香港浸會大學
HONG KONG BAPTIST UNIVERSITY

FACULTY OF SCIENCE
DEPARTMENT OF COMPUTER SCIENCE
TECHNICAL REPORT (JNG07-98) **JULY, 1998**

Title:

Integrated End-to-end Delay Analysis in ATM Networks $^{\alpha}$

Authors:

Joseph Kee-Yin, NG (Dept. of Computer Science, HKBU)
Shibin, SONG (Dept. of Mathematics, ZhongShan U)
Chengzhi, LI (Dept. of Computer Science, Texas A & M)
Wei, ZHAO (Dept. of Computer Science, Texas A & M)

ABSTRACT:

For admitting a hard real-time connection to an ATM network, it requires that the end-to-end delays of cells belonging to the connection meet their deadline 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 <u>*decomposed*</u> *method will analyze the worst case delay for each switch and then compute the total delay as the sum of the delays at individual switches. On the other hand, an* <u>*integrated*</u> *method will analyze all the switches involved in an integrated manner and derive the total delay directly. In this paper, we present an efficient and effective integrated method. We evaluate the network performance under different system parameters. We compare the performance of our proposed method with the existing decomposed and integrated methods [CRU91, CRU95, AGR96, BOU97].*

Keywords:

Real-Time Communication, ATM Networks, End-to-end Delay Analysis, Performance Evaluation.

$^{\alpha}$ This is a joint-research project between Hong Kong Baptist University and Texas A & M University, U.S.A. This work is supported by the Faculty Research Grant of Hong Kong Baptist University (FRG/96-97/II-103) and the Research Grant Council Earmarked Research Grant of Hong Kong (RGC/97-98/54).

Glossary of Terms

TERM	DEFINITION	SECTION of FIRST USE
$M = \{M_1, \ldots, M_m\}$	The set of m hard real-time connections.	2.2
P_i	The priority of connection M_i.	2.2
$R_{i,k}(t)$	The arrival function of connection M_i at the k-th layer server.	2.5
$F_{i,k}(I)$	The arrival curve for connection M_i at the k-th layer server.	2.5
β_i	The burst parameter of connection M_i.	2.5
ρ_i	The long term average rate of connection M_i.	2.5
I_i	The flex point of function $F_{i,1}(I)$.	2.5
$d_{i,k}$	The worst case delay experienced by connection M_i at the k-th layer server.	3.1
$d_i^{End-to-end}$	The worst case end-to-end delay experienced by connection M_i.	3.1
$G_{i,k}$	The set of indices of connections that have higher priorities than connection M_i and enter the same server at the k-th layer with connection M_i.	3.1
$C_{i,k}(t)$	The service curve guaranteed by the k-th layer server to connection M_i.	3.2
$C_i^{End-to-end}(t)$	The network service curve guaranteed to connection M_i.	3.2
$\pi_{i,k}$	The parameter of service curve $C_{i,k}(t)$.	3.2
$\theta_{i,k}$	The parameter of service curve $C_{i,k}(t)$.	3.2
$R_i^{IN,S}(t)$	The input traffic function of connection M_i in a single server system.	3.3
$R_i^{OUT,S}(t)$	The output traffic function of connection M_i in a single server system.	3.3
$R_i^{IN,M}(t)$	The input traffic function of connection M_i in a multiple server system.	3.3
$R_i^{OUT,M}(t)$	The output traffic function of connection M_i in a multiple server system.	3.3
P_{adm}	The admission probability.	4.3
N_{accept}	The number of accepted connections.	4.3
N_{reject}	The number of rejected connections.	4.3
$I_{i,k}$	The flex point of the arrival curve $F_{i,k}(I)$.	Appendix B.
$\beta_{i,k}$	The burst parameter of the arrival curve $F_{i,k}(I)$.	Appendix B.
$\theta'_{i,k}$	The parameter of service curve $C_{i,k}(t)$.	Appendix B.

List of Figures

1 A general ATM switch architecture. …….………………………………………… 3

2 An ATM LAN with 3 switches. ……………………………………………………. 4

3 Example of decomposing a network into servers. …………………………….. 5

4 Connection-server graph. ……………………………………………………….. 6

5 A single server system. …………………………………………………………. 9

6 Admission probability vs. total utilization. ……………………………..…... 13

7 Sensitivity of deadlines (Utilization 75%). ……………………………………. 14

8 Sensitivity of burstiness (Utilization 75%). …………………………………….. 14

1. Introduction

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

In terms of *effectiveness*, we mean that a delay analysis method 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*. That is, it should be simple and fast in order to be used in on-line connection admission.

There have been two different approaches to derive the end-to-end delay for an ATM network. A *decomposed* approach is to analyze the worst case cell delay for each ATM switch involved, and compute the worst case end-to-end delay as the 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 bound than the decomposed method and thus, improving the system effectiveness. However, such an integrated delay analysis may be very complicated and hence is sometimes difficult 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 a time interval. Then, the end-to-end worst case delay can be derived [CRU95, AGR96, BOU97]. 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. We take a *system equivalency* approach. That is, 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. This is observed through an extensive simulation study of systems with a wide range of parameter settings.

