Connection-Oriented Communications for Real-Time Applications in FDDI-ATM-FDDI Heterogeneous Networks

Biao Chen[*], Anirudha Sahoo,[†] Wei Zhao[†], and Amitava Raha[‡]

Abstract

In this paper, we study connection-oriented service in an FDDI-ATM-FDDI heterogeneous network for real-time applications. We design and analyze an algorithm for connection admission control (CAC) for such a network. Upon a request of connection establishment, the CAC determines if the worst case delays of the requesting and existing connections can be satisfied given the available network resources. If so, the CAC allocates appropriate network resources to the requesting connection. The process of allocating resources for homogeneous networks (e.g. FDDI-only or ATM-only) may not be applied directly to a heterogeneous network environment(e.g. FDDI-ATM-FDDI network) because heterogeneity adds more complexity to the process. Hence resource allocation in a heterogeneous network needs more careful analysis than its homogeneous counterpart. In this paper, we propose a CAC algorithm that will, by proper parameter tuning, allocate sufficient but not excessive network resources to the requesting connection in an FDDI-ATM-FDDI network. We show that the system can achieve satisfactory performance with this CAC algorithm. Our approach is compatible with current network standards and hence can be readily used in practical systems.

Keywords: *Heterogeneous Networks, Real-Time Communications, Connection Admission Control.*

1 Introduction

Real-time communications have been studied since 1980s. Issues pertaining to meeting message deadlines have been addressed for CSMA/CD [9], token ring [19], FDDI [1, 11], slotted ring [13], DQDB [18], and ATM and point-to-point networks [12]. Majority of these studies addressed the problem for homogeneous communication networks. However, most of the existing communication networks are heterogeneous: usually consisting of a wide variety of homogeneous sub-network segments. In this paper, we study the real-time communication problem in *heterogeneous networks*. In particular, we will concentrate on FDDI-ATM-FDDI heterogeneous networks where ATM serves as a backbone that connects FDDI LAN segments by interface devices.

In this paper, we will study connection-oriented service over an FDDI-ATM-FDDI heterogeneous network. The method of allocating resources in an FDDI-only or ATM-only LAN may not be applied directly to FDDI-ATM-FDDI network. This is due to the fact that a connection in a heterogeneous network is more complex and allocation of resources cannot be made by analyzing each homogeneous segment in an isolated manner. Resorting to an efficient method of resource allocation for one LAN segment may be good for that LAN segment, but may adversely affect the performance of other LAN segments in the heterogeneous network. So an integrated resource allocation scheme should be followed in a heterogeneous environment so that the performance of the entire heterogeneous network is optimized. In this paper, we have made an attempt to devise such an approach for an FDDI-ATM-FDDI heterogeneous network.

The key problem for connection-oriented services over an FDDI-ATM-FDDI network is the admission control of connections. The procedure of connection admission control can be divided into two steps:

Step 1: to determine if the worst case delay of the requesting and existing connections can be satisfied given the available network resources.

Step 2: if the delay constraints can be satisfied, to allocate a *proper* amount of network resource and to establish the connection.

In Step 1, the worst case end-to-end packet delay has to be analyzed. For this purpose, we take a decomposition approach in which a connection path is decomposed into a sequence of servers; then the worst case end-to-end delay is obtained by summing up the worst case delays suffered by the connection at individual servers. Previous studies have shown that such an approach is efficient and effective, if proper traffic specification is provided. Similar approach was also used in the design of Tenet protocol suite [8].

The second step is an interesting one. Allocation of network resources to the requesting connection impacts the admission of future connections. If too much resource were allocated to the requesting connection, a future connection request might be rejected due to insufficient available resource. On the other hand, if too little resource were allocated to the requesting connection, its worst case delay would be very "tight" (i.e., very close to its deadline). Consequently, disturbance introduced by future connections may likely result in the violation of its deadline constraint. Thus, the new connection will be rejected. Hence, excessive or insufficient allocation of resources to a connection may jeopardize the chance of a future connection being admitted.

---

[*] Department of Computer Science, University of Texas at Dallas, Richardson, TX 75083-0688, email: bchen@utdallas.edu

[†] Department of Computer Science, Texas A&M University, College Station, Texas 77843-3112, e-mail: {asahoo, zhao}@cs.tamu.edu

[‡] Fujitsu Software Co., 3055 Orchard Dr., San Jose, CA 95134, email: amitava@fsc.fujitsu.com

We address this problem in our CAC algorithm. Once our CAC algorithm determines that a requesting connection is admissible, it will allocate β percent resource between the minimum amount that is absolutely necessary and the maximum amount that is needed. β is an user adjustable parameter. We study the relationship between system performance and the chosen value of β. We found that the system is relatively stable in the sense that for a wide range of β values, the system performs very well.

2 Related Work

Considerable amount of work has been done in the design and analysis of networks for supporting real-time systems. Generally speaking, determining delay bounds has been the pivotal issue in the development of real-time networking technology. Previous work concentrate on designing and analyzing homogeneous communication networks in which the worst case delays can be bounded [1, 4, 7, 12, 13, 14, 15, 18, 20]. Both shared media networks (e.g., 802.5, FDDI, and DQDB) and point-to-point networks (e.g., ATM) have been studied for real-time communications over homogeneous networks.

Worst case delays were derived for 802.5 token ring networks [20], slotted ring networks [13], DQDB networks [18], and FDDI networks [1, 4]. Buffer space was considered in [11] for FDDI networks. For connection-oriented networks, much of the previous work concentrates on obtaining the delay bounds and connection admission criteria for individual scheduling policies. Ferrari and Verma [7] and Zheng and Shin [25] studied the use of Earliest Deadline First scheduling in wide area networks. Zhang and Ferrari [23] discussed 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 [22] and by Cruz in [6]. In [2, 5], the decomposition methodology has been used to compute the delays in a connection-oriented packet-switched networks. In [14], we decomposed an ATM network and analyzed its basic servers and introduce a general traffic descriptor to describe traffic in ATM networks.

3 Network and Connection Models

In this section, we will first describe a generic ATM-based heterogeneous network (ABHN) model and its operations, which we will use in the anlysis of our FDDI-ATM-FDDI heterogeneous network. We will then formally define the real-time connection.

3.1 ABHN Architecture

ATM-Based Heterogeneous Network (ABHN) is a high performance and scalable network architecture. It was proposed recently to meet the performance and management challenges in heterogeneous networks. This model has been accepted by many industries as the platform to migrate from router-based to switch-based heterogeneous networks. Figure 1 shows the architecture of a typical ABHN. There are four types of major components in the network: 1) Hosts, 2) Legacy local area network segments, 3) Interface devices and 4) ATM backbone.

