of an ATM switch determines the order of the cells (from different connections) to be transmitted. In this paper, we assume that a priority driven scheduler is utilized at the output ports of the ATM switches. Priority driven schedulers are popular for real-time application commercial products. Many commercial vendors have implemented them in their products. For example, the ForeRunner switch from the Fore Co. adopts a priority driven scheduler.

In ATM networks, messages are segmented into fixed size packets called *cells*. Let τ be the transmission time of an ATM cell. In this paper, we assume that time is normalized by *tau* as one time unit. This simplifies our analysis without any loss of generality.

B. Connections

ATM is a connection-oriented technology. Before two hosts begin to communicate, a connection has to be set up between them. The term *connection* is often used to denote the stream of messages send from a source host to a destination host. In this paper, we deal with *hard real-time* connections. A hard real-time connection has a stringent deadline constraint on the delay of its cells.

Let D_i be the deadline of connection M and $M = \{M_1, M_2, ..., M_m\}$ denote the connection set. Without loss of generality, we assume that the priorities are assigned in the same order as the connection indices, i.e., $P_i = i$, for $i = 1, 2, ..., m$. M_1 has the highest priority and M_m has the lowest priority. Specific priority assignment methods will be discussed in Section IV.

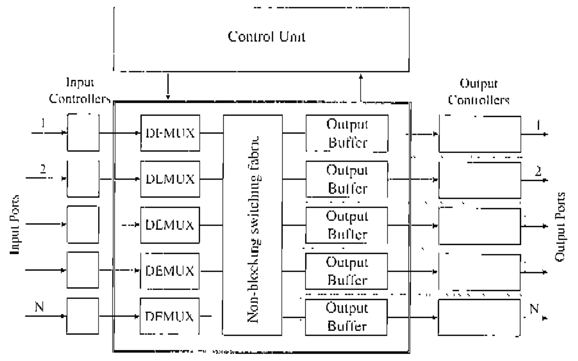


Fig. 1. A general ATM switch architecture.

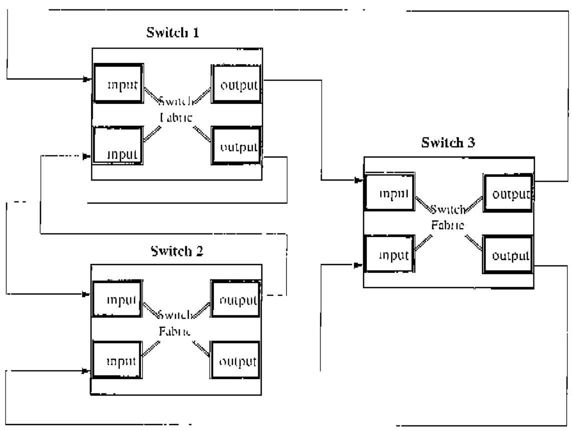


Fig. 2. An ATM LAN with 3 switches.

Fig. 2 shows an ATM network with three switches. This sample network will be used to illustrate various concepts introduced in this paper.

C. Servers

To simplify the analysis, a concept of a *server* was introduced in [5]. A *server* is an abstraction of a network component that is traversed by the cells of a connection. The input ports, the switching fabric, the output ports and the physical links can be modeled as servers serving the ATM connections. We can therefore model an ATM network as a collection of servers.

Servers are classified into two categories: constant servers and variable servers [5], [28]. A *constant* server offers a constant delay to each cell that uses it but does not by itself change the traffic flow characteristics of a connection. For example, the physical links and switching fabric are constant servers. Furthermore, the function of an input port is to de-multiplex the arriving cells based on the information in the cell header. This can be achieved in constant time by the hardware associated with the input port. Thus, we can also model the input port of an ATM switch as a constant server.

On the other hand, the functionality of an output port of a switch is more complex. An output port may simultaneously receive cells belonging to different connections competing for the

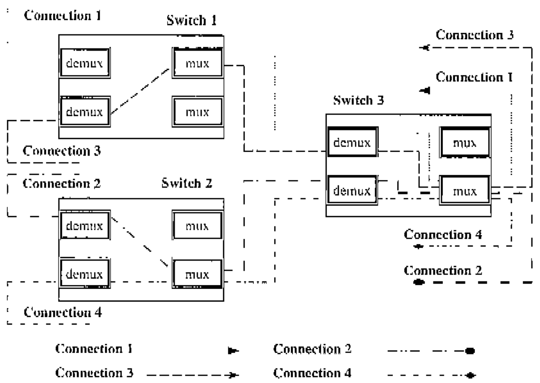


Fig. 3. Example of decomposing a network into server.

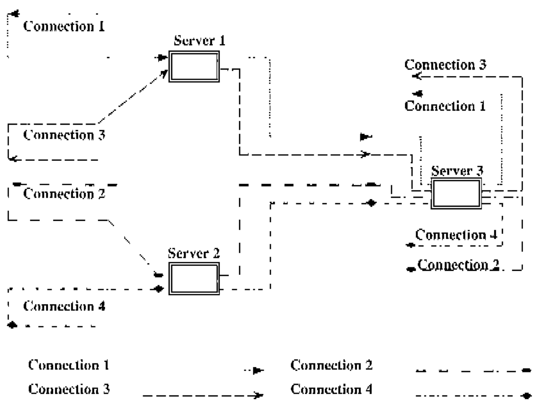


Fig. 4. Connection-server graph.

transmission on the link associated with the output port. The cells may be transmitted in an order determined by the scheduling policy adopted by the switch. A multiplexor server must therefore be considered as a *variable* server since the delay suffered by a cell in this server depends on the queue length in the buffer. Consequently, the traffic characteristics of a connection at the output of a variable server may differ from that at the input of the server.

D. Connection-Server Graph

A connection-server graph is constructed as a labeled, directed graph with the servers as its node. A directed edge is introduced from server A to server B if there is a connection that is served by server A followed by server B. The edge is labeled by the connection that uses it. Fig. 3 presents the connection-server graph for the network shown in Fig. 2. Note that there are four connections in the system. They travel through Switch 1 or Switch 2 and exit at Switch 3.

Recall that the constant servers serving a connection only add a fixed amount of delay to its cells and do not change its traffic characteristics. Hence, if we subtract the appropriate constant delays encountered at these constant servers from the deadline of the connection, the impact of constant servers can be eliminated. For the rest of the paper, we assume that the delay requirements of connections are modified as described above. Consequently, we eliminate all the constant servers from further consideration and focus only on the variable servers in the remainder of the paper. Thus, we can view a connection as being served by a sequence of variable servers only. Fig. 4 shows the connection-server graph after eliminating the constant servers for the system shown in Fig. 3. For the rest of the paper, we will often omit the prefix 'variable' when referring to variable servers to avoid repetitiousness.

The end-to-end delays of the connections depend on how the severs are connected. Thus, the network topology plays an important role. It is easy to recognize that a connection will suffer the worst case end-to-end delay when the network is tree-structured, i.e., when connections from different hosts eventually merge into one single server. Many studies have assumed this kind of topology [17], [27]. Hence, in this paper we will also focus on this kind of topology. We will discuss how to extend our methods to other topologies in Section V.

Fig. 4 shows the connection-server graph for a 2-layer multi-server tree network system. The first layer consists of Switch 1 and Switch 2. The second layer consists of Switch 3. In general, we consider N-layer multi-server systems in this paper.

E. Traffic Descriptions

The main challenge of analyzing the worst case end-to-end delay in a network lies in the difficulty of accurately describing the connection's traffic inside the network. Many traffic models have been presented to characterize the network traffic. Refer to [37], [38] for detailed information.