Our work complements the recent progress in real-time communications [HAN95a, HAN95b, LEE96, HAN97, SAH94, MUK93]. In particular, much of the previous work in ATM has been concentrating on obtaining the delay bounds and connection admission criteria for individual scheduling polices [CHA94, CRU91, CRU95, KNI95, RAH95, RAH96, WRE96]. Ferrari and Verma [FER90] and Zheng and Shin [ZHE96] studied the use of Earliest Deadline First scheduling in wide area networks. Zhang and Ferrari [ZHA94] discuss how local deterministic delay bounds can be guaranteed over an ATM link for 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 in [YAT93] and by Cruz in [CRU91, CRU95].

Analyzing delay in an integrated manner has been addressed by [CRU95, GEO96]. While Cruz provided a general framework for integrated analysis [CRU95], Georgiadis, Guerin, Peris, and Sivarajan concentrated on a network consisting of rate-controlled services [GEO96]. Furthermore, several research groups have also been investigating the admission control based on the delay analysis utilizing the service curves [AGR96, BOU97, SAR95].

The remainder of this paper is organized as follows. In Section 2, we give a brief review for the hard real-time connections, present our ATM network model, and introduce the traffic description functions. Section 3 discusses the two integrated methods in analyzing the worst case end-to-end delay. Section 4 is the performance comparison among the decomposed method, and the two integrated methods on the end-to-end delay. Section 5 concludes the paper with a summary and future work.

2. Models

2.1 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 Figure 1 for a general architecture of an ATM switch.

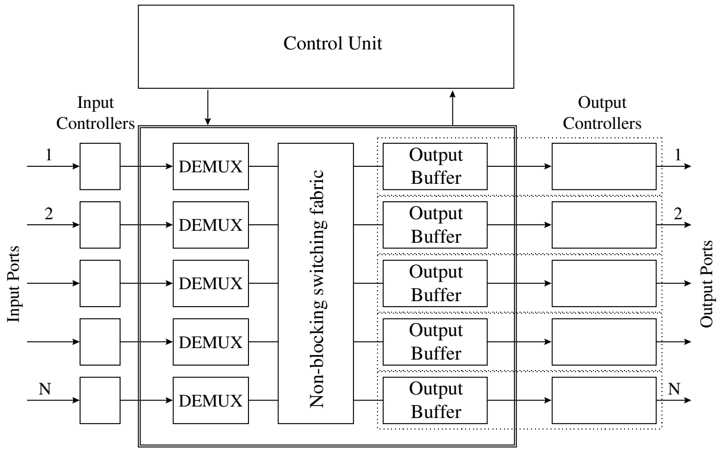

Figure 1. A general ATM switch architecture.

The scheduling policy at an output port controller of an ATM switch determines the order of the cells (from different connections) being transmitted. In this paper, we assume that a priority driven scheduler is utilized in the output ports of ATM switches. Priority driven schedulers are popular for real-time applications. 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 in terms of τ as one time unit. This simplifies our analysis without any loss of generality.

2.2 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 sent 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. For connection M_i, its deadline is D_i.

Let $M = \{M_1, M_2, \ldots, M_m\}$ 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 $i = 1, 2, \ldots, m$. M_1 has the highest priority and M_m has the lowest priority. Specific priority assignment methods will be discussed in Section 4.

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

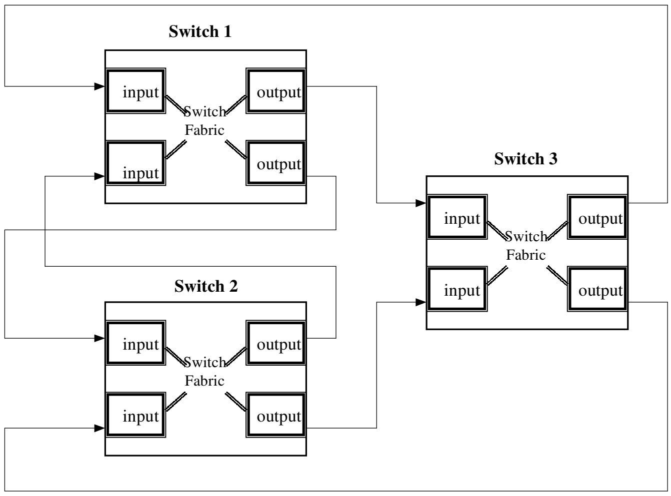

Figure 2. An ATM LAN with 3 switches.

2.3 Servers

To simplify the analysis, a concept of *server* was introduced [CRU91]. 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 ATM connection. We can, therefore, model an ATM network as a collection of servers.

The servers are classified into two categories: constant servers and variable servers [CRU91,RAH96]. 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, physical links and the 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 is 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 transmission on the link associated with the output port. Thus, cells may be transmitted in an order that is 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 this server may differ from that at the input of this server.

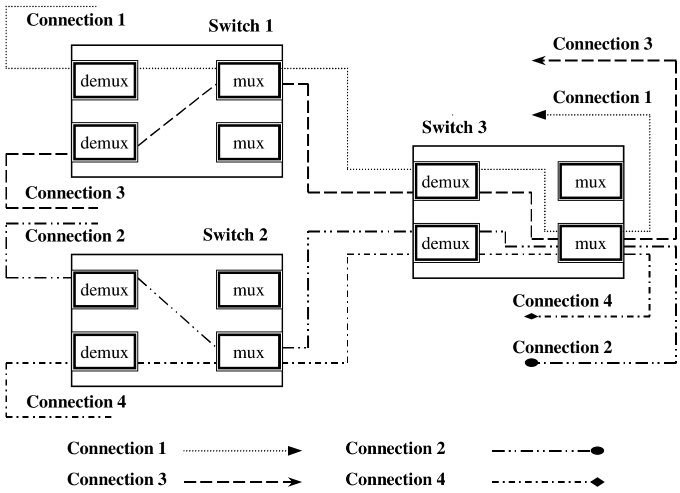

Figure 3. Example of decomposing a network into server.