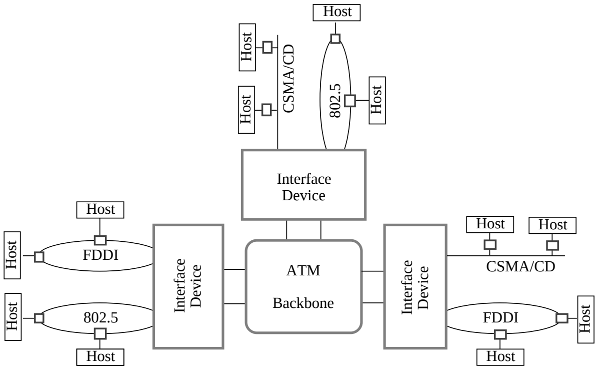


Figure 1: An ATM-Based Heterogeneous Network

Hosts are connected to legacy LAN segments which can be almost any type of LANs including CSMA/CD, 802.5 token ring, and FDDI. Therefore, the applications developed over legacy LANs can be migrated to ABHN without any change. We assume that there are L LAN segments in the network and segment i has L_i hosts. We denote the j-th host on segment i by $\text{Host}_{i,j}$.

The most important components in ABHN are the interface devices. An interface device serves as an interface between the legacy LAN segments and the ATM backbone. It enables LAN segments to be interconnected through the ATM backbone. An interface device is responsible for traffic mapping between legacy LAN and ATM. That is, for a data frame from a LAN segment, if the destination LAN segment is connected on the same interface device, then the data frame will be converted into the right format and forwarded to the destination LAN. Otherwise, the interface device will switch an incoming LAN frame to appropriate output port, where the frame is converted to ATM cells. These ATM cells are then transmitted over the link into the ATM backbone. On the other hand, the cells, arriving from ATM backbone, are assembled back to a LAN frame and switched to the proper output port to which the destination LAN segment is connected. The frame is then transmitted over the destination LAN.

ATM is chosen as the backbone to interconnect all the legacy LANs because of its high bandwidth, scalability, flexibility, and support of connection-oriented communication. An ATM network consists of a collection of switches interconnected by physical links. In ATM networks, the data are packetized into fixed size packets called cells. As cells belonging to different connections travel in the network, they may share network resources such as communication links. It is the task of ATM switches to multiplex cells of different connections onto shared links.

For our FDDI-ATM-FDDI network, all the LAN segments will be FDDI rings. FDDI is a fiber optical token ring. It has been widely used in real-time systems. Its timed token media access control protocol is designed to support real-time communication. On the FDDI ring, an FDDI station is assigned a synchronous bandwidth, H. Every time a station receives the token, it is allowed to transmit its real-time data up to a time period of length H. Hence, data packets on a station are split into frames of size $F_S = H \cdot \mathrm{BW}_{\mathrm{FDDI}}$, where H is synchronous bandwidth allocated and $\mathrm{BW}_{\mathrm{FDDI}}$ is the maximum bandwidth of FDDI ring. The FDDI protocol requires that the summation of all the synchronous bandwidth allocated to all the stations does not exceed TTRT, the target token rotation time. In Section 7, we will discuss how to extend our method to other types of LAN segments, e.g., 802.5 token ring.

3.2 Real-Time Connections

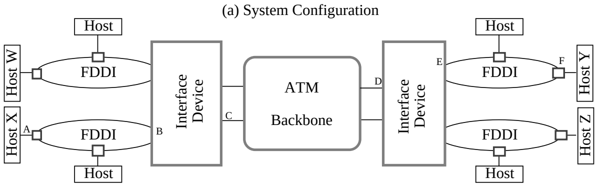
A connection is a relationship between the application and the network, which can be viewed in terms of a contract: the communicating application specifies the characteristics of the traffic which it may generate (say, the maximum rates) and the network agrees to provide the requested quality-of-service to the application. The network will not admit a connection if the requested quality-of-service cannot be guaranteed. Specifically, this paper we specify a connection for real-time application by the following parameters:

- *Source traffic specification.* This defines the traffic behavior of the source. Traffic specification plays an important role in guaranteeing deadlines for real-time connections. We will discuss in detail about our method of traffic specification in the next section.
- *Quality-of-service requirement.* For a connection for real-time application, its QoS requirement is that the worst case end-to-end delay of its packets should be no more than its deadline (D). Note that if there is a buffer overflow, the data will be lost and hence the delay of a packet will be infinite. That is, the delay requirement also demands that there should be no buffer overflow in the network.
- *Route.* This is the path from the sender's host to the receiver's host via LAN segments, interface devices, and a path in the ATM backbone. Routing in digital network for real-time applications is an interesting and challenge problem. Many studies have been reported [10]. We will adopt these solutions without further addressing this problem here.

To simplify the discussion, we assume that there is at most one connection originating from a host. In [1], it is shown that a system with more than one connection per host is logically equivalent to the one in which there is at most one connection per host. Hence, this assumption is made without loss of generality. The connection that originates from $\mathrm{Host}_{i,j}$ is identified as $M_{i,j}$. Its deadline is denoted as $D_{i,j}$.

As stated earlier, the objective of this study is to develop a connection admission control scheme which determines whether or not the QoS requirements of a new connection can be satisfied while continuing to provide the QoS of other connections. A key step in this process is to derive the worst case delay of a packet from its source to destination in the FDDI-ATM-FDDI network. We address this problem in the next section.

(a) System Configuration



(b) Decomposition of the Connection Path

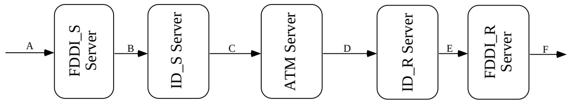


Figure 2: An Example

4 Delay Analysis

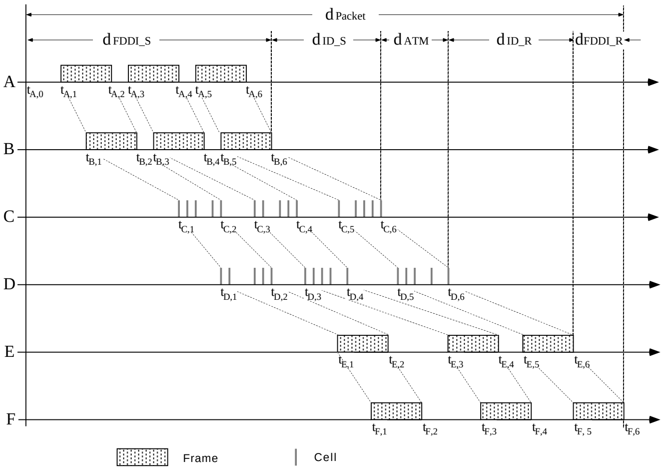
To obtain the worst case end-to-end delay, we adopt a decomposition approach in which the path of a connection is decomposed into a sequence of servers; then the worst case end-to-end delay is obtained by summing up the worst case delay suffered by the connection at each server. A general description of the approach is given in Section 4.1. This approach relies on 1) accurate traffic description of connections at the source as well as inside FDDI-ATM-FDDI network, and 2) efficient analysis of individual servers in terms of the worst case delay and output traffic. These two issues are addressed in Sections 4.2 and 4.3, respectively.