Definition 1: The *arrival function* $R_{i,k}(t)$ of connection M_i on the k-th layer server is defined as the amount of data coming from the connection M_i to the k-th layer server during time interval [0,t).

Note that the output traffic of the connection M_i at the k-th layer server is the input traffic of the connection M_i at the $(k+1)$-th layer server

Definition 2: A function $F_{i,k}(I)$ is called the arrival curve for connection M_i at the k-th layer server if for any $t \geq 0, I \geq 0$,

$$R_{i,k}(t+I) - R_{i,k}(t) \leq F_{i,k}(I). \tag{1}$$

In this paper, we assume that the initial arrival curves $F_{i,1}(I),\ i = 1, ..., m$, belongs to the (β, ρ) model proposed by Cruz in [5], i.e.,

$$F_{i,1}(I) = \begin{cases} I, & 0 \leq I \leq I_i, \\ \beta_i + I * \rho_i, & I_i \leq I, \end{cases} \tag{2}$$

where

$$I_i = \frac{\beta_i}{1-\rho_i}. \tag{3}$$

III. INTEGRATED DELAY ANALYSIS

In this section, we derive the end-to-end worst case delays using integrated analysis methods. The first method is based on the guaranteed service curve [1], [3], [9] and the second method is based on system equivalency.

A. Notations

Let us define some notations that will be used for the rest of this section.

Definition 3: Let $d_{i,k}$ be the worst case delay experienced by connection M_i at the k-th layer server and $d_i^{End-to-end}$ be the worst case end-to-end delay experienced by connection M_i.

Definition 4: $G_{i,k} = \{j : j < i$, connection M_j enters the same server at the k-th layer with $M_i\}$.

That is, $G_{i,k}$ is the set of indices of the connections that enter the same server at the k-th layer with connection M_i and have higher priorities than connection M_i. For example, for the network shown in Fig. 4, $G_{3,2} = \{1, 2\}$. It is easy to show that $G_{i,k} \subseteq G_{i,k+1}$ for $k = 1, 2, \ldots, N$.

Definition 5: A time interval $[t_1, t_2)$ is a *busy period* of connection M_i at the k-th layer server if during this interval,

(1) the k-th layer server only serves the traffic coming from connection M_i or other connections with higher priority than connection M_i, and
(2) the capability of the k-th layer server is fully utilized.

Definition 6: A time interval $[t_1, t_2)$ is a maximum busy period of connection M_i at the k-th layer server if $[t_1, t_2)$ is the longest busy period of M_i that contains t_1.

It is obvious that during the life-time of a connection, there are possibly many maximum busy periods of connection M_i at the k-th layer server. However, by definition these maximum busy periods do not intersect each other.

B. Integrated Delay Analysis by Guaranteed Service Curve

In this subsection, we will derive an upper bound for the end-to-end delay based on the guaranteed service curve. This method was the first integrated method to be proposed [9]. Various issues related to this method have been discussed in [1], [3], [31]–[32]. In this paper, we derive a closed-form formula of the delay bound. Using the closed-form formula, performances of different methods can be compared analytically. Closed-form formula also simplifies the connection admission control procedure in practice.

We first formally introduce the definition of guaranteed service curves [1], [3], [9], [32], which characterizes the service received by a connection at a particular server. Hence, it can help to derive the worst case delay at the server.

Definition 7: Function $C_{i,k}(t)$ is called a *service curve guaranteed* by the k-th layer server to connection M_i if for any t, there exists s, $0 \leq s \leq t$, such that

$$\begin{aligned} &R_{i,k+1}(s) - R_{i,k}(s) = 0, \\ &R_{i,k+1}(t) - R_{i,k}(s) \geq C_{i,k}(t-s). \end{aligned} \tag{4}$$

The main idea of an integrated method is to treat the entire network as one server. From this point of view, the guaranteed network service curve is defined as follows.

Definition 8: Function $C_i^{End-to-end}(t)$ is called a *guaranteed network service curve* to connection M_i, if for all t, there exists s, $0 \leq s \leq t$, such that

$$\begin{aligned} &R_{i,N+1}(s) - R_{i,1}(s) = 0, \\ &R_{i,N+1}(t) - R_{i,1}(s) \geq C_i^{End-to-end}(t-s). \end{aligned} \tag{5}$$

To derive a closed-form formula for the delay bound, we need a closed-form formula for the guaranteed service curves. For the systems we concern in this paper, we have the following results.

Lemma 1: A service curve $C_{i,k}(I)$ guaranteed by the k-th layer server to connection M_i is given as follows:

$$C_{i,k}(I) = \begin{cases} 0, & 0 \leq I \leq \theta_{i,k}, \\ (1-\pi_{i,k})(I-\theta_{i,k}), & I > \theta_{i,k}, \end{cases} \tag{6}$$

where $\pi_{i,k}$ and $\theta_{i,k}$ are defined as follows:

$$\pi_{i,k} = \sum_{j \in G_{i,k}} \rho_j, \tag{7}$$

and

$$\theta_{i,k} = \frac{\sum_{j \in G_{i,k}} \beta_j}{1-\pi_{i,k}}. \tag{8}$$

Proof: For the proof of this lemma, see Appendix A, also available in [24]. □

Lemma 2: A network service curve guaranteed to connection M_i is given as follows:

$$C_i^{End-to-end}(I) = \begin{cases} 0, & 0 \leq I \leq \sum_{k=1}^{N} \theta_{i,k}, \\ (1-\pi_{i,N})\left(I - \sum_{k=1}^{N} \theta_{i,k}\right), & I > \sum_{k=1}^{N} \theta_{i,k}, \end{cases} \tag{9}$$

where $\pi_{i,N}$ and $\theta_{i,k}$ are defined in (7) and (8), respectively.

Sketch of the Proof: According to [9], if the k-th layer server in the route of connection M_i guarantees a service curve $C_{i,k}(I)$ to connection M_i, $k = 1, 2, \ldots, N$, the network guarantees a network service curve of $C_i^{End-to-end}(I)$ to connection M_i, where

$$C_i^{End-to-end}(I) = \min\left\{\sum_{k=1}^{N} C_{i,k}(x_k) \middle| x_k \geq 0, \sum_{k=1}^{N} = I\right\}. \tag{10}$$

Substituting (6) into (10) with certain algebraic manipulations, we can then derive (9). For the details of the proof, see Appendix A or [24].

Based on Lemma 2, we are now ready to derive the closed-form formula for the end-to-end delay bound. The result is shown in Theorem 1.

Theorem 1: With the integrated method by guaranteed service curve, the end-to-end delay $d_i^{End-to-end}$ experienced by connection M_i is bounded by

$$d_i^{End-to-end} \leq \sum_{k=1}^{N} \theta_{i,k} + I_i \frac{\pi_{i,N}}{1-\pi_{i,N}}, \tag{11}$$

where $I_i, \pi_{i,N}$, and $\theta_{i,k}$ are defined in (3), (7) and (8), respectively.

Sketch of the Proof: According to Theorem 2.7 in [32], the worst case end-to-end delay bound satisfies the following inequality

$$d_i^{End-to-end} \le \max_{c\ge 0}\left\{C_i^{-1^{End-to-end}}(c) - F_{i,1}^{-1}(c)\right\}, \quad (12)$$

Substituting (2) and (9) into (12), we can derive (11). For the details of the proof, see Appendix A or [24].

C. Integrated Delay Analysis by System Equivalency

In this section, we derive end-to-end delay bound based on equivalent systems.

Definition 9: Two network systems are said to be equivalent if whenever they have the same input traffic, they have the same output traffic.

In this section, we first show that the multi-server system with a tree topology and priority driven scheduling algorithm is equivalent to a single server system with the same priority assignment.