2.4 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 the servers in immediate sequence. Figure 3 presents the connection-server graph for the network shown in Figure 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 constant servers from the deadline of a connection, the impact of constant servers can be eliminated. In the rest of the paper, we assume that the delay requirements of connections are modified in the way 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. Figure 4 shows the connection-server graph after eliminating the constant servers for the system shown in Figure 3. In 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 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 through a tree-structured network. That is, when connections from different hosts are going through the servers and eventually concentrating into one single server. Many studies have assumed this kind of topology [KUR92, RAH95]. 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 5.

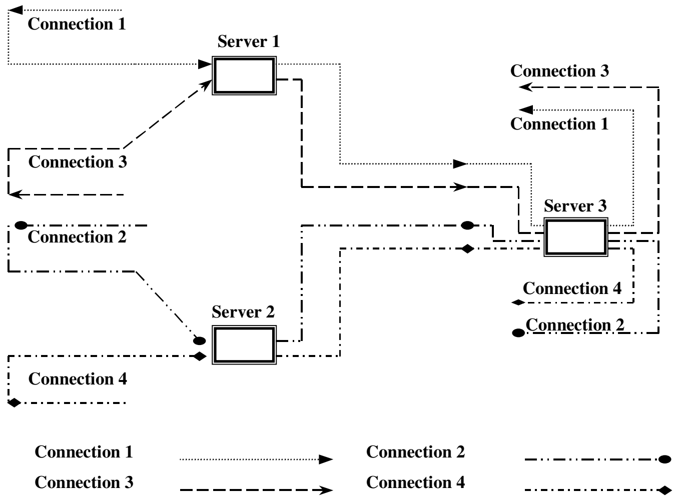

Figure 4. Connection-server graph.

Figure 4 already 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.

2.5 Traffic Descriptions

A 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. See the survey paper [ZHA95a, ZHA95b] 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 connection M_i to the k-th layer server during time interval [0,t).

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

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

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

In this paper, we assume that the initial arrival curve $F_{i,1}(I)$, i = 1, …, m, belongs to (β, ρ) model [CRU91]. That is,

$$F_{i,1}(I) = \begin{cases} I & 0 \le I \le I_i \\ \beta_i + I * \rho_i & I_i \le I \end{cases} \qquad (2.2)$$

where

$$I_i = \frac{\beta_i}{1-\rho_i}. \qquad (2.3)$$

3. Integrated Delay Analysis

In this section, we derive the end-to-end worst case delays by using integrated analysis methods. The first method is based on guaranteed service curve [CRU95, AGR96, BOU97]. The second method is based on the system equivalency.

3.1 Notations

We need to introduce some notations that will be used in the rest of this section.

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

3.2 Integrated Delay Analysis by Guaranteed Service Curve

In this sub-section, we will derive an upper bound on end-to-end delay based on guaranteed service curve. This method was the first integrated method, introduced in [CRU95]. Various issues related to this method have been discussed in [SAR95, SAR96, AGR96, BOU97]. Our contribution here is to derive a closed-form formula of the delay bound. Using a closed-form formula allows us to analytically compare the performances of different methods. Closed-form formula also simplifies the connection admission control procedure in practice.

We will first formally introduce the definition of guaranteed service curves [CRU95, SAR96, AGR96, BOU97]. As pointed out by many researchers, the guaranteed service curve 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 as the *service curve guaranteed* by the k-th layer server to connection M_i if for any t, there exists s, $0 \le s \le t$, such that

(3.1) $$R_{i,k+1}(t) - R_{i,k}(s) \ge C_{i,k}(t-s).$$

The central 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 as a *guaranteed network service curve* to connection M_i, if for all t, there exists s, $0 \le s \le t$, such that

(3.2) $$R_{i,N+1}(t) - R_{i,1}(s) \ge C_i^{End-to-end}(t-s).$$

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:

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

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

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

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

Proof. See Appendix A.

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

$$C_i^{End-to-end}(I) = \begin{cases} 0 & 0 \le I \le \sum_{k=1}^{N} \theta_{i,k} \\ (1-\pi_{i,N})(I - \sum_{k=1}^{N} \theta_{i,k}) & I > \sum_{k=1}^{N} \theta_{i,k} \end{cases} \tag{3.6}$$

where $\theta_{i,k}$ *and* $\pi_{i,N}$ are defined in (3.4) and (3.5), respectively.

Sketch of the Proof. According to [CRU95], 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, …, N, the network guarantees a network service curve of $C_i^{End-to-end}(I)$ to connection M_i, where

$$C_i^{End-to-end}(I) = \min\{\sum_{k=1}^{N} C_{i,k}(x_k) : x_k \ge 0, and \sum_{k=1}^{N} x_k = I\}. \tag{3.7}$$

Substituting (3.3) into (3.7) with certain algebraic manipulations, we then can derive (3.6). For the details of the proof, see Appendix A.

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

$$d_i^{End-to-end} \le \sum_{k=1}^{N} \theta_{i,k} + I_i \frac{\pi_{i,N}}{1-\pi_{i,N}}, \tag{3.8}$$

where $I_i, \pi_{i,N},$ *and* $\theta_{i,k}$ are in (2.3), (3.4), and (3.5), respectively.