4.1 The Decomposition Approach

A path of a connection may span various components in a heterogeneous environment. Depending on the sender and the receiver, a connection can have different routes. 1) both are on the same LAN segment, 2) they are on different LAN segments which are connected to the same interface device, 3) they are on different LAN segments which are connected to different interface devices. Obviously, the first two cases are the special cases of the third one. Hence, our study will only concentrate on the third case.

Nevertheless, we can view the route of a connection in our heterogeneous network as a sequence of servers. At the highest level, the source and destination hosts, the FDDI rings, the ATM backbone, and interface devices spanned by the connection's route, each can be modeled as a server. Consider an example shown in Figure 2 (a). A connection runs from host X to host Y. The traffic of the connection first goes through FDDI ring on the sender site (denoted as FDDI_S), an interface device on the sender site (denoted as ID_S), and then the ATM backbone. After that, it passes the second interface device (denoted as ID_R), and the second FDDI ring (denoted as FDDI_R). Figure 2 (b) illustrates how we can decompose the portion of network serving this connection into several servers. These are compound servers as they will be further decomposed.

Given such a decomposition, let us consider the packet delay of this connection. Assume that the



The symbols in the above figure are defined as follows:

A, B, C, D, E, and F are points in the network shown in Figure 2.

$t_{A,0}$ = *the arrival time of the packet at FDDI_S*

For X = A, B, ..., F, and i = 1, 2, 3,

$t_{X,2i-1}$ = *the time when the first bit of (the first cell of) the i-th frame reaches point X.*

$t_{X,2i}$ = *the time when the last bit of (the last cell of) the i-th frame reaches point X.*

Figure 3: Timing Relationship of a Packet Delay

packet is broken into three FDDI frames when transmitted over the FDDI rings. See Figure 3 for an illustration of the timing relationship at different points of the network. Let the transmission request of the packet arrive at $t_{A,0}$ and assume the last bit of the packet reaches the receiver at $t_{F,6}$. Then, the delay of the packet is given by

$$\begin{aligned} d_{\mathrm{Packet}} &= t_{\mathrm{F},6} - t_{\mathrm{A},0} \\ &= d_{\mathrm{FDDI\_S}} + d_{\mathrm{ID\_S}} + d_{\mathrm{ATM}} + d_{\mathrm{ID\_R}} + d_{\mathrm{FDDI\_R}} \quad (1) \end{aligned}$$

where $d_{\mathrm{FDDI\_S}}$, $d_{\mathrm{ID\_S}}$, d_{ATM}, $d_{\mathrm{ID\_R}}$, and $d_{\mathrm{FDDI\_R}}$ are delays encountered in compound servers FDDI_S, ID_S, ATM, ID_R, FDDI_R, respectively. They are defined as follows:

$$d_{\mathrm{FDDI\_S}} = t_{\mathrm{B},6} - t_{\mathrm{A},0} \quad (2)$$

$$d_{\mathrm{ID\_S}} = t_{\mathrm{C},6} - t_{\mathrm{B},6} \quad (3)$$

$$d_{\mathrm{ATM}} = t_{\mathrm{D},6} - t_{\mathrm{C},6} \quad (4)$$

$$d_{\mathrm{ID\_R}} = t_{\mathrm{E},6} - t_{\mathrm{D},6} \quad (5)$$

$$d_{\mathrm{FDDI\_R}} = t_{\mathrm{F},6} - t_{\mathrm{E},6} \quad (6)$$

Let $d^{wc}_{\mathrm{FDDI\_S}}$, $d^{wc}_{\mathrm{ID\_S}}$, d^{wc}_{ATM}, $d^{wc}_{\mathrm{ID\_R}}$, and $d^{wc}_{\mathrm{FDDI\_R}}$ be the worst case values (i.e., upper bounds) of $d_{\mathrm{FDDI\_S}}$, $d_{\mathrm{ID\_S}}$, d_{ATM}, $d_{\mathrm{ID\_R}}$, and $d_{\mathrm{FDDI\_R}}$, respectively. We call them the worst case delays suffered by the traffic of the connection at corresponding servers. Then, the worst case end-to-end delay of a packet is given by

$$d^{wc}_{\mathrm{Packet}} = d^{wc}_{\mathrm{FDDI\_S}} + d^{wc}_{\mathrm{ID\_S}} + d^{wc}_{\mathrm{ATM}} + d^{wc}_{\mathrm{ID\_R}} + d^{wc}_{\mathrm{FDDI\_R}}. \quad (7)$$

(7) is the expression that we will use to arrive at the worst case end-to-end delay. That is, we calculate the worst case delay suffered by a connection by summing the the delays encountered in each of the compound servers traversed by the connection path. The worst case delay in each server can be obtained by performing a delay analysis for each individual server. As we will see, during this process, a compound server may need to be further decomposed in order to obtain this worst case delay.

While this approach may seem straightforward, its success hinges on two critical issues that must be thoroughly addressed. Clearly, we cannot compute the bound of the delay suffered by a packet at a server, without information about the connection's traffic as seen by the server. Further, for a particular server, we must model its service discipline and understand its impact on the connections. Therefore, the two key issues are: traffic description of a connection and individual server analysis. We discuss these two issues in the next two subsections.

4.2 Traffic Description Method

We use the term *traffic descriptor* for a method used to provide relevant information about a connection's traffic. In this paper, we adopt a traffic descriptor called *the maximum rate function* $\Gamma(I)$ which is defined as the maximum arrival rate in any interval of length I (in bits/second).

$\Gamma(I)$ has been successfully used as a traffic descriptor in ATM networks for real-time applications [14]. A similar maximum rate function has also been used in [7]. It is clear that $\Gamma(I)$ is capable of representing the worst case traffic behavior both at the source and inside of the network. In [16, 17], a CAC algorithm based on $\Gamma(I)$ has been designed and analyzed for ATM networks. It was shown that with this traffic descriptor, the CAC algorithm can make a connection admission decision effectively and efficiently.

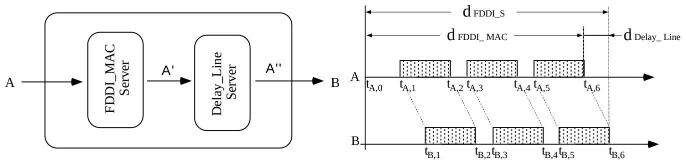
To distinguish traffic of different connections at different locations of the network, we use $\Gamma_{i,j,X}(I)$ to represent the traffic of connection $M_{i,j}$ at a particular location identified by X.

4.3 Server Analysis

The objectives of server analysis are to obtain 1) the worst case delay suffered by a connection at a server and 2) the traffic description of a connection at the output of a server. Obviously, the first objective is directly related to calculating worst case packet delay by using (7). The description of a connection's traffic pattern at the output of a server is needed for carrying out a similar analysis at the subsequent server.

The variety and complexity of various networking components make the server analysis a challenging problem. Our analysis is facilitated by the observation that any network component can be systematically decomposed into simpler servers.

In the rest of this section, we are going to analyze the major components encountered in our FDDI-ATM-FDDI heterogeneous network.