For connection M_i in a multi-server system, we define $R_i^{IN,M}(t)$ and $R_i^{OUT,M}(t)$ to be the arrival traffic function at the entrance of the network and the output traffic function at the exit of the network, respectively. Thus, $R_{i,1}(t) = R_i^{IN,M}(t)$ and $R_{i,N+1}(t) = R_i^{OUT,M}$. Let $R_i^{IN,S}(t)$ and $R_i^{OUT,S}(t)$ be the corresponding functions for a single server system as show in Fig. 5.

To prove the equivalence between a multi-server and a single server system, we need to establish the relationship between their input and output traffic. The following lemmas provide the required results.

Lemma 3: In the single server priority driven system, the output traffic function of connection M_i is given as

$$\begin{aligned} R_i^{OUT,S}&(t) \\ &= \inf_{s\le t}\left\{\left[(t-s) - \sum_{j=1}^{i-1}\left(R_j^{OUT,S}(t) - R_j^{OUT,S}(s)\right)\right]^+ \right.\\ &\quad \left. + R_i^{IN,S}(s)\right\} \\ &= \left[(t-s') - \sum_{j=1}^{i-1}\left(R_j^{OUT,S}(t) - R_j^{OUT,S}(s')\right)\right]^+ \\ &\quad + R_i^{IN,S}(s'), \end{aligned} \quad (13)$$

where $[x]^+ = \max\{0, x\}$ and s' is the starting point of the maximum busy period of connection M_i which contains time t. If time t is not in any connection M_i's maximum busy period, $s' = t$.

The proof of this lemma can be found in [1]. Here we explain its physical meaning. Because s' is the starting point of the maximum busy interval, all arrival traffic of connection M_i at time s' is transmitted. The amount of this part of output traffic at time s' is $R_i^{IN,S}(s')$. Starting from s', the server

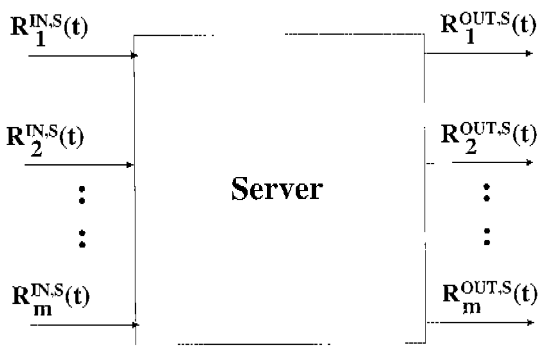


Fig. 5. A single server system.

will serve connections with the highest priority first. The left-over time is given to serve the connection M_i. No connection with priority lower than i is served. Hence, the amount of traffic of the connection M_i served by the server is given by $\left[(t-s') - \sum_{j=1}^{i-1}\left(R_j^{OUT,S}(t) - R_j^{OUT,S}(s')\right)\right]^+$. This explains why

$$R_i^{OUT,S}(t) = \left[(t-s') - \sum_{j=1}^{i-1}\left(R_j^{OUT,S}(t) - R_j^{OUT,S}(s')\right)\right]^+ + R_i^{IN,S}(s').$$

For a multi-server system, the relationship between its input and output traffic is given by the following lemma.

Lemma 4: In a multi-server system, the output traffic function of connection M_i is given as follows:

$$\begin{aligned} R_i^{OUT,M}&(t) \\ &= \inf_{s\le t}\left\{\left[(t-s) - \sum_{j=1}^{i-1}\left(R_j^{OUT,M}(t) - R_j^{OUT,M}(s)\right)\right]^+ \right.\\ &\quad \left. + R_i^{N-1}(s)\right\} \\ &= \left[(t-s'') - \sum_{j=1}^{i-1}\left(R_j^{OUT,M}(t) - R_j^{OUT,M}(s'')\right)\right]^+ \\ &\quad + R_i^{IN,M}(s''), \end{aligned} \quad (14)$$

where s'' is the starting point of the maximum busy period of connection M_i, which contains t, at the N-th layer server. If time t is not in any maximum busy period of connection M_i at the N-th layer server, then $s'' = t$.

Proof: For the proof of this lemma, see Appendix A or [24]. Now we can prove our main result. □

Theorem 2: A multi-server priority driven tree system is equivalent to a single server priority driven system if the priorities of connections are assigned in the same way, i.e., if

Table 1. Delay formulae for the worst case end-to-end delay in an ATM network.

Methods	Delay Formula
Decomposed Method	$d_i^{End-to-end} = \sum_{k=1}^{N}\left(\theta'_{i,k} + I_{i,k}\frac{\pi_{i,k}}{1-\pi_{i,k}}\right)$
Integrated Method (Guaranteed Service Curve)	$d_i^{End-to-end} = \sum_{k=1}^{N}\theta_{i,k} + I_i\frac{\pi_{i,N}}{1-\pi_{i,N}}$
Integrated Method (System Equivalency)	$d_i^{End-to-end} = \theta_{i,N} + I_i\frac{\pi_{i,N}}{1-\pi_{i,N}}$

$R_i^{IN,S}(t) = R_i^{IN,M}(t)$, then $R_i^{OUT,S}(t) = R_i^{OUT,M}(t)$, $i = 1, 2, \ldots, m$.

Proof: For the proof of this Theorem, see Appendix A or [24]. This equivalency result greatly simplifies the computation of end-to-end delay bounds. In particular, we have the following closed-form formula for the end-to-end delay bound. □

Theorem 3: The end-to-end delay, $d_i^{End-to-end}$, experienced by connection M_i in the multi-server priority driven tree system is bounded by

$$d_i^{End-to-end} \leq \theta_{i,N} + I_i\frac{\pi_{i,N}}{1-\pi_{i,N}}, \quad (15)$$

where $I_i, \pi_{i,N}$, and $\theta_{I,k}$ are defined in (3), (7) and (8), respectively.

Proof: From Theorem 2, the tree system is equivalent to a single server priority driven system. Therefore, the end-to-end delay for the tree system is the same as the delay in the equivalent single server system. Therefore, we can find the end-to-end delay bound for the tree system as if it was a single server system by using Theorem 1 directly. □

IV. PERFORMANCE COMPARISON

In this section, we first compare the closed-form formulae for the worst case end-to-end delay bounds derived by various methods. We then introduce the performance metric and evaluate network performance with different delay derivation methods.

A. Comparison of Delay Formulae

Table 1 shows the delay formulae for the worst case end-to-end delay bounds derived by the decomposed method, the integrated method by guaranteed service curves and our new integrated method by system equivalency. The derivation of delay formulae with the decomposed method is given in Appendix B or in [24]. Comparing between the decomposed method and the guaranteed service curve (GSC) method, since $\theta'_{i,k} \geq \theta_{i,k}$ and the second term in both formulae differs by $\sum_{k=1}^{N-1} I_{i.k}\frac{\pi_{i,N}}{1-\pi_{i,N}}$, it is seen that the GSC method produces a smaller delay bound than the decomposed method. Similarly, comparing the two integrated methods, we can clearly see that the integrated method by System Equivalency (SEQ) produces an even smaller delay bound. In particular, the delay bound derived by our SEQ method will always be $\sum_{k=1}^{N-1}\theta_{i,k}$ better than that by the GSC method. In short, our newly proposed integrated method is the best among the three methods that have been proposed and analyzed so far.

B. Performance Metric

We wish to evaluate some statistical performance of network systems with different delay derivation methods. Performance metric we are interested in is *admission probability*, the probability that a hard real-time connection can be admitted to the ATM network. Recall that a hard real-time connection is admitted if and only if its worst case end-to-end delay can meet its deadline constraint without violating the deadline constraints of connections that are already in the system. Obviously, the method to derive the worst case end-to-end delay will impact the admission probability. For the simulations, the admission probability P_{adm} is defined as follows:

$$P_{adm} = \frac{N_{accept}}{N_{accept} + N_{reject}}, \quad (16)$$

where N_{accept} is the number of connections accepted and N_{reject} is the number of connections rejected.