Sketch of the Proof. According to Theorem 2.7 in [SAR96], the worst case end-to-end delay bound satisfies the following inequality

$$d_i^{End-to-end} \le \max_{c \ge 0}\{C_i^{-1\,End-to-end}(c) - F_{i,1}^{-1}(c)\}. \tag{3.9}$$

Substituting (2.2) and (3.6) into (3.9), we can derive (3.8). For the details of the proof, see Appendix A.

3.3 Integrated Delay Analysis by System Equivalency

In this section, we will derive end-to-end delay bound by using equivalent systems.

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

In this section, we will 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.

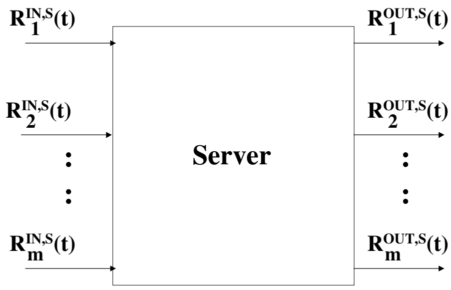

Figure 5. A single server system.

For connection M_i in a multi-server system, we denote $R_i^{IN,M}(t)$ 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. That is, $R_{i,1}(t) = R_i^{IN,M}(t)$ and $R_{i,N+1}(t) = R_i^{OUT,M}(t)$. For a single server system as shown in Figure 5, let $R_i^{IN,S}(t)$ and $R_i^{OUT,S}(t)$ be the corresponding functions, respectively.

To prove the equivalency of multi-server and single server systems, we need to establish the relationship between their input and output traffic. The following lemmas provide the needed results.

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

(3.10)
$$\begin{aligned} R_i^{OUT,S}(t) &= \inf_{s \le t}\{[(t-s) - \sum_{j=1}^{i-1}(R_j^{OUT,S}(t) - R_j^{OUT,S}(s))]^+ + R_i^{IN,S}(s)\} \\ &= [(t-s^{'}) - \sum_{j=1}^{i-1}(R_j^{OUT,S}(t) - R_j^{OUT,S}(s^{'}))]^+ + R_i^{IN,S}(s^{'}). \end{aligned}$$

where $[x]^+ = \max\{0, x\}$ and $s^{'}$ is the starting point of the $M_i^{'}s$ maximum busy period of that 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 [AGR96]. Here we explain its physical meaning. Because $s^{'}$ is the start point of the maximum busy interval, at time $s^{'}$ all arrival traffic of connection M_i is transmitted. The amount of this part of output traffic is $R_i^{IN,S}(s^{'})$. Starting from $s^{'}$, the server will serve connections with high priority first. The leftover time is given to serve connection M_i. No connection with priority lower than i is served. Hence, the amount of M_i traffic served by the server is given by $[(t-s^{'})-\sum_{j=1}^{i-1}(R_j^{OUT,S}(t)-R_j^{OUT,S}(s^{'}))]^{+}$. This explains why $R_i^{OUT,S}(t)=[(t-s^{'})-\sum_{j=1}^{i-1}(R_j^{OUT,S}(t)-R_j^{OUT,S}(s^{'}))]^{+}+R_i^{IN,S}(s^{'})$.

For a multi-server system, we can also establish the relationship between its input and output traffic of connections as follows.

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

(3.11)
$$\begin{aligned} R_i^{OUT,M}(t) &= \inf_{s\le t}\{[(t-s)-\sum_{j=1}^{i-1}(R_j^{OUT,M}(t)-R_j^{OUT,M}(s))]^{+}+R_i^{N-1}(s)\} \\ &= [(t-s^{''})-\sum_{j=1}^{i-1}(R_j^{OUT,M}(t)-R_j^{OUT,M}(s^{''}))]^{+}+R_i^{IN,M}(s^{''}). \end{aligned}$$

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

Proof. See Appendix A.

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. That is, if, for i = 1, 2, …, m, $R_i^{IN,S}(t)=R_i^{IN,M}(t)$, then $R_i^{OUT,S}(t)=R_i^{OUT,M}(t)$.

Proof. See Appendix A.

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

(3.12)
$$d_i^{End-to-end}\le\theta_{i,N}+I_i\frac{\pi_{i,N}}{1-\pi_{i,N}}$$

where $I_i, \pi_{i,N}$, *and* $\theta_{i,k}$ are defined in (2.3), (3.4), and (3.5), respectively.

Proof. This theorem follows by directly using Theorem 1 for the single server system.

4. Performance Comparison

In this section, we first compare the closed-form of 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.

4.1. Comparison of Delay Formulae

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

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

Table 1 shows the delay formulae of 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. Comparing between the decomposed method and the Guaranteed Service Curve (GSC) method, as $\theta'_{i,k} \geq \theta_{i,k}$ and the second term in both formulae differs by $\sum_{k=1}^{N-1} I_{i,k} \frac{\pi_{i,k}}{1-\pi_{i,k}}$, we see that the GSC method produces a smaller delay bound than the decomposed method. Similarly, comparing between 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 brief, our newly proposed integrated method is the best among the three methods that have been proposed and analyzed so far.

4.2. Performance Metric

We would like to evaluate some statistical performance of the network systems with different delay derivation methods. The performance metric we are interested in is *admission probability*. By admission probability, we meant 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.