(a) Decomposition of Server FDDI_S (b) Decomposition of $d_{\mathrm{FDDI\_S}}$

Figure 4: FDDI_S Server

4.3.1 FDDI_S Server

Let $\Gamma_{i,j,\mathrm{A}}(I)$ be the traffic of connection $M_{i,j}$ at the entrance of FDDI_MAC on $\mathrm{Host}_{i,j}$. We need to find the worst case delay at FDDI_S and the traffic description of connection $M_{i,j}$ at the exit of FDDI_S.

According to Figure 2, FDDI_S is the first major server after decomposition. In FDDI_S, the packet, consisting of one or more FDDI frames, will be first served by the FDDI_MAC on the sender host. When a frame departs from the host, it will propagate over the FDDI ring and be received by the interface device (ID_S). Thus, we can further decompose FDDI_S into two simple servers: an FDDI_MAC server and a Delay_Line server as shown in Figure 4. The worst case delay in FDDI_S is now given by

$$d^{wc}_{\mathrm{FDDD\_S}} = d^{wc}_{\mathrm{FDDI\_MAC}} + d^{wc}_{\mathrm{Delay\_Line}} \tag{8}$$

We now analyze each of the simple servers FDDI_MAC and Delay_Line. Considering the connection $M_{i,j}$, we have the following result :

THEOREM 1 *Let $Host_{i,j}$ connect to an FDDI ring of bandwidth BW_{FDDI} whose target token rotation time is TTRT, and the buffer size in the FDDI_MAC be $S_{i,j}$. If the synchronous bandwidth allocated to $M_{i,j}$ is $H_{i,j}$, then*

1. *the maximum length of the busy interval of the FDDI_MAC on $Host_{i,j}$ is given by*

$$B_{i,j} = \min_{\forall t > 0} \{ t \mid t \cdot \Gamma_{i,j,\mathrm{A}}(t) \leq avail(t) \} \tag{9}$$

2. *the maximum buffer requirement, $F_{i,j}$, is given by*

$$F_{i,j} = \max_{0 < t \leq B_{i,j}} \{ t \cdot \Gamma_{i,j,\mathrm{A}}(t) - avail(t) \} \tag{10}$$

3. *the worst case delay, $d^{wc}_{\mathrm{FDDI\_MAC}}$, is given by*

$$d^{wc}_{\mathrm{FDDI\_MAC}} = \begin{cases} \infty & \textit{if } F_{i,j} > S_{i,j}, \\ \chi & \textit{otherwise;} \end{cases} \tag{11}$$

4. *the output traffic, $\Gamma_{i,j,\mathrm{A'}}(I)$, is given by*

$$\Gamma_{i,j,\mathrm{A'}}(I) = \min(BW_{\mathrm{FDDI}}, \Upsilon) \tag{12}$$

where
$avail(t) = \max(0, (\lfloor t/TTRT \rfloor - 1) \cdot H_{i,j} \cdot BW_{\mathrm{FDDI}})$,

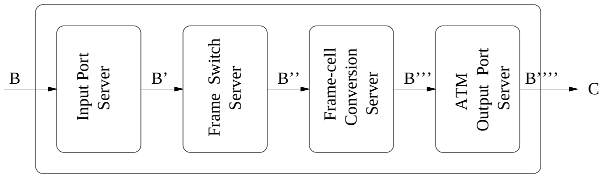
$$\chi = \max_{0 < t \leq B_{i,j}} \{ \min\{ d \mid avail(t+d) \geq t \cdot \Gamma_{i,j,A}(t) \} \},$$

and

$$\Upsilon = \max_{0 \leq t \leq B_{i,j}} \{ ((t+I) \cdot \Gamma_{i,j,\mathrm{A}}(t+I) - avail(t))/I \}.$$

For a proof of the above theorem, refer to [3].

(a) Decomposed Components of ID_S server



(b) Decomposition of $d_{\mathrm{ID\_S}}$

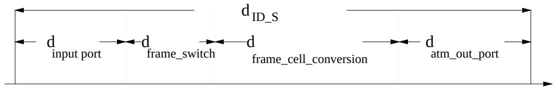


Figure 5: ID_S Server

Now we consider the Delay_Line server. Let $\Gamma_{i,j,A''}(I)$ be the traffic descriptor of connection $M_{i,j}$ at the exit of Delay_Line server. The Delay_Line server does nothing but simply delays every bit by a fixed amount of time which depends on the locations of the host and interface device on the ring. It does not change the traffic characteristics of a connection. Thus, we have

$$\Gamma_{i,j,\mathrm{A''}}(I) = \Gamma_{i,j,\mathrm{A'}}(I). \tag{13}$$

Because the Delay_Line server is the last server traversed by the connection on the path in FDDI_S, $\Gamma_{i,j,\mathrm{A''}}(I)$ represents the output traffic at the exit of FDDI_S. Furthermore,

$$d^{wc}_{\mathrm{Delay\_Line}} = \text{bit propagation delay from the host to the interface device.} \tag{14}$$

Note that the bit propagation delay can be calculated or measured. Substituting (11) and (14) into (8), we have $d^{wc}_{\mathrm{FDDI\_S}}$ computed.

4.3.2 ID_S Server

Let $\Gamma_{i,j,\mathrm{B}}(I)$ be the traffic descriptor for connection $M_{i,j}$ at the entrance of ID_S. Because the input of ID_S is the output of FDDI_S,

$$\Gamma_{i,j,\mathrm{B}}(I) = \Gamma_{i,j,\mathrm{A''}}(I) \tag{15}$$

where $\Gamma_{i,j,\mathrm{A''}}(I)$ is given in (13). The purpose of this sub-subsection is to obtain the worst case delay and output traffic descriptor for the ID_S server given $\Gamma_{i,j,B}(I)$.

First, we further decompose this server to simplify the analysis. The traffic passes through four stages in this device: 1) input port collects the frame from incoming FDDI segment 2) the frame is switched to the appropriate buffer that identifies a particular output port; 3) frame is then converted to ATM cells; 4) the output port schedules and transmits the ATM cells into the ATM backbone. Thus, we decompose ID_S into four simple servers, namely Input Port Server, Frame Switch Server, Frame_Cell_Conversion server, and Output_Port server. Figure 5 (a) shows such a decomposition. Consequently, the worst case delay encountered in ID_S can be decomposed into four components:

$$d^{wc}_{\mathrm{ID\_S}} = d^{wc}_{\mathrm{Input\_port}} + d^{wc}_{\mathrm{Frame\_Switch}} + d^{wc}_{\mathrm{Frame\_Cell\_Conversion}} + d^{wc}_{\mathrm{Output\_Port}} \qquad (16)$$

Input port just feeds frames from LAN segment to the Frame Switch Server. Hence, it is a constant delay server, i.e., a frame will encounter constant delay across this server. Traffic description of a connection will not change across this server. Thus, we have

$$\Gamma_{i,j,\mathrm{B'}}(I) \quad = \quad \Gamma_{i,j,\mathrm{B}}(I) \qquad (17)$$

$$d^{wc}_{\mathrm{Input\_Port}} = \text{ packet delay in the input port server} \quad (18)$$

This delay can be measured or can be specified by the manufacturer.