C. Simulation Experiments

We have written a discrete event simulation program to simulate the ATM network. The simulation program is written in the C++ programming language in a SUN/Solaris environment. We used a randomly generated synthetic workload which models video communications over an ATM network. Although the quantitative accuracy of our results are not validated, we believe that the qualitative conclusions based on the simulation data are correct. There are five system parameters for our video communication simulation: λ is the mean of inter-arrival rate between connections; μ is the mean of the life-time of each connection; ρ and β are as defined in earlier sections for connection utilization and burstiness and D is the relative deadline for a connection. For each connection, ρ, β and D are chosen from uniform distribution with D having a minimum value of β/ρ. The inter-arrival time and the lifetime of connections are chosen from exponential distributions. Hence, the estimated utilization of the server at the N-th layer is $U = \lambda\rho/\mu$. All simulations reported here are based on a network with 4 layers and 15 switches.

For the priority assignment, we use the Deadline Monotonic policy in which the priorities of connections are assigned in accordance to their relative deadlines. The smaller the deadline, the higher the priority. For each run of our simulation experiments, we generate over 5,000 connections according to the predefined system parameters and record the admission probability. We started the simulation with no connection in the network initially. To eliminate the warm-up effect, the system was allowed to reach the steady state before collecting the data. In particular, we let the system run through at least U/ρ number of connections before recording the data. For each connection arrival, the worst case end-to-end delays under a particular method were calculated. The system then decided to accept or reject the newly arrived connection.

D. Simulation Results

Fig. 6 shows the admission probability for the network systems when using different delay derivation methods. All three methods show similar trends between the admission probability and the total utilization. When the total utilization is low ($< 30\%$), almost all the connections are accepted no matter which method is used. As the total utilization increases, the admission probabilities start to decrease. That is because some newly arrived connections have to be rejected so as to keep the guaranteed delays given to the previous connections. Finally, when the system is almost fully utilized ($\sim 100\%$), the admission probability drops to around 0.78 for SEQ, 0.75 for GSC and 0.71 (for the decomposed method) respectively. As expected, throughout the entire spectrum of system utilization, both integrated methods work better than the decomposed method. And between the two integrated methods, our new method (SEQ) always performs better.

Fig. 7 shows the performance of the three methods under the test of deadline sensitivity. In this experiment, we fixed the total utilization at 75% and varied the relative deadline to the connections. From Fig. 7, we observe that with very tight deadlines, the admission probability is low. As the deadline to the connections increase, the admission probabilities for all three methods increase. With relatively large deadlines, almost 99% of the connections are admitted to the network. Again, our integrated method performs well throughout the deadline sensitivity test and performs especially well with tight deadline assignments. We also tested the different methods for traffic burstiness. Recall that the burstiness of connection traffic is defined by its β value. The results are shown in Fig. 8. With less bursty traffic, the admission probability for all three methods is higher. As the burstiness increases, all three methods experience performance degradation in terms of admission probability. Among the three methods, the decomposed method is the most affected. Our integrated method (SEQ) performs the best and can maintain the admission probability at about 90% even for large β settings.

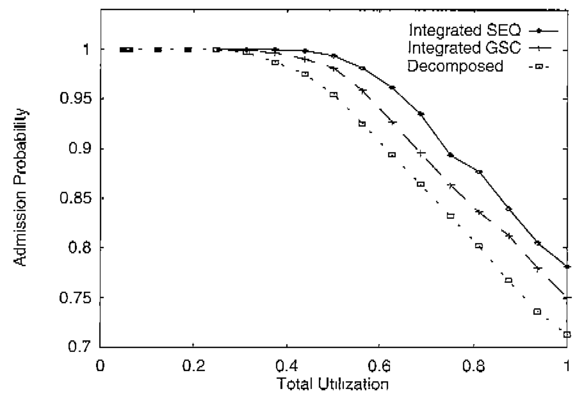


Fig. 6. Admission probability vs. total utilization.

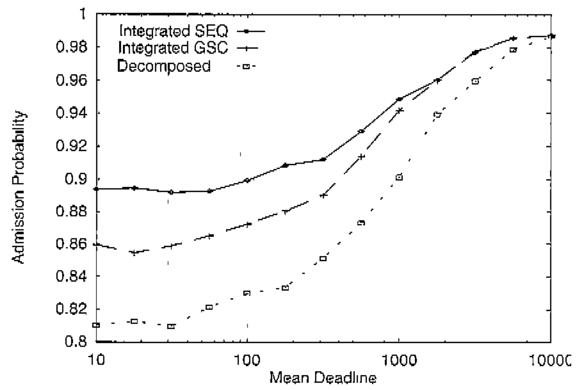


Fig. 7. Sensitivity to deadlines (Utilization = 75%).

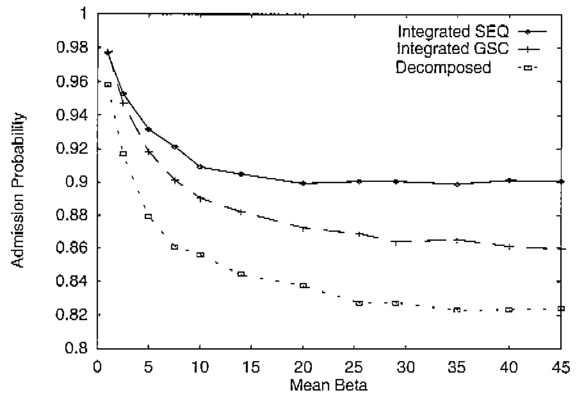


Fig. 8. Sensitivity to burstiness (Utilization = 75%).

V. CONCLUSIONS

In this paper, we have proposed and analyzed a new method for deriving the worst case end-to-end delay bounds in ATM networks. We have shown both analytically and through extensive simulations that our new method performs better than other conventional methods. This is observed through both analytical comparisons and extensive simulation experiments.

For future investigation, the work reported in this paper can be extended in several ways. Our methodology can be extended to handle networks with different topologies. When dealing with a non-tree topology, we ought to carefully take into account those connections that branch out from a server, so that we will not overestimate the traffic and delay at the next layer. On the other hand, in this paper, we adopted the Deadline Monotonic priority assignment method, which is a homogeneous priority assignment scheme. Integrated end-to-end delay analysis under heterogeneous priority assignment schemes should also be investigated.

APPENDIX A: PROOFS OF LEMMAS AND THEOREMS [1]

Lemma 1: A service curve $C_{i,k}(I)$ guaranteed by the k-th layer server to connection M_i is given as follows:

$$C_{i,k}(I) = \begin{cases} 0, & 0 \leq I \leq \theta_{i,k}, \\ (1-\pi_{i,k})(I-\theta_{i,k}), & I > \theta_{i,k}, \end{cases} \quad (6)$$

where $\pi_{i,k}$ *and* $\theta_{i,k}$ are defined as follows:

$$\pi_{i,k} = \sum_{j \in G_{i,k}} \rho_j \quad (7)$$

and

$$\theta_{i,k} = \frac{\sum_{j \in G_{i,k}} \beta_j}{1-\pi_{i,k}}. \quad (8)$$

Proof: For any time $t > 0$, according to [1], we have

$$R_{i,k+1}(t) = \left[(t-s) - \sum_{j \in G_{i,k}} (R_{j,k+1}(t) - R_{j,k+1}(s))\right]^+ + R_{i,k}(s), \quad \text{(A-1)}$$

where s is the starting point of the maximum busy period of connection M_i which contains time t at the k-th layer server.