4.3. Simulation Experiments

We developed a discrete event simulation model to simulate the ATM network. The simulation program is written in the C++ programming language and run in a SUN/Solaris environment. There are altogether five system parameters for this simulation: λ is defined as 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 lastly, 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, we obtained our estimated utilization of the server at the N-th layer as $U = \lambda\, \rho/\mu$. All our simulation experiments reported here are based on a network with 4 layers and 15 switches.

For the priority assignment, we use the Deadline Monotonic policy. That is, the priorities of connections are assigned in accordance to their relative deadlines. The smaller the deadline of a connection, the higher its priority.

For each run of our simulation experiments, we generate 5,000 connections according to the predefined system parameters, and estimate the admission probability, P_{adm} as follows:

(4.1)
$$P_{adm} = \frac{N_{accept}}{\left(N_{accept} + N_{reject}\right)}$$

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

We started the simulation with no connection in the network initially. To eliminating the warm-up effect, we let the system to run into a 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 connection.

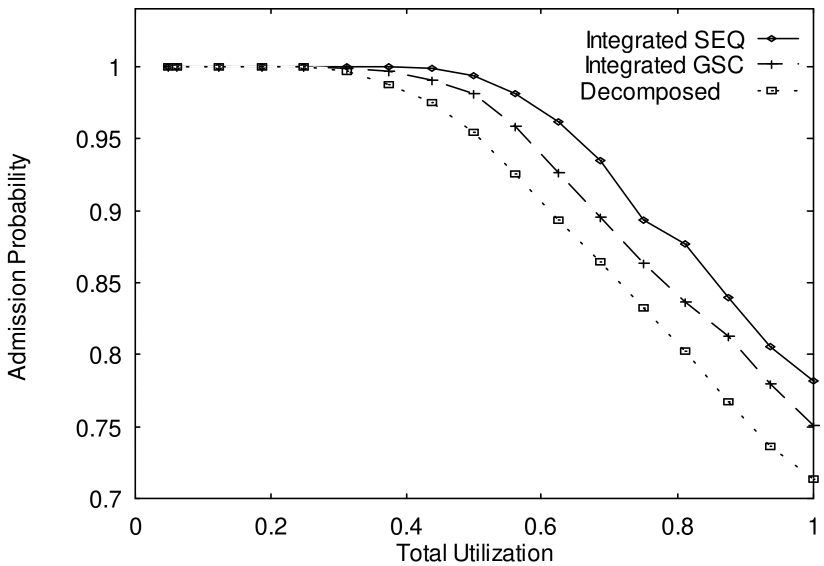

Figure 6. Admission probability vs. total utilization.

4.4. Simulation Results

Figures 6 shows the admission probability for the network systems when using different delay derivation methods. All three methods show a typical trend 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 (~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.

Figure 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 Figure 7, we see that with very tight deadlines, the admission probability is low. It is 0.89 for our SEQ method, 0.86 for GSC, and 0,81 for the decomposed method. As we assigned larger deadline to the connections, the admission probability for all three methods increases. 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.

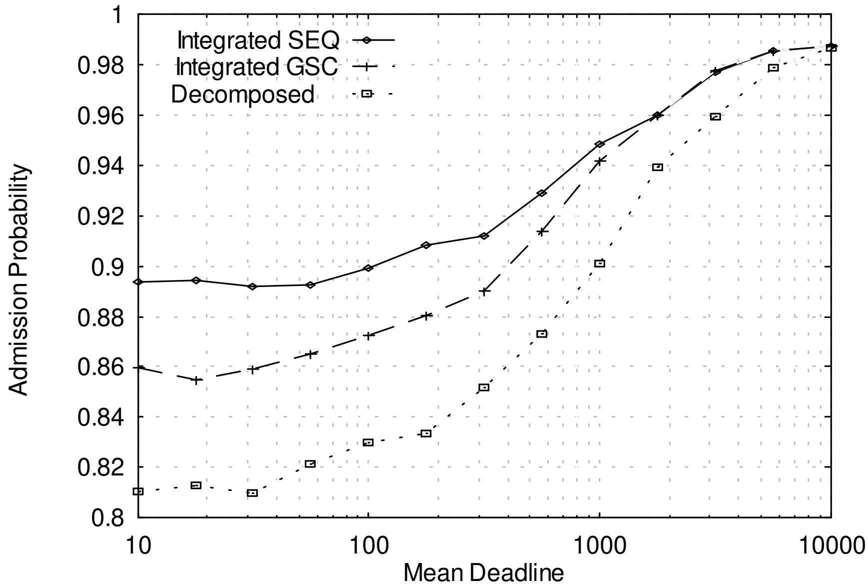

Figure 7. Sensitivity of deadlines (Utilization 75%).