Frame Switch server is also a constant delay server, since it only places a frame in the appropriate buffer for the destination output port. The behavior of this server is similar to that of the Input Port server. Thus, we have

$$\Gamma_{i,j,\mathrm{B''}}(I) \quad = \quad \Gamma_{i,j,\mathrm{B'}}(I) \qquad (19)$$

$$d^{wc}_{\mathrm{Frame\_Switch}} = \text{ packet delay in the frame switch server} \quad (20)$$

This delay can also be measured or can be specified by the manufacturer.

For the Frame_Cell_Conversion server, we assume that a frame is converted into cells before the arrival of the next frame. This is reasonable because the ATM backbone has a higher transmission rate than the FDDI segment and consequently a frame should only experience processing delay but not queuing delay in the frame cell conversion. Formally, we can establish the following:

THEOREM 2 $\Gamma_{i,j,B'''}$, *the descriptor of the output traffic at the exit of the Frame_Cell_Conversion server, is given by*

$$\Gamma_{i,j,\mathrm{B'''}}(I) \quad = \quad \frac{\lceil \frac{I\Gamma_{i,j,\mathrm{B''}}(I)}{F_S} \rceil F_C C_S}{I} \qquad (21)$$

where F_S is the frame size (in bits), F_C is the number of cells converted by a frame, and C_S is the payload size of ATM cell (in bits). Further,

$$d^{wc}_{\mathrm{Frame\_Cell\_Conversion}} = \textit{the maximum processing time of a frame.} \qquad (22)$$

A proof is given in [3]. The processing time can be measured or should be given by the manufacturer.

The last server in ID_S is the Output_Port server. This server multiplexes cells from different connections and transmits them into the output link that connects to the ATM backbone. It actually functions in exactly the same way as an output port server in an ATM switch. In [2, 14], this type of server has been thoroughly analyzed. We will utilize the results from these previous studies to compute 1) $\Gamma_{i,j,\mathrm{B''''}}(I)$, the traffic descriptor of connection $M_{i,j}$ at the exit of Output_Port, and 2) $d^{wc}_{\mathrm{Output\_Port}}$, the worst case delay suffered by the traffic of connection $M_{i,j}$. Once this is done, using (18), (20) and (22) the worst case delay at server ID_S can be computed.

4.3.3 Other Servers

We have analyzed FDDI_S and ID_S servers and obtained the worst case delays and output traffic descriptors. The remaining servers to be analyzed are the ATM backbone server, the interface device, and FDDI ring at the receiver's site (i.e., ATM, ID_R, and FDDI_R).

ATM backbone has been extensively studied lately [12, 22]. In particular, the methods of computing delay bounds and output traffic descriptors have been reported in [2, 14, 15]. In this study, we adopt these methods. Analyzing ID_R and FDDI_R is similar to that of ID_S and FDDI_S, respectively. The only difference is that the process is reversed: cells from the ATM backbone are converted into FDDI frames. FDDI frames are transmitted from ID_R to the receiver's host. The details of analyzing ATM, ID_R, and FDDI_R servers will be not be given here due to the space limitation. Interested reader is referred to [3].

5 The CAC Algorithm

First, we will formally define the problem of connection admission control. We will then investigate how the bandwidth allocation of the source and destination FDDI of a new connection can affect the end-to-end worst case delays of other connection. It should be noted here that this bandwidth allocation scheme is for FDDI LAN in a heterogeneous environment. This is different from those described in [1, 24], which are applicable to FDDI-only LAN. Finally, we develop our connection admission control algorithm.

5.1 Problem Definition

Recall that the basic functions of the connection admission control are 1) to determine if the worst case delays of the requesting and existing connections are no more than their deadlines with the available network resource, 2) if the previous condition is satisfied, then to admit the requesting connection and to allocate it proper amount of network resource. The network resources include the buffer space in various components of the network and synchronous bandwidth in both sender and receiver's FDDI LAN segments. The buffer space has been implicitly taken into account during the computation of the worst case delays (e.g., Theorem 1 in Section 4.3.1.). Hence, the resource needed to be explicitly allocated is the synchronous bandwidth on FDDI segments. We would like to reiterate that bandwidth for FDDI segments in heterogeneous network is different from that used for FDDI-only LAN. The performance of the entire network has to be considered while allocating resources in a heterogeneous network. If we only consider the performance of one FDDI segment, the performance of other LAN segments may be adversely affected.

Let $M_{i,j}$ be the new connection requesting for admission. Let H_S and H_R be the synchronous bandwidth allocated to connection $M_{i,j}$ on FDDI_S and FDDI_R, respectively. Let $d^{wc}_{i,j}(H_S, H_R)$ denote the worst case delay of $M_{i,j}$, assuming that synchronous bandwidth H_S and H_R has been allocated to $M_{i,j}$. This allocation may affect the worst case delay of existing connections. Let $\mathcal{M}$ be the set of the identifiers of the connections which currently exist in the network, i.e.,

$$\mathcal{M} = \{(p,q) \mid M_{p,q} \text{ is currently active}\}. \tag{23}$$

Let $M_{p,q}$ be an existing connection, i.e., $(p,q) \in \mathcal{M}$. Let $d^{wc}_{p,q}(H_S, H_R)$ denote the worst case delay of $M_{p,q}$, assuming that synchronous bandwidth H_S and H_R has been allocated to $M_{i,j}$. Now the problem of admitting $M_{i,j}$ can be formally stated as follows: To find H_S and H_R for $M_{i,j}$ such that the following conditions hold:

1. For every $(p,q) \in \mathcal{M}$,

   $$d^{wc}_{p,q}(H_S, H_R) \quad \leq \quad D_{p,q} \tag{24}$$

   where $D_{p,q}$ is the deadline of connection $M_{p,q}$.

2. For $M_{i,j}$,

   $$d^{wc}_{i,j}(H_S, H_R) \quad \leq \quad D_{i,j} \tag{25}$$

Note that $d^{wc}_{i,j}(H_S, H_R)$ and $d^{wc}_{p,q}(H_S, H_R)$ are calculated using (7).

5.2 Feasible Region of H_S and H_R

It is clear that the connection admission problem is tightly coupled with the synchronous bandwidth allocation. We say that (H_S, H_R) is a *feasible allocation* if both (24) and (25) are satisfied. A region on H_S-H_R plane is feasible if every point in the region is a feasible allocation. In this subsection, we identify such a feasible region. Proof of theorems could not be given here due to space limitation. Interested reader may refer to [3].