Observe that if, for very small $\varepsilon > 0$, $s - \varepsilon$ is not in any maximum busy period of connection M_i at the k-th layer server, $s-\varepsilon$ is not in any maximum busy period of connection M_j at the j-th layer server for $j = 1, 2, \ldots, k-1$. Hence, at time s all arrival traffic coming to the first layer server from connection M_i has been transmitted from the k-th layer server, i.e., $R_{j,k+1}(s) = R_{j,1}(s)$. On the other hand, $R_{j,k+1}(t) \leq R_{j,1}(t)$. Therefore, for $j \in G_{i,k}$,

$$R_{j,k+1}(t) - R_{j,k+1}(s) = R_{j,k+1}(t) - R_{j,1}(s) \leq R_{j,1}(t) - R_{j,1}(s) \leq F_j(t-s). \quad \text{(A-2)}$$

Using (A-1) and (A-2), we have

$$\begin{aligned} R_{j,k+1}(t) - R_{i,k}(s) &= \left[(t-s) - \sum_{j \in G_{i,k}} (R_{j,k+1}(t) - R_{j,k+1}(s))\right]^+ \\ &\geq \left[(t-s) - \sum_{j \in G_{i,k}} (R_{j,1}(t) - R_{j,1}(s))\right]^+ \\ &\geq \left[(t-s) - \sum_{j \in G_{i,k}} F_{i,1}(t-s)\right]^+ . \end{aligned} \quad \text{(A-3)}$$

[1] These proofs can also be found [24] available at (http://www.comp.hkbu.edu.hk/~jng/Tech-Rpt/).

Thus, by Definition 7,

$$C_{i,k}(I) = \left[I - \sum_{j \in G_{i,k}} F_{j,1}(I)\right]^+ \quad \text{(A-4)}$$

is a service curve guaranteed by the k-th layer server to connection M_i. Finally, substituting (2) into (A-4) with some algebraic manipulations, we have

$$C_{i,k}(I) = \begin{cases} 0, & 0 \leq I \leq \theta_{i,k}, \\ (1-\pi_{i,k})(I-\theta_{i,k}), & I > \theta_{i,k}. \end{cases}$$

□

Lemma 2: A network service curve guaranteed to connection M_i is given as follows:

$$C_i^{End-to-end}(I) = \begin{cases} 0, & 0 \leq I \leq \sum_{k=1}^{N} \theta_{i,k}, \\ (1-\pi_{i,N}) \left(I - \sum_{k=1}^{N} \theta_{i,k}\right), & I > \sum_{k=1}^{N} \theta_{i,k}, \end{cases} \quad (9)$$

where $\pi_{i,N}$ *and* $\theta_{i,k}$ are defined in (7) and (8), respectively.

Proof: By Lemma 1, for connection M_i, the k-th layer server guarantees a service curve $C_{i,k}(I)$:

$$C_{i,k}(I) = \begin{cases} 0, & 0 \leq I \leq \theta_{i,k}, \\ (1-\pi_{i,k})(I-\theta_{i,k}), & I > \theta_{i,k}. \end{cases} \quad \text{(A-5)}$$

According to Theorem 3 in [9], we have

$$C_i^{End-to-end}(I) = \min\left\{\sum_{k=1}^{N} C_{i,k}(x_k) : x_k \geq 0, \text{ and} \sum_{k=1}^{N} = I\right\}. \quad \text{(A-6)}$$

First, we show that if $I = \sum_{k=1}^{N} \theta_{i,k}, C_i^{End-to-end}(I) = 0$. Let $x_k = \theta_{i,k}$. Then, we have

$$\begin{aligned} \sum_{k=1}^{N} C_{i,k}(x_k) &= \sum_{k=1}^{N} C_{i,k}(\theta_{i,k}) \\ &= \sum_{k=1}^{N} 0 \\ &= 0. \end{aligned} \quad \text{(A-7)}$$

Therefore, $C_i^{End-to-end}(I) = 0$. Since any service curve is a non-decreasing function, for $I \leq \sum_{k=1}^{N} \theta_{i,k}$,

$$C_i^{End-to-end}(I) = 0. \quad \text{(A-8)}$$

Second, we consider the case when $I > \sum_{k=1}^{N} \theta_{i,k}$. For any partition $\left\{x_k : k = 1, \ldots, N, \sum_{k=1}^{N} x_k = I, \ x_k \geq 0\right\}$ such that

$I > \sum_{k=1}^{N} \theta_{i,k}$, there must exist some $x_k > \theta_{i,k}$. Let $H = \{k : x_k > \theta_{i,k}\}$. Then

$$\begin{aligned}\sum_{k\in H}(x_k - \theta_{i,k}) &\geq \sum_{k\in H}(x_k - \theta_{i,k}) + \sum_{k\notin H}(x_k - \theta_{i,k})\\ &= \sum_{k=1}^{N}(x_k - \theta_{i,k})\\ &= \sum_{k=1}^{N} x_k - \sum_{k=1}^{N}\theta_{i,k}\\ &= I - \sum_{k=1}^{N}\theta_{i,k}.\end{aligned} \quad \text{(A-9)}$$

Furthermore, $\pi_{i,k} \leq \pi_{i,N}$ because of $G_{i,k} \subseteq G_{i,N}$. Hence,

$$\begin{aligned}\sum_{k=1}^{N} C_{i,k}(x_k) &= \sum_{k\in H} C_{i,k}(x_k)\\ &= \sum_{k\in H}(1-\pi_{i,k})(x_k - \theta_{i,k})\\ &\geq \sum_{k\in H}(1-\pi_{i,N})(x_k - \theta_{i,k})\\ &= (1-\pi_{i,k})\sum_{k\in H}(x_k - \theta_{i,k})\\ &\geq (1-\pi_{i,N})\left(I - \sum_{k=1}^{N}\theta_{i,k}\right).\end{aligned} \quad \text{(A-10)}$$

According to Lemma 1, if $I - \sum_{k=1}^{N-1}\theta_{i,k} > \theta_{i,N}$, $C_{i,N}\left(I - \sum_{k=1}^{N-1}\theta_{i,k}\right) = (1 - \pi_{i,k})\left(I - \sum_{k=1}^{N}\theta_{i,k}\right)$. Otherwise, $C_{i,N}\left(I - \sum_{k=1}^{N-1}\theta_{i,k}\right) = 0$. Therefore,

$$\begin{aligned}\sum_{k=1}^{N} C_{i,k}(x_k) &\geq C_{i,N}\left(I - \sum_{k=1}^{N-1}\theta_{i,k}\right)\\ &= \sum_{k=1}^{N-1} C_{i,k}(\theta_{i,k}) + C_{i,N}(I - \sum_{k=1}^{N-1}\theta_{i,k}).\end{aligned} \quad \text{(A-10')}$$

This means

$$C_i^{End-to-end}(I) \geq \sum_{k=1}^{N-1} C_{i,k}(\theta_{i,k}) + C_{i,N}\left(I - \sum_{k=1}^{N-1}\theta_{i,k}\right). \quad \text{(A-11)}$$

On the other hand, $\left\{\theta_{i,1}, \theta_{i,2}, \ldots, \theta_{i,N-1}, I - \sum_{k=1}^{N-1}\theta_{i,k}\right\}$ is one of I's partition. According to (A-6), we know that

$$C_i^{End-to-end}(I) \leq \sum_{k=1}^{N-1} C_{i,k}(\theta_{i,k}) + C_{i,N}\left(I - \sum_{k=1}^{N-1}\theta_{i,k}\right). \quad \text{(A-12)}$$

Hence, for $I > \sum_{k=1}^{N}\theta_{i,k}$, we have

$$C_i^{End-to-end}(I) = \sum_{k=1}^{N-1} C_{i,k}(\theta_{i,k}) + C_{i,N}\left(I - \sum_{k=1}^{N-1}\theta_{i,k}\right). \quad \text{(A-13)}$$

□

Theorem 1: With the integrated method by guaranteed service curve, the end-to-end delay $d_i^{End-to-end}$ experienced by connection M_i is bounded by

$$d_i^{End-to-end} \leq \sum_{k=1}^{N}\theta_{i,k} + I_i\frac{\pi_{i,N}}{1-\pi_{i,N}}, \quad (11)$$

where $I_i, \pi_{i,N}$, and $\theta_{i,k}$ are defined in (3), (7) and (8), respectively.