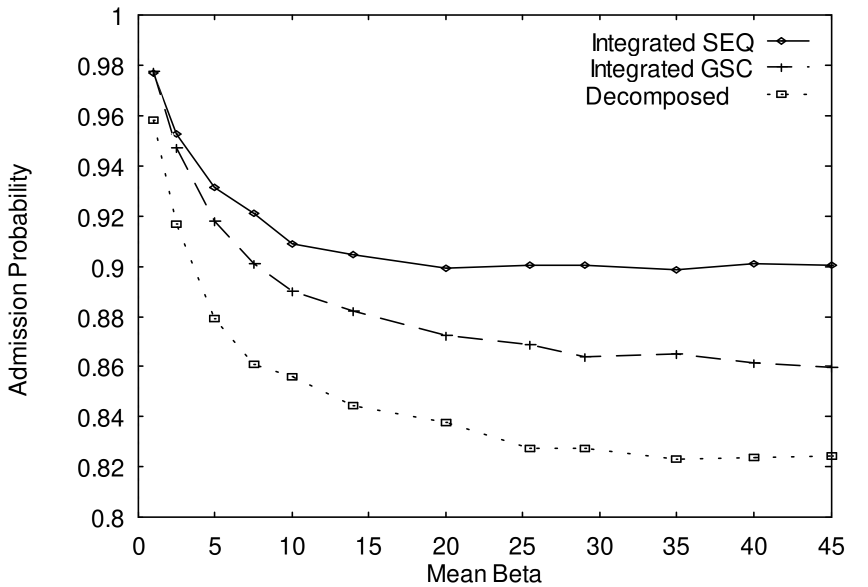

Figure 8. Sensitivity of burstiness (Utilization 75%).

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 Figure 8. With less bursty traffic, the admission probability for all three methods is higher. As burstiness increases, all three methods experience performance drops in term 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 at large β settings.

5. Summary and Future Work

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 that our new method performs the best in comparison to other known methods. This is observed through both analytical comparisons and extensive simulation experiments.

The work reported here 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. This is a homogeneous priority assignment scheme. We, therefore, are also interested in working on the integrated end-to-end delay analysis under heterogeneous priority assignment schemes.

6. References

[AGR96] R. Agrawal and R. Rajan, "Performance bounds for guaranteed and adaptive service," *IBM Research Report,* May, 1996.

[BHZ96] J. Bennett and H. Zhang, "WF^2Q: Worst-case fair weighted fair queuing", *Proc. of INFOCOM*, 1996.

[BOU97] J. Boudec, "Application of network calculus to guaranteed service networks," Accepted for *IEEE Trans. Inform. Theory,* 1998.

[CHA94] C. S. Chang, "Stability queue length, and delay of deterministic and stochastic queuing networks," *IEEE Trans. Automat. Contr.,* vol. 39, no.5, May 1994.

[CRU91]R. L. Cruz, "A calculus for network delay, Part I & II," *IEEE Trans. Inform. Theory,* vol.37, no.1, Jan. 1991.

[CRU92] R. L. Cruz, "Service burstiness and dynamic burstiness measures: A framework," *J. High Speed Networks,* vol. 1, no.2, 1992.

[CRU93] 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, March 1993.

[CRU94] R. L. Cruz and H. Liu, "End-to-end queuing delay in ATM networks," *J. High Speed Networks,* vol.3, no.4, 1994.

[CRU95] R. L. Cruz, "Quality of service guarantees in virtual circuit switched networks," *IEEE. J. Select.Areas Commun.,* vol.13, no.6, 1995.

[FER90] 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.

[GEO96] L. Georgiadis, R. Guerin, V. Peris and K. N. Sivarajan, "Efficient network QoS provisioning based on per node traffic shaping," *Proc. of IEEE INFOCOM*,1996.

[HLU88] M. G. Hluchyj and M. J. Karol, " Queueing in high performance packet switching, " *IEEE J. Select. Areas Commun.,* vol. 6, no. 9, Sept. 1988.

[HAN95a] C. Han, C. Hou, and K. Shin, "On slot allocation for time-constrained messages in DQDB networks," *Proc. of IEEE INFOCOM*, 1995.

[HAN95b] C. Han and K. Shin, "Real-time communication in FieldBus multiaccess networks," *Proc. of IEEE RTAS,* 1995.

[HAN97] C. Han, C. Hou, and K. Shin, "On slot allocation for time-constrained messages in dual-bus networks," to appear in *IEEE Trans. on Computers*, July 1997.

[KNI95] E. Knightly and H. Zhang, "Traffic characterization and switch utilization using deterministic bounding interval dependent traffic models," *Proc. of IEEE INFOCOM,* 1995.

[KUR92] J. Kurose, "On computing per-session performance bounds in high-speed multi-hop computer networks," in *Proc. of ACM Sigmetrics, Perform.,* 1992.

[LEE96] C. Lee, K. Yoshida, C. Mercer and R. Rajkumar, "Predictable Communication Protocol Processing in Real-Time Mach", *Proc. of RTAS*, June 1996.

[LIZ97] C. Li, R. Bettati and W. Zhao, "Delay computation and priority assignment for ATM networks with static priority scheduling," *Proc. of RTSS*, 1997.

[LIE96] 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.

[MUK93] Mukherjee, Tripathi, Saha, and Saksena, "A Bandwidth Allocation Scheme for Time Constrained Message Transmission on A Slotted Ring LAN," *Proc. of RTSS*, 1993.

[NEW92] P. Newman, "ATM technology for corporate networks," *IEEE Comm. Magazine*, Apr. 1992.

[NGS97] J. Ng, S. Song, and W. Zhao, "Integrated Delay Analysis of Regulated ATM Switch," *Proc. of RTSS*, 1997.

[PAR93] 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.

[PAR94] 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.

[RAH95] A. Raha, S. Kamat, and W. Zhao, "Guaranteeing end-to-end deadlines in ATM networks," *Proc. of ICDCS,* 1995.