First, we observe upper and lower bounds on H_S and H_R. Define $H_S^{max\_avai}$ and $H_R^{max\_avai}$ to be the synchronous bandwidth available on FDDI_S and FDDI_R at the time when connection $M_{i,j}$ requests for admission. $H_S^{max\_avai}$ and $H_R^{max\_avai}$ are given as follows.

$$H_S^{max\_avai} = \text{TTRT of FDDI\_S} - (\Omega_{\text{S}} + \Delta) \tag{26}$$

$$H_R^{max\_avai} = \text{TTRT of FDDI\_R} - (\Omega_{\text{R}} + \Delta) \tag{27}$$

where Δ is the protocol dependent overhead[1], Ω_{S} and Ω_{R} are the total synchronous bandwidth that has been allocated on FDDI_S and FDDI_R respectively. The synchronous bandwidth to be allocated to $M_{i,j}$ should not be more than that available. On the other hand, the amount of bandwidth allocated to a connection can not be arbitrarily small, because the overheads of FDDI frames will severely affect the throughput. Let $H_S^{min\_abs}$ and $H_R^{min\_abs}$ be the required minimum allocation on sender and receiver's FDDI rings. Clearly, $0 < H_S^{min\_abs} \leq H_S \leq H_S^{max\_avai}$ and $0 < H_R^{min\_abs} \leq H_R \leq H_R^{max\_avai}$.

Let (f,g) be the identifier of a connection that belongs to $\mathcal{M} \cup \{(i,j)\}$. Define region $R_{f,g}$ on the $H_S - H_R$ plane as follows

$$R_{f,g} = \quad \{(H_S, H_R) \mid d^{wc}_{f,g}(H_S, H_R) \leq D_{f,g}\} \tag{28}$$

Now we have the following theorems:

THEOREM 3 *For $H_S^{max\_avai} \geq H_S \geq H_S^{min\_abs}$ and $H_R^{max\_avai} \geq H_R \geq H_R^{min\_abs}$, the region represented by*

$$R_{f,g} = \quad \{(H_S, H_R) \mid d^{wc}_{f,g}(H_S, H_R) \leq D_{f,g}\} \tag{29}$$

on the H_S-H_R plane is a closed and convex region[1].

THEOREM 4 *If $H_S^{max\_avai}$ and $H_R^{max\_avai}$ do not satisfy either (24) or (25), the feasible region is empty. Otherwise, the feasible region is closed convex region in the first quadrant in the H_S-H_R plane and is given by*

$$R \quad = \quad R_{i,j} \bigcap \left(\bigcap_{M_{p,q} \in \mathcal{M}} R_{p,q} \right). \tag{30}$$

Figure 6 gives an example of the feasible region. It is a rectangle with the bottom side replaced by a concave curve. The geometric property helps us to develop an efficient CAC algorithm in the next subsection.

5.3 The Algorithm

Note that if the feasible region is not empty, the connection can be admitted. The remaining question is how to allocate bandwidth (H_S, H_R). As argued earlier, this bandwidth allocation problem is different from that studied in [1, 24], because 1) we are dealing with FDDI LANs in a heterogeneous environment, where the performance of the entire network, not just the FDDI segment, be kept in mind, 2) the allocation scheme is a dynamic allocation scheme i.e. the admission decision is made as the connections arrive. Hence, we propose the following rules for the allocation:

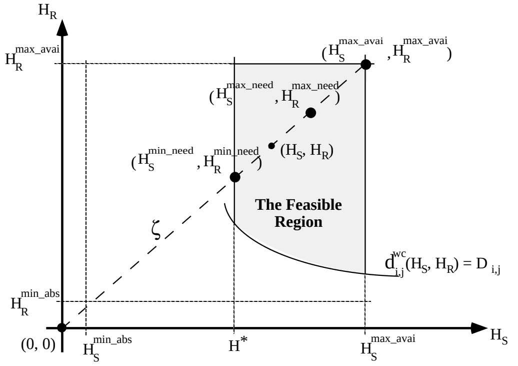
Rule 1. (H_S, H_R) must be in the feasible region. This is obviously necessary.

Rule 2. The ratio of H_S to H_R should be the same as the ratio of $H_S^{max\_avai}$ to $H_R^{max\_avai}$. That is, (H_S, H_R) should be a point on the line segment (ζ) connecting $(0,0)$ and $(H_S^{max\_avai}, H_R^{max\_avai})$. See Figure 6. By doing so, we reserve resources from both rings in a proportional manner.

Note that allocating bandwidth for the new connection is a complex problem due to the fact that the CAC algorithm does not have the knowledge about the connections that might arrive in future. Given this situation, we propose the above proportional scheme. In this way, if an FDDI LAN has a higher load than the other, a lesser amount of bandwidth will be allocated from the heavily loaded FDDI while a higher amount will be allocated from the lightly loaded FDDI, making the remaining bandwidth on the two FDDI segments relatively balanced. Thus, a proportional allocation scheme is a good heuristic to use.

---

[1] A region is convex if and only if for any two points in the region, all the points on the segment connecting these two points are also in the region. A region is closed if it is a closed set. For a formal definition of a closed set, see [21].



$H^* = \max\{H_S$ that satisfies (24) and (25) $\}$

Figure 6: Selecting (H_S, H_R) in the Feasible Region.

Let ($H_S^{min\_need}$, $H_R^{min\_need}$) be the lower intersecting point of line ζ and the feasible region (see Figure 6). ($H_S^{min\_need}$, $H_R^{min\_need}$) is the *minimum bandwidth* that needs to be allocated if the allocation has to be chosen from segment ζ. By Theorem 3, the feasible region is closed and convex. Hence, any point (H_S, H_R) between ($H_S^{min\_need}$, $H_R^{min\_need}$) and ($H_S^{max\_avai}$, $H_R^{max\_avai}$) on line ζ is a feasible allocation. The problem is which point to choose.

One would argue to select (H_S, H_R) = ($H_S^{max\_avai}$, $H_R^{max\_avai}$). That is, $M_{i,j}$ is allocated to the maximum available bandwidth. The problem of this algorithm is that it allocates all the bandwidth available on sender and receiver's FDDI rings to connection $M_{i,j}$. This will result in the rejection of any future connection originated from or designated to these two rings simply because no bandwidth is available.

On the other hand, one might like to choose (H_S, H_R) = ($H_S^{min\_need}$, $H_R^{min\_need}$). That is, $M_{i,j}$ is allocated with the minimum amount of bandwidth which just makes it possible to meet all the deadlines. With this allocation, the worst case delay of some connection(s) may be very tight – very close to the deadline(s). Because of this, the disturbance generated by a future connections may cause it (them) to miss its (their) deadline(s). If this happens, the new connection cannot be admitted. Thus, this kind of allocation is not ideal either.