Proof: According to Theorem 2.7 in [32], the worst case end-to-end delay bound satisfies the following inequality

$$d_i^{End-to-end} \leq \max_{c\geq 0}\left\{C_i^{-1^{End-to-end}}(c) - F_{i,1}^{-1}(c)\right\}. \quad \text{(A-14)}$$

According to (9) and (2), we have

$$C_i^{-1^{End-to-end}}(c) = \sum_{k=1}^{N}\theta_{i,k} + \frac{1}{1-\pi_{i,N}} * c \quad \text{(A-15)}$$

and

$$F_{i,1}^{-1}(c) = \begin{cases} c, & 0 \leq c < I_i,\\ -\dfrac{\beta_i}{\rho_i} + \dfrac{1}{\rho_i} * c, & I_i \leq c.\end{cases} \quad \text{(A-16)}$$

So,

$$\begin{aligned}&C_i^{-1^{End-to-end}}(c) - F_{i,1}^{-1}(c)\\ &= \begin{cases}\displaystyle\sum_{k=1}^{N}\theta_{i,k} + \frac{\pi_{i,N}}{1-\pi_{i,N}} * c, & 0 \leq c < I_i,\\ \displaystyle\sum_{k=1}^{N}\theta_{i,k} + \frac{\beta_i}{\rho_i} + \left(\frac{1}{1-\pi_{i,N}} - \frac{1}{\rho_i}\right) * c, & I \leq c.\end{cases}\end{aligned} \quad \text{(A-17)}$$

Since we assume that $\sum_{k=1}^{m}\rho_k \leq 1$, this results in $1 - \pi_{i,N} \geq \rho_i$. Therefore, we have that $\frac{1}{1-\pi_{i,N}} - \frac{1}{\rho_i} \leq 0$. Consequently, when $c < I_i$, the derivative of $C_i^{-1^{End-to-end}}(c) - F_{i,1}^{-1}(c)$ is positive. When $c > I_i$, the derivative of $C_i^{-1^{End-to-end}}(c) - F_{i,1}^{-1}(c)$ is negative. This means that $C_i^{-1^{End-to-end}}(c) - F_{i,1}^{-1}(c)$ has the global maximum value at $c = I_i$. Thus,

$$\max_{c\geq 0}\left\{C_i^{-1^{End-to-end}}(c) - F_{i,1}^{-1}(c)\right\} = \sum_{k=1}^{N}\theta_{i,k} + I_i\frac{\pi_{i,N}}{1-\pi_{i,N}}. \quad \text{(A-18)}$$

□

Lemma 4: In a multi-server system, the output traffic function of connection is given as follows:

$$\begin{aligned} &R_i^{OUT,M}(t) \\ &= \inf_{s\le t}\left\{\left[(t-s)-\sum_{j=1}^{i-1}\left(R_j^{OUT,M}(t)-R_j^{OUT,M}(s)\right)\right]^+ \right. \\ &\quad \left. + R_i^{N-1}(s)\right\} \\ &= \left[(t-s'')-\sum_{j=1}^{i-1}\left(R_j^{OUT,M}(t)-R_j^{OUT,M}(s'')\right)\right]^+ \\ &\quad + R_i^{IN,M}(s''), \end{aligned} \tag{14}$$

where s'' is the starting point of the maximum busy period of connection M_i, which contains t, at the N-th layer server. If time t is not in any connection M_i's maximum busy period of connection M_i at the N-th layer server, then $s''=t$.

Proof: According to [1], for any $t>0$,

$$\begin{aligned} &R_i^{OUT,M}(t) \\ &= \inf_{s\le t}\left\{\left[(t-s)-\sum_{j=1}^{i-1}\left(R_j^{OUT,M}(t)-R_j^{OUT,M}(s)\right)\right]^+ \right. \\ &\quad \left. + R_i^{N-1}(s)\right\} \\ &= \left[(t-s'')-\sum_{j=1}^{i-1}\left(R_j^{OUT,M}(t)-R_j^{OUT,M}(s'')\right)\right]^+ \\ &\quad + R_i^{N-1}(s''), \end{aligned} \tag{A-19}$$

where s'' is the starting point of the maximum busy period of connection M_i, which contains t, at the N-th layer server. Observe that if $s''-\varepsilon$ is not in any maximum busy period of connection M_i at the N-th layer server for very small $\varepsilon>0$, $s''-\varepsilon$ is not in any maximum busy period of connection at the j-th layer server for $j=1,2,\ldots,N-1$. Hence at time s'', all arrival traffic coming to the first layer server from connection M_i has been transmitted from the N-th layer server, i.e., $R_i^{IN,M}(s'')=R_i^{N-1}(s'')$. Therefore, we have

$$\begin{aligned} &R_i^{OUT,M}(t) \\ &= \inf_{s\le t}\left\{\left[(t-s)-\sum_{j=1}^{i-1}\left(R_j^{OUT,M}(t)-R_j^{OUT,M}(s)\right)\right]^+ \right. \\ &\quad \left. + R_i^{N-1}(s)\right\} \\ &= \left[(t-s'')-\sum_{j=1}^{i-1}\left(R_j^{OUT,M}(t)-R_j^{OUT,M}(s'')\right)\right]^+ \\ &\quad + R_i^{IN,M}(s''). \end{aligned} \tag{A-20}$$

□

Theorem 2: A multi-server priority driven tree system is equivalent to a single server priority driven system if the priorities of connections are assigned in the same way, i.e., if $R_i^{IN,S}(t)=R_i^{IN,M}(t)$, then $R_i^{OUT,S}(t)=R_i^{OUT,M}(t)$, $i=1,2,\ldots,m$.

Proof: We prove this theorem by induction.

Let $n=1$. In the single server priority driven system, by Lemma 3, the output traffic function of connection M_i is

$$R_1^{OUT,S}(t)=(t-s')+R_1^{IN,S}(s'), \tag{A-21}$$

where s'' is the starting point of the maximum busy period of connection M_i which contains t.

In the multi-server priority driven tree system, by Lemma 4, the output traffic function of connection M_i is

$$R_1^{OUT,M}(t)=(t-s'')+R_{1,N}(s''), \tag{A-22}$$

where s'' is the starting point of the maximum busy period of connection M_i, which contains t, at the N-th layer server. Since connection M_1 has the highest priority, $s'=s''$ and $R_1^{IN,M}(s'')=R_{1,N}(s'')$. This means that for connection $M_1$the maximum busy periods at the single server system and at the N-th layer server of the multi-server system are the same and

$$R_1^{OUT,M}(t)=R_1^{OUT,S}(t). \tag{A-23}$$

Let $n>1$. Suppose that for $1\le i<n$, $R_1^{OUT,S}(t)=R_1^{OUT,M}(t)$ and connection M_i has the same busy periods at the single server system and at the N-th layer server of the multi-server system. We claim that $R_n^{OUT,S}(t)=R_n^{OUT,M}(t)$ and connection M_n has the same maximum busy periods at the single server system and at the N-th layer server of the multi-server system.