[RAH96] A. Raha, S. Kamat, and W. Zhao, "Admission control for hard real-time connections in ATM networks," in *Proceedings of IEEE INFOCOM*, Mar. 1996.
[ROO94] R. Rooholamini, V. Cherkassky and M. Garver, "Finding the right ATM switch for the market," *IEEE Computer,* vol. 27, no. 4, Apr. 1994.
[SAH94] Saha, Tripathi, Saksena, and Mukherjee, "On Guaranteed Delivery of Time-Critical Messages in DQDB", *Proc. of IEEE INFOCOM* , 1994.
[SAR95] H. Sariowan, L. Cruz and G. Polyzos, "Scheduling for quality of service guarantees via service curves," *Proc. of ICCCN,* 1995.
[SAR96] H. Sariowan. " A service-curve approach to performance guarantees in integrated-service networks". Ph. D disertation. University of California, San Diego, 1996.
[WRE96] 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.
[WRE97] D. E. Wrege and J. Liebeherr, "A near-optimal packet scheduler for QoS networks," *Proc. of IEEE INFOCOM,* 1997.
[YAT93] 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.
[ZHA94] H. Zhang and D. Ferrari, "Rate-controlled service disciplines," *J. High Speed Networks,* vol. 3, no. 4, 1994.
[ZHA95a] H. Zhang, "Service disciplines for guaranteed performance service in packet switching networks", Proc. IEEE, 1995.
[ZHA95b] H. Zhang, "Providing end-to-end performance guarantees using non-work-conserving disciplines", Computer Communications: Special Issue on System Support for Multimedia Computing, 18(10), 1995.
[ZHE96] Q. Zheng and K. G. Shin, "On the ability of establishing real-time channels in point-to-point packet-switched networks, " *IEEE Trans. Communications*, 1996.

Appendix A: Proofs of Lemmas and Theorems

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

(3.3)
$$C_{i,k}(I) = \begin{cases} 0 & 0 \le I \le \theta_{i,k} \\ (1-\pi_{i,k})(I-\theta_{i,k}) & I > \theta_{i,k} \end{cases}$$

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

(3.4)
$$\pi_{i,k} = \sum_{j \in G_{i,k}} \rho_j \,,$$

and

(3.5)
$$\theta_{i,k} = \frac{\sum_{j \in G_{i,k}} \beta_j}{1-\pi_{i,k}} .$$

Proof: For any time t > 0, according to [ARG96], we have

(A.1)
$$R_{i,k+1}(t) = [(t-s) - \sum_{j=1}^{i-1} (R_{j,k+1}(t) - R_{j,k+1}(s))]^+ + R_{i,k}(s)$$

where s is the starting point of $M_i^{'}s$ maximum busy period 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_i at the j-th layer server for j = 1, 2, …, 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., we have $R_{j,k+1}(s) = R_{j,1}(s)$. On the other hand, $R_{j,k+1}(t) \le R_{j,1}(t)$. Therefore, we have, for j = 1, 2, …, i-1,

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

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

(A.3)
$$\begin{aligned} R_{i,k+1}(t) - R_{i,k}(s) &= [(t-s) - \sum_{j=1}^{i-1} (R_{j,k+1}(t) - R_{j,k+1}(s))]^+ \\ &\ge [(t-s) - \sum_{j=1}^{i-1} (R_{j,1}(t) - R_{j,1}(s))]^+ \\ &\ge [(t-s) - \sum_{j=1}^{i-1} F_{j,1}(t-s)]^+ . \end{aligned}$$

Thus, by Definition 5,

(A.3') $$C_{i,k}(I) = [I - \sum_{j=1}^{i-1} F_{j,1}(I)]^{+}$$

is a service curve guaranteed by the k-th layer server to connection M_i. Finally, substituting (2.2) into (A.3') with certain algebraic manipulations, we have

(A.4) $$C_{i,k}(I) = \begin{cases} 0 & 0 \le I \le \theta_{i,k} \\ (1-\pi_{i,k})(I-\theta_{i,k}) & I > \theta_{i,k} \end{cases}.$$

Q.E.D.

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

(3.6) $$C_i^{End-to-end}(I) = \begin{cases} 0 & 0 \le I \le \sum_{k=1}^{N} \theta_{i,k} \\ (1-\pi_{i,N})(I - \sum_{k=1}^{N} \theta_{i,k}) & I > \sum_{k=1}^{N} \theta_{i,k} \end{cases}$$

where $\theta_{i,k}$ *and* $\pi_{i,N}$ are defined in (3.4) and (3.5), respectively.

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

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

According to Theorem 5 in [CRU95], we have

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

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

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

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

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

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

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

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

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

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

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

This means

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

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

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

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

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

Q.E.D.

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

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

where $I_i, \pi_{i,N}, and\ \theta_{i,k}$ are in (2.3), (3.4), and (3.5), respectively.

Proof. According to Theorem 2.7 in [SAR96], the worst case end-to-end delay bound satisfies the following inequality

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

According to (2.2) and (3.6), we have

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

and

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

So,

(A.17) $$C_i^{-1\,End-to-end}(c)-F_{i,1}^{-1}(c)=\begin{cases}\sum_{k=1}^{N}\theta_{i,k}+\dfrac{\pi_{i,N}}{1-\pi_{i,N}}*c & 0\le c<I_i\\ \sum_{k=1}^{N}\theta_{i,k}+\dfrac{\beta_i}{\rho_i}+(\dfrac{1}{1-\pi_{i,N}}-\dfrac{1}{\rho_i})*c & I_i\le c\end{cases}.$$

Since we assume that $\sum_{k=1}^{m}\rho_k\le 1$, so $1-\pi_{i,N}\ge\rho_i$. Therefore, we have that $\frac{1}{1-\pi_{i,N}}-\frac{1}{\rho_i}\le 0$. 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

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

(3.8) then follows from (A.14) and (A.18).