Now let us discuss the maximum amount of the bandwidth needed by $M_{i,j}$. We need to define relation "<"over vectors. For $a, b, c, d \in R$, (a, b) < (c, d) if and only if $a < c$ and $b < d$. Similarly, relations "≤", ">", and "≥", and functions min and max can be defined. Consider allocation (H_S, H_R) such that $(H_S^{max\_avai}, H_R^{max\_avai}) \geq (H_S, H_R) \geq (H_S^{min\_need}, H_R^{min\_need})$. If for all $(p, q) \in \mathcal{M}$,

$$d_{p,q}^{wc}(H_S^{max\_avai}, H_R^{max\_avai}) = d_{p,q}^{wc}(H_S, H_R) \qquad (31)$$

$$d_{i,j}^{wc}(H_S^{max\_avai}, H_R^{max\_avai}) = d_{i,j}^{wc}(H_S, H_R), \qquad (32)$$

then allocation (H_S, H_R) is a better choice than ($H_S^{max\_avai}$, $H_R^{max\_avai}$). This is because (H_S, H_R) achieves the same delay performance while it may save some bandwidth in comparison with ($H_S^{max\_avai}$, $H_R^{max\_avai}$). We can define the maximum amount of the bandwidth needed by $M_{i,j}$ as follows:

$$(H_S^{max\_need}, H_R^{max\_need}) = \min_{(H_S, H_R) \in \zeta} \{(H_S, H_R) \mid (H_S, H_R) \text{ satisfies both (31) and (32)}\}. \quad (33)$$

Obviously, the bandwidth allocation (H_S, H_R) should satisfy

$$(H_S^{min\_need}, H_R^{min\_need}) \leq (H_S, H_R) \leq (H_S^{max\_need}, H_R^{max\_need}). \qquad (34)$$

Based on this observation, we propose to choose (H_S, H_R) as follows:

$$H_S = H_S^{min\_need} + \beta \cdot (H_S^{max\_need} - H_S^{min\_need}) \quad (35)$$

$$H_R = H_R^{min\_need} + \beta \cdot (H_R^{max\_need} - H_R^{min\_need}) \quad (36)$$

where β is a real number between zero and one. It is easy to verify that this selection conforms to the two rules mentioned earlier. The main steps for the CAC algorithm are given below :

1. Compute $H_S^{max\_avai}$ and $H_R^{max\_avai}$ by using (26) and (27).
2. If $H_S^{max\_avai}$ and $H_R^{max\_avai}$ do not satisfy (24) and (25) then reject $M_{i,j}$.
3. Do a binary search, by using (24) and (25), along the straight line joining points ($H_S^{min\_abs}$, $H_R^{min\_abs}$) and ($H_S^{max\_avai}$, $H_R^{max\_avai}$) to identify ($H_S^{min\_need}$, $H_R^{min\_need}$).
4. Do a binary search, by using (31) and (32), along the straight line joining points ($H_S^{min\_need}$, $H_R^{min\_need}$) and ($H_S^{max\_avai}$, $H_R^{max\_avai}$) to identify ($H_S^{max\_need}$, $H_R^{max\_need}$).
5. Compute H_S and H_R by using (35) and (36). Accept $M_{i,j}$ with bandwidth allocation (H_S, H_R).

6 Performance Evaluation

In this section we present performance results for the FDDI-ATM-FDDI heterogeneous networks which use our CAC algorithm.

The performance metric we are interested in is *admission probability* (AP) which can be estimated as the ratio of total number of admitted connections to total number of connection admission requests. This metric has been used in evaluation of real-time communication systems.

To obtain the performance data, we developed a program to simulate an FDDI-ATM-FDDI network. The simulation program is written in C programming language and run in a Sun/Solaris environment. The simulated network consists of three interface devices and three ATM switches. In each FDDI ring, there are four hosts. In the simulation, requests for connection establishment arrive as a Poisson process with rate λ.

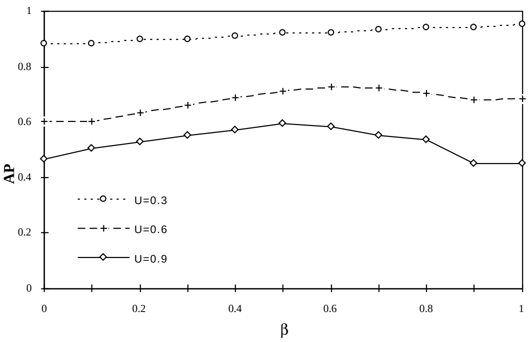


Figure 7: Sensitivity of β.

The source node of a connection is randomly chosen from the hosts which are currently inactive. The route of a connection will always go through the ATM backbone. All admitted connections have a life time which is exponentially distributed with mean $\frac{1}{\mu}$. The source traffic of a connection is characterized by the dual periodic model. That is, the maximum arrival rates are C_1/P_1 bits/second in an interval of P_1 seconds and C_2/P_2 bits/second in an interval of P_2 seconds. This model generalizes the one period model, allowing certain burstiness in source traffic [14]. The $\Gamma(I)$ function for this type of traffic is given by

$$\Gamma(I) = \frac{1}{I}(\lfloor \frac{I}{P_1} \rfloor C_1 + \min(C_1, \lfloor \frac{I - \lfloor \frac{I}{P_1} \rfloor P_1}{P_2} \rfloor C_2 + \min(C_2, I - \lfloor \frac{I}{P_1} \rfloor P_1 - \lfloor \frac{I - \lfloor \frac{I}{P_1} \rfloor P_1}{P_2} \rfloor P_2))) \quad (37)$$

The long term arrival rate, ρ, of a connection is given by

$$\rho \;=\; \lim_{I \to \infty} \Gamma(I) = \frac{C_1}{P_1}. \quad (38)$$

The Average Utilization in the ATM backbone is the ratio of the average load on one link of the backbone to the backbone link capacity. The *average load on one link of the backbone* equals $\frac{\lambda}{3\mu} \cdot \frac{C_1}{P_1}$ and the backbone link capacity is equal to 155 Mbps.

6.1 Sensitivity of β

To assess the sensitivity of β, we simulate the system with the average link utilization (U) in the ATM backbone being 0.3, 0.6, and 0.9. We vary the values of β from 0 to 1. The results are shown in Figure 7. We can make the following observations from this figure.

- When the system load is heavy ($U = 0.9$), the admission probability is sensitive to the selection of β. The admission probability is lower when either $\beta = 0$ or $\beta = 1$. Performance can be improved by choosing β between 0 and 1.
- The sensitivity is small when the load is light. In addition, at $U = 0.3$, AP increases as β increases. This is because when load is light, allocating more bandwidth can reduce the impact of disturbance by future connections without jeopardizing the chances of admitting them.

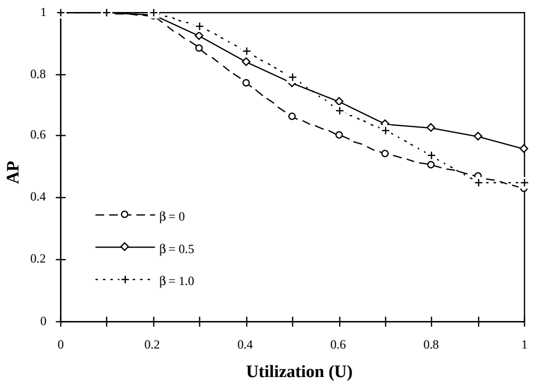


Figure 8: Sensitivity of System Load.