By Lemma 3 and Lemma 4, we have

$$\begin{aligned} &R_n^{OUT,M}(t) \\ &= \left[(t-s'')-\sum_{j=1}^{n-1}\left(R_j^{OUT,M}(t)-R_j^{OUT,M}(s'')\right)\right]^+ \\ &\quad + R_n^{IN,M}(s''), \end{aligned} \tag{A-24}$$

and

$$\begin{aligned} &R_n^{OUT,S}(t) \\ &= \left[(t-s')-\sum_{j=1}^{n-1}\left(R_j^{OUT,S}(t)-R_j^{OUT,S}(s')\right)\right]^+ \\ &\quad + R_n^{IN,S}(s'). \end{aligned} \tag{A-25}$$

Because for each connection M_i, $i<n$, the single server system and the N-th layer server in the multi-server system have the same maximum busy periods, the remaining capabilities of systems that connection M_n can utilize are the same. Therefore the maximum busy periods of connection M_n, which contains time t at the single server system and at the N-th layer server of

the multi-server system are identical, i.e., $s' = s''$. From (A-24), (A-25) and the induction hypothesis, we have

$$R_n^{OUT,S}(t) = R_n^{OUT,M}(t). \quad \text{(A-26)}$$

□

APPENDIX B: DELAY ANALYSIS BY THE DECOMPOSED METHOD.

The decomposed method was first proposed in [5] to derive the end-to-end delay bound. We include the derivation here for the sake of completeness. The formula derived here allows us to compare the decomposed method via the integrated method. The following lemma is one of the famous results obtained by. Cruz in [5]. It states that the shape of arrival curve of a connection inside network remains the same except that the burst parameter increases as the number of hops visited by the connection increases.

Lemma 5: For connection M_i, if the worst case delay $d_{i,j}$, $j = 1, 2, \ldots, k-1$, is known, the arrival curve of connection $F_{i,k}(t)$ of M_i at the k-th layer server is given as follows:

$$F_{i,k}(I) = \begin{cases} I, & 0 \le I \le I_{i,k}, \\ \beta_{i,k} + \rho_i * I, & I_{i,k} \le I, \end{cases} \quad \text{(B-1)}$$

where $I_{i,k}, \beta_{i,1}, \beta_{i,k}$ are defined as following:

$$\beta_{i,1} = \beta_i, \quad \text{(B-2)}$$
$$\beta_{i,k} = \beta_{i,k-1} + \rho_i * d_{i,k-1}, \quad \text{(B-3)}$$

and

$$I_{i,k} = \frac{\beta_{i,k}}{1 - \rho_i}. \quad \text{(B-4)}$$

Proof: See Lemma 1 in [5] and Theorem 6 in [19].

After the traffic description for every connection at each server is given, it is easy to derive the following formula. □

Theorem 4: The worst case delay $d_{i,k}$ experienced by connection M_i at the k-th layer server is bounded by

$$d_{i,k} \le \theta'_{i,k} + I_{i,k} * \frac{\pi_{i,k}}{1 - \pi_{i,k}}, \quad \text{(B-5)}$$

where $\pi_{i,k}$ is defined in (7), $I_{i,k}$ is defined in (B-4) and $\theta'_{i,k}$ defined as follows:

$$\theta'_{i,k} = \frac{\sum_{j \in G_{i,k}} \beta_{j,k}}{1 - \pi_{i,k}}. \quad \text{(B-6)}$$

Proof: At the k-th layer server, the input traffic of connection M_i is constrained by arrival curve $F_{i,k}(I)$. Hence, a service curve guaranteed by the k-th layer server to connection M_i is given by

$$C_{i,k}(I) = \max\left\{0, I - \sum_{j=1}^{i-1} F_{i,j}(I)\right\} = \begin{cases} 0, & 0 \le I \le \theta'_{i,k}, \\ (1 - \pi_{i,k})(I - \theta'_{i,k}), & \theta'_{i,k} \le I. \end{cases} \quad \text{(B-7)}$$

According to Theorem 2.7 in [31], we have

$$d_{i,k} \le \max_{c \ge 0} \left\{ C_{i,k}^{-1}(c) - F_{i,k}^{-1}(c) \right\}. \quad \text{(B-8)}$$

Substituting (B-1) and (B-7) into (B-8) with certain algebraic manipulations, we have

$$d_{i,k} \le \theta'_{i,k} + I_{i,k} * \frac{\pi_{i,k}}{1 - \pi_{i,k}}. \quad \text{(B-9)}$$

□

Now, an upper bound of the end-to-end worse case delay can be obtained by summing up the worst case delays at individual servers.

Theorem 5: The worst case end-to-end delay $d_i^{End-to-end}$ experienced by connection M_i is bounded by

$$d_i^{End-to-end} \le \sum_{k=1}^{N} \left(\theta'_{i,k} + I_{i,k} * \frac{\pi_{i,k}}{1 - \pi_{i,k}} \right), \quad \text{(B-10)}$$

where $\theta'_{i,k}, I_{i,k}$ and $\pi_{i,k}$ are defined as in (B-6), (B-4) and (7) respectively.

REFERENCES

[1] R. Agrawal and R. Rajan, "Performance bounds for guaranteed and adaptive service," *IBM Research Report*, May, 1996.

[2] J. Bennett and H. Zhang, "WF^{2}Q: Worst-case fair weighted fair queuing," in *Proc. of INFOCOM*, 1996.

[3] J. Boudec, "Application of network calculus to guaranteed service networks," *IEEE Trans. Inform. Theory*, Accepted for publication, 1998.

[4] C. S. Chang, "Stability, queue length, and delay of deterministic and stochastic queuing networks," *IEEE Trans. Automat. Contr.*, vol. 39, no. 5, May 1994.

[5] R. L. Cruz, "A calculus for network delay, Part I & II," *IEEE Trans. Inform. Theory*, vol. 37, no. 1, Jan. 1991.

[6] R. L. Cruz, "Service burstiness and dynamic burstiness measures: A framework," *J. High Speed Networks*, vol. 1, no. 2, 1992.

[7] R. L. Cruz and M. C. Chuah, "Consequences of rate control in exhaustive service polling systems," in *Proc. 27th Ann. Conf. Inform. Sci. Syst.*, Johns Hopkins University, Mar. 1993.

[8] R. L. Cruz and H. Liu, "End-to-end queuing delay in ATM networks," *J. High Speed Networks*, vol. 3, no. 4, 1994.

[9] R. L. Cruz, "Quality of service guarantees in virtual circuit switched networks," *IEEE. J. Select. Areas Commun.*, vol. 13, no. 6, 1995.

[10] D. Ferri and D. C. Verma, "A scheme for real-time channel establishment in wide-area networks," *IEEE J. Select. Areas Commun.*, vol. SAC-8, no. 3, Apr. 1990.

[11] L. Georgiadis, R. Guerin, V. Peris, and K. N. Sivarajan, "Efficient network QoS provisioning based on per node traffic shaping," in *Proc. of IEEE INFOCOM*, 1996.

[12] M. G. Hluchyj and M. J. Karol, "Queueing in high performance packet switching," *IEEE J. Select. Areas Commun.*, vol. 6, no. 9, Sept. 1988.

[13] C. Han, C. Hou, and K. Shin, "On slot allocation for time-constrained messages in DQDB networks," in *Proc. of IEEE INFOCOM*, 1995.

[14] C. Han and K. Shin, "Real-time communication in FieldBus multiaccess networks," in *Proc. of IEEE RTAS*, 1995.