Q.E.D.

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

(3.11) $$\begin{aligned}R_i^{OUT,M}(t)&=\inf_{s\le t}\{[(t-s)-\sum_{j=1}^{i-1}(R_j^{OUT,M}(t)-R_j^{OUT,M}(s))]^{+}+R_i^{N-1}(s)\}\\&=[(t-s^{"})-\sum_{j=1}^{i-1}(R_j^{OUT,M}(t)-R_j^{OUT,M}(s^{"}))]^{+}+R_i^{IN,M}(s^{"}).\end{aligned}$$

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

Proof: According to [ARG96], for any t > 0,

(A.19) $$\begin{aligned}R_i^{OUT,M}(t)&=\inf_{s\le t}\{[(t-s)-\sum_{j=1}^{i-1}(R_j^{OUT,M}(t)-R_j^{OUT,M}(s))]^{+}+R_i^{N-1}(s)\}\\&=[(t-s^{"})-\sum_{j=1}^{i-1}(R_j^{OUT,M}(t)-R_j^{OUT,M}(s^{"}))]^{+}+R_i^{N-1}(s^{"}),\end{aligned}$$

where $s^{"}$ is the starting point of the connection $M_i^{'}s$ maximum busy period, 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 M_i at the j-th layer server for j = 1, 2, …, 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., we have $R_i^{IN,M}(s^{"}) = R_i^{N-1}(s^{"})$. Therefore, we have

(A.20)
$$\begin{aligned} R_i^{OUT,M}(t) &= \inf_{s \le t}\{[(t-s) - \sum_{j=1}^{i-1}(R_j^{OUT,M}(t) - R_j^{OUT,M}(s))]^+ + R_i^{N-1}(s)\} \\ &= [(t-s^{"}) - \sum_{j=1}^{i-1}(R_j^{OUT,M}(t) - R_j^{OUT,M}(s^{"}))]^+ + R_i^{IN,M}(s^{"}). \end{aligned}$$

Q.E.D.

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. That is, for i = 1, 2, …, m, if $R_i^{IN,S}(t) = R_i^{IN,M}(t)$, then $R_i^{OUT,S}(t) = R_i^{OUT,M}(t)$.

Proof: We prove this theorem by induction.

For n = 1. In the single server priority driven system, by Lemma 3, the output traffic function of connection M_1 is

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

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

In the multi-server priority driven tree system, by Lemma 4, the output traffic function of connection M_1 is

(A.22)
$$R_1^{OUT,M}(t) = (t-s") + R_{1,N}(s"),$$

where $s^{"}$ is the starting point of connection $M_i^{'}s$ maximum busy period, 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

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

For n > 1, suppose that for 1 ≤ i < n , $R_i^{OUT,S}(t) = R_i^{OUT,M}(t)$ 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

(A.24) $$R_n^{OUT,M}(t) = [(t - s'') - \sum_{j=1}^{n-1}(R_j^{OUT,M}(t) - R_j^{OUT,M}(s''))]^+ + R_n^{IN,M}(s''),$$

and

(A.25) $$R_n^{OUT,S}(t) = [(t - s') - \sum_{j=1}^{n-1}(R_j^{OUT,S}(t) - R_j^{OUT,S}(s'))]^+ + R_n^{IN,S}(s').$$

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. So connection $M_n^{'}s$ maximum busy periods, 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

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

Q.E.D.

Appendix B: Delay analysis by the Decomposed Method.

The decomposed method was first proposed in [CRU91] 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 R. Cruz in [CRU91]. It pointed out that the shape of arrival curve of a connection inside network keeps the same except that the burst parameter increases with the number of hops visited by the connection increasing.

Lemma 5. For connection M_i, if the worst case delay $d_{i,j}$, j = 1, 2,…, 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:

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

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

(B.2) $$\beta_{i,1} = \beta_i,$$

(B.3) $$\beta_{i,k} = \beta_{i,k-1} + \rho_i * d_{i,k-1},$$

and

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

Proof. See Lemma 1 in [CRU91] and Theorem 6 in [LIZ97].

After given the traffic description for every connection at each server, 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

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

where $\pi_{i,k}$ is defined in (3.4), $I_{i,k}$ is defined in (B.4), and $\theta'_{i,k}$ defined as follows:

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

Proof. At the k-th layer server, the input traffic of connection M_i is constraint 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

$$C_{i,k}(I) = \max\{0, I - \sum_{j=1}^{i-1} F_{i,j}(I)\}$$

(B.7)
$$= \begin{cases} 0 & 0 \le I \le \theta'_{i,k}, \\ (1-\pi_{i,k})(I-\theta'_{i,k}) & \theta'_{i,k} \le I \end{cases} .$$

According to Theorem 2.7 in [SAR96], we have

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

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

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

Q.E.D.

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

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

where $\theta'_{i,k}$, $I_{i,k}$, *and* $\pi_{i,k}$ are defined as in (B.6), (B.4) and (3.4) respectively.