- Nevertheless, the sensitivity is limited in that for a wide range of β values, the system achieves good performance, close to its maximum. Generally speaking, a good range for β for all the three loading conditions seems to be [0.4, 0.7].

6.2 Sensitivity of the System Load

To assess the sensitivity of U, we simulate the system with $\beta = 0$, 0.5, and 1.0. We vary the values of U from 0 to 1. The results are shown in Figure 8. We can make the following observations.

- The admission probability decreases as the utilization increases. This is expected because as the system load increases, the chance of admitting a new connection is reduced.
- Choice of $\beta = 0.5$ seems to be reasonable. Especially, when the load is heavy (say, $U = 0.9$), the admission probability with $\beta = 0.5$ is much better than the case of $\beta = 0$ or $\beta = 1$.

7 Final Remarks

We developed a decomposition approach to obtain the worst case delay in FDDI-ATM-FDDI network. The method is intuitive and can be easily understood. It is compatible with existing network standards and can be readily used in existing systems. Based on the delay analysis, we designed a CAC algorithm that is capable of 1) determining whether the deadline constraints of the requesting connection and the existing ones can be met; 2) if the deadline constraints can be met, then admitting the new connection and allocating proper amount of the network resource for it. We argued that the resource allocation should be sufficient but not excessive. Improper allocation of network resource may jeopardize the chance for admission of future connections. From our study, it is clear that the bandwidth allocation scheme as reported in [1, 24] for an FDDI-only network may not be applied directly to an FDDI-ATM-FDDI heterogeneous network. This is due to the fact that in a heterogeneous network, an allocation scheme, that is efficient, when one LAN segment is considered separately, may not be as good for other LAN segments. Hence, an allocation scheme that takes the performance of the entire network into consideration should be used.

Our methodology can be easily extended to the networks with different configurations. For example, if the LAN segments are IEEE 802.5 token rings, one only needs to analyze an 802.5_MAC server in addition to the servers that have been analyzed in this paper.

Acknowledgment

This work was partially sponsored by the Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant number F49620-96-1-1076 and by Texas Higher Education Coordinating Board under its Advanced Technology Program with grant number 999903-204. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research, the U.S. Government, Texas State Government, Texas Higher Education Coordinating Board, or Texas A&M University.

References

[1] G. Agrawal, B. Chen, W. Zhao, and S. Davari. “Guaranteeing Synchronous Message Deadlines in High Speed Token Ring Networks with Timed Token Protocol”. In *Proc. of the 12th IEEE ICDCS*, pages 468–475, June 1992.

[2] A. Raha. *Real Time Communication in ATM Networks*. PhD thesis, Department of Computer Science, Texas A&M University, 1996.

[3] B. Chen. *Real-Time Communication in Homogeneous and Heterogeneous Networks*. PhD thesis, Department of Computer Science, Texas A&M University, 1996.

[4] B. Chen, S. Kamat, and W. Zhao. Fault-tolerant real-time communication in FDDI-based networks. In *Proceedings of the IEEE Real-Time Systems Symposium*, pages 141–151, December 1995.

[5] R. L. Cruz. “A Calculus for Network Delay, Part I: Network Elements in Isolation”. *IEEE Transactions on Information Theory*, 37(1):114–131, January 1991.

[6] R. L. Cruz. “A Calculus for Network Delay, Part II: Network Analysis”. *IEEE Transactions on Information Theory*, 37(1):132–141, January 1991.

[7] D. Ferrari and D. C. Verma. “A Scheme for Real-Time Channel Establishment in Wide-Area Networks”. *IEEE Journal on Selected Areas in Communications*, SAC-8(3):368–379, April 1990.

[8] D. Ferrari, A. Banerjee, and H. Zhang. “Network Support for Multimedia - A Discussion of Tenet Approach”. *Technical report TR-92-072*, International Computer Science Institue, Berkeley, CA, October 1992.

[9] J. F. Kurose, M. Schwartz, and Y. Yemini. Controlling window protocols for time-constrained communication in a multiple access environment. In *Proc. of the 8th IEEE International Data Communication Symposium*, 1983.

[10] J. Y-T. Leung, T. W. Tam, and G. H. Young. On-line routing of real-time messages. In *Proc. of the IEEE RTSS*, pages 126–135, Lake Buena Vista, Florida, December 1990.

[11] N. Malcolm and W. Zhao. “Use of The Timed-Token Protocol for Real-Time Communications”. *IEEE Computer*, 27(1):35–41, January 1994.

[12] A. Mehra, A. Indiresan, and K. G. Shin. Resource management for real-time communication: Making theory meet practice. In *Proc. of RTAS*, 1996.

[13] S. Mukherjee, D. Saha, M. C. Saksena, and S. K. Tripathi. A bandwidth allocation scheme for time constrained message transmission on a slotted ring LAN. In *Proc. of the IEEE RTSS*, pages 44–53, December 1993.

[14] A. Raha, S. Kamat, and W. Zhao. “Guaranteeing End-to-End Deadlines in ATM Networks”. In *Proc. of the 15th ICDCS*, pages 60–68, May 1995.

[15] A. Raha, S. Kamat, and W. Zhao. “Using Traffic Regulation to Meet End-to-End Deadlines in ATM LANs”. In *Proc. of the ICNP*, November 1995.

[16] A. Raha, S. Kamat, and W. Zhao. “Admission Control for Hard Real-Time Connections in ATM LANs”. In *Proc. of INFOCOM '96*, pages 180–188, March 1996.

[17] A. Raha and W. Zhao. Evaluation of admission policies in ATM based embedded hard real-time systems. Technical report, Department of Computer Science, Texas A&M University, June 1994.

[18] L. Sha, S. S. Sathaye, and J. K. Strosnider. Scheduling real-time communication on dual-link networks. In *Proc. of the IEEE RTSS*, pages 188–197, December 1992.

[19] K. G. Shin and C-J Hou. Analysis of three contention protocols in distributed real-time systems. In *Proc. of the IEEE RTSS*, pages 136–145, December 1990.

[20] J. K. Strosnider. *Highly Responsive Real-Time Token Rings*. PhD thesis, Department of Electrical and Computer Engineering, Carnegie Mellon University, 1988.

[21] S. Willard. *General Topology*. Addision-Wesley, 1st edition, 1970.

[22] 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. of ACM SIGCOMM'93*, pages 2–12, September 1993.

[23] H. Zhang and D. Ferrari. “Improving Utilization for Deterministic Service in Multimedia Communication”. In *International Conference on Multimedia Computing and Systems*, pages 295–304, May 1994.

[24] S. Zhang, A. Burns, and A. Wellings. “An Efficient and Practical Local Synchronous Bandwidth Allocation Scheme for the Timed-Token MAC Protocol”. *Proc. of IEEE Infocom '96*, volume-2 pages 920-927, March 1996.

[25] Q. Zheng and K. G. Shin. “On the Ability of Establishing Real-Time Channels in Point-to-Point Packet-Switched Networks”. *IEEE Transactions on Communications*, 42(2/3/4), Feb./Mar./Apr. 1994.