[15] C. Han, C. Hou, and K. Shin, "On slot allocation fortime-constrained messages in dual-bus networks," to appear in *IEEE Trans. Computers*, July 1997.

[16] E. Knightly and H. Zhang, "Traffic characterization and switch utilization using deterministic bounding interval dependent traffic models," in *Proc. of IEEE INFOCOM*, 1995.

[17] J. Kurose, "On computing per-session performance bounds in high-speed multi-hop computer networks," in *Proc. of ACM Sigmetrics, Perform.*, 1992.

[18] C. Lee, K. Yoshida, C. Mercer, and R. Rajkumar, "Predictable communication protocol processing in real-time mach," in *Proc. of RTAS*, June 1996.

[19] C. Li, R. Bettati, and W. Zhao, "Static priority scheduling for ATM networks with static priority scheduling," in *Proc. of RTSS*, 1997.

[20] J. Liebeherr, D. Wrege, and D. Ferrari, "Exact admission control in networks with bounded delay services," *IEEE/ACM Trans. Networking*, vol. 4, no. 6, Dec. 1996.

[21] Mukherjee, Tripathi, Saha, and Saksena, "A bandwidth allocation scheme for time constrained message transmission on a slotted ring LAN," in *Proc. of RTSS*, 1993.

[22] J. Ng, S. Song, and W. Zhao, "Integrated delay analysis of regulated ATM switch for real-time applications," *Technical Report*, Dept. of Computing Studies, Hong Kong Baptist University, May 1997. (http://www.comp.hkbu.edu.hk/~jng/Tech-Rpt).

[23] J. Ng, S. Song, and W. Zhao, "Integrated delay analysis of regulated ATM switch," in *Proc. of RTSS*, 1997.

[24] J. Ng, S. Song, C. Li, and W. Zhao, "Integrated end-to-end delay analysis in ATM networks," *Technical Report*, Dept. of Computer Science, Hong Kong Baptist University, July 1998. (http://www.comp.hkbu.edu.hk/~jng/Tech-Rpt).

[25] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: the single node case," *IEEE/ACM Trans. Networking*, vol. 1, no. 3, June 1993.

[26] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: the multiple node case," *IEEE/ACM Trans. Networking*, vol. 2, no. 2, Apr. 1994.

[27] A. Raha, S. Kamat, and W. Zhao, "Guaranteeing end-to-end deadlines in ATM networks," in *Proc. of ICDCS*, 1995.

[28] A. Raha, S. Kamat, and W. Zhao, "Admission control for hard real-time connections in ATM networks," in *Proc. of IEEE INFOCOM*, Mar. 1996.

[29] R. Rooholamini, V. Cherkassky, and M. Garver, "Finding the right ATM switch for the market," *IEEE Computer*, vol. 27, no. 4, Apr. 1994.

[30] Saha, Tripathi, Saksena, and Mukherjee, "On guaranteed delivery of time-critical messages in DQDB," in *Proc. of IEEE INFOCOM*, 1994.

[31] H. Sariowan, L. Cruz, and G. Polyzos, "Scheduling for quality of service guarantees via service curves," in *Proc. of ICCCN*, 1995.

[32] H. Sariowan, *A Service-curve Approach to Performance Guarantees in Integrated-service Networks*, Ph. D. disertation. University of California, San Diego, 1996.

[33] D. Wrege, E. Knightly, H. Zhang, and J. Liebeherr, "Deterministic delay bounds for VBR video in packet-switching networks: fundamental limits and practical trade-offs," *IEEE/ACM Trans. Networking*, vol. 4, no. 3, June, 1996.

[34] D. E. Wrege and J. Liebeherr, "A near-optimal packet scheduler for QoS networks," in *Proc. of IEEE INFOCOM*, 1997.

[35] D. Yates, J. Kurose, D. Towsley, and M. G. Hluchyj, "On per-session delay distributions and the call admission problem for real-time applications with QoS requirements," in *Proc. ACM SIGCOMM*, 1993.

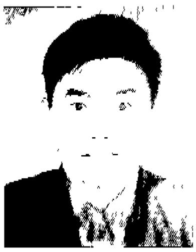
[36] H. Zhang and D. Ferrari, "Rate-controlled service disciplines," *J. High Speed Networks*, vol. 3, no. 4, 1994.

[37] H. Zhang, "Service disciplines for guaranteed performance service in packet switching networks," in *Proc. IEEE*, 1995.

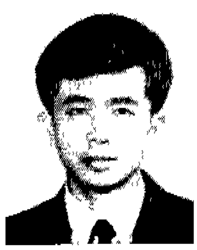
[38] H. Zhang, "Providing end-to-end performance guarantees using non-work-conserving disciplines," *Computer Commun.: Special Issue on System Support for Multimedia Computing*, vol.18, no. 10, 1995.

[39] Q. Zheng and K. G. Shin, "On the ability of establishing real-time channels in point-to-point packet-switched networks," *IEEE Trans. Commun.*, 1996.

Joseph Kee-Yin Ng received a B.S. in Mathematics and Computer Science, a M.Sc. in Computer Science, and a Ph.D. in Computer Science from the University of Illinois at Urbana-Champaign in the years 1986, 1988, and 1993, respectively. Dr. Ng is currently an associate professor in the Department of Computer Science at Hong Kong Baptist University. His research areas include: Real-Time Networks, Multimedia Communication, ATM Delay Analysis, High-Speed Network Simulation, and Distributed Systems Performance Evaluation. He presently serves as the General and Program Co-Chair for The International Computer Congress 1999 (ICC'99), the Program Co-Chair for The Sixth International Conference on Real-Time Computing Systems and Applications (RTCSA'99) and the General Co-Chair for The 1999 International Computer Science Conference (ICSC'99). Dr. Ng is a member of the IEEE and the IEEE Computer Society since 1991, and has been an exco-member (1993-95), General Secretary (1995-1997), and Vice-Chair (1997-present) of the IEEE, Hong Kong Section, Computer Chapter. He is also a member of the IEEE Communication Society, ACM, and EUROMICRO.



Shibin Song received his Diploma in Mathematics from Hubei TV University, and received the M.Sc. degree in probability and statistics from Zhongshan University (Sun Yat-sen University) in 1985 and 1991, respectively. Mr. Song is currently a lecturer and a PhD candidate in the Department of Mathematics at Zhongshan University. His research interests include: Stochastic Point Processes, Queueing Theory, Reliability, High-Speed Network Performance Evaluation, and Real-time Systems.



Chengzhi Li received the B.S. degree in Applied Mathematics from Fuzhou University in 1982, the M.S. degree in System Science from Xiamen University in China in 1985. From 1985 to 1991, he was a faculty at the department of computer science and system science in Xiamen University in China. He is currently a research assistant in the real-time system group at the department of computer science in the Texas A&M University in U.S.A., where he is working toward the Ph.D. degree. His research interests are in Real-time multimedia communication, Internet telephony, IP QoS and traffic engineering.

Wei Zhao received the B.S. degree in Physics from Shaanxi Normal University in 1977, and the M.S. and Ph.D. in Computer & Info. Sciences from University of Mass., Amherst in 1983, and 1986, respectively. Professor Zhao is currently the department head of the Computer Science Department in Texas A&M. University. He leads the Real-Time Systems Research Group and his research interests include secured real-time computing and communication, distributed operating systems, database systems and fault tolerant systems. Professor Zhao was an Editor of the IEEE Transactions on Computers. He was also the General and Program Chair of The IEEE Real-Time Technology and Applications Symposium in 1995 and 1996, respectively. Zhao presently serves as Program and General Chair of the IEEE Real-Time Systems Symposium, 1999-2000. Professor Zhao is a member of the ACM, the IEEE, and the IEEE Computer Society.