Stability in ATM Networks

Chengzhi Li*, Amitava Raha†, and Wei Zhao*

Abstract

In this paper, we address the issues of stability in ATM networks. A network is stable if and only if all the packets have a bounded delay. We first consider ATM networks with FCFS scheduling policy. We then study networks with priority driven scheduling policy. For each network, we develop criteria for testing the stability of an ATM network and methods of deriving delay bounds in a stable network.

In previous work, the Cruz-Gallager-Parekh ring has been a "benchmark" architecture to study the stability problem. For example, Gallager and Parekh claimed that the ring with size no more than four switches is stable when the total utilization of the links is less than 100% *[10]. We validated this result. Furthermore, we find that a ring with large number of switches is stable if the total utilization of the links is less than or equal to* 73%.

1 Introduction

In this paper, we address the issue of stability in communication networks. A network is said to be *stable* if all the data packets experience bounded delays within the network. Obviously, unbounded packet delays will have a detrimental impact on the performance of any distributed application communicating via the network. Therefore, ensuring stability within the network has been a pivotal issue in the design and management of communication networks. Network stability has been a research problem studied by many researchers. For example, it was established that for satellite packet switching using the ALOHA protocol if the network throughput is pushed much above 36% then the network can be potentially unstable [7].

We choose ATM to address the issue of stability. ATM networks are expected to provide guaranteed quality of services (QoS). With the proliferation of multimedia applications, bounded delay has become an important quality of service requirement. Hence, it is more important to *ensure* stability in ATM networks.

The objectives of this paper are to first develop criteria for testing the stability of an ATM network and then once network stability is established, to determine the delay bounds of all the connections in the network.

We first consider an ATM network in which first come first service (FCFS) is the scheduling policy employed at the multiplexors, (for example the output link schedulers of an ATM switch). We consider the FCFS scheduling policy because it is widely available in practical networks. For networks with arbitrary topology, we develop the stability criteria and an iterative method to derive the delay bounds in a stable network. We show that for a stable network the iteration procedure converges. Further, we also show that the criteria for stability in a FCFS based ATM network also applies to a network using any work conserving scheduling policy.

The stability problem for ATM networks has been addressed by several researchers in the context of a specialized ring topology (called Cruz-Gallager-Parekh ring)[1, 10]. This network presents (in some sense) the worst case scenario and becomes a "benchmark" to compare the techniques and results in the study of stability. By applying theorems from fixed point theory and by utilizing the link transmission constraints we found that Cruz-Gallager-Parekh rings with large ring size are stable if the total utilization of the links is less than or equal to 73%. Our result also validates Parekh's claim that for Cruz-Gallager-Parekh ring's with ring size less than or equal to four the system is stable when the total utilization of the links is less than 100%[10].

We also addressed the stability problem for a network with multiplexors employing a priority driven scheduling mechanism. The stability of such networks depends on the priority assignment mechanism employed at the different servers within the network. We found that one class of priority assignment mechanism, namely a static, fixed, and globally distinct priority assignment, will guarantee the stability as long as the utilization of the individual links is less than 100%.

*Computer Science Dept., Texas A&M University, College Station, TX 77840.

†Fujitsu Software Corporation, San Jose, CA 95134.

2 Previous Work

Considerable progress has been made recently towards solving the stability problem in ATM networks. The key factor which causes instability in practical ATM networks is the presence of cyclic inter-dependencies between the cell traffic of different connections. There have been several approaches to solve the problem. The first approach considers ATM networks with specialized mechanism's (both hardware and software) so as to prevent cyclic inter-dependencies between connections. Much of the previous studies using this approach have concentrated on designing specialized scheduling policies for ATM switches [5, 10, 15]. In [5, 15], non-work conserving systems were studied where traffic regulation and restoration are used to ensure the network stability. In [10], system stability was ensured by considering a specialized weight assignment for the PGPS scheduling policy in order to restore the connections traffic at the output of a sever.

The other approaches deal with the problem using information on network topology or application semantics. For example, [1, 10] considered ATM networks with the specialized ring topology. In [12], a CAC algorithm for real-time applications was developed. The CAC algorithm addresses the traffic dependence issue and determines if all the deadlines of real-time messages can be met. In this paper, we consider general ATM networks and establish stability criteria for the networks with arbitrary topology and without deadline specifications.

3 Problem Definition

In this section, we formally introduce the problem of stability in ATM networks. First we present the preliminary concepts and techniques which we employ for studying the problem. We will also introduce some of the notations and terminologies we use in the rest of this paper.

First, we develop an abstraction of the system. In an ATM network, hosts are connected to ATM switches and ATM switches are connected to each other using physical links. As an example, consider the 4 switche ATM network shown in Figure 1. Although this example may not be representative of a typical ATM network, it is used to illustrate important concepts discussed in this paper. As shown in Figure 1, the switch itself consists of input ports, a switching fabric, and output ports. An ATM cell arrives at an input port of a switch, is transported by the switching fabric to an output port, and is transmitted along the physical link associated with the output port. We model the ATM network as a collection

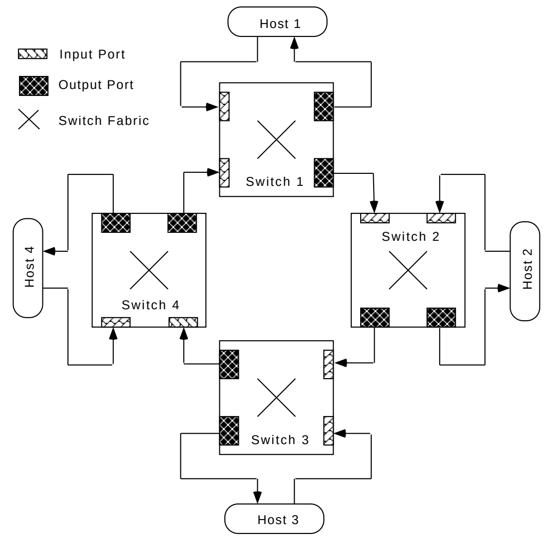

Figure 1: An example of a four switch ATM network.

of servers. A server is an abstraction of a network component that is traversed by a connection's cells. Therefore, the input ports, the switching fabric, the output ports, and the physical links can be modeled as servers serving ATM connections.

The servers are classified into two categories: *constant servers* and *variable servers*[1, 12, 13]. A constant server is the one that offers a constant delay to each cell that uses it and does not by itself change the traffic flow characteristics of a connection. For example, physical links and the switching fabric are constant delay line servers. The function of an input port is to demultiplex 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 demultiplexor server. 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 buffered at an output port and transmitted in an order that is determined by the scheduling discipline employed by the switch hardware. Note that an multiplexor server must be considered as a variable server since the delay suffered by a cell in this server varies depending upon 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.

Figure 2 shows the same network modeled as a collection of servers serving four connections M_1, M_2, M_3,

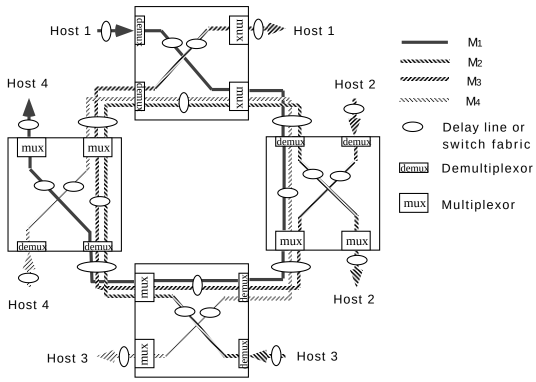

Figure 2: Server representation of ATM network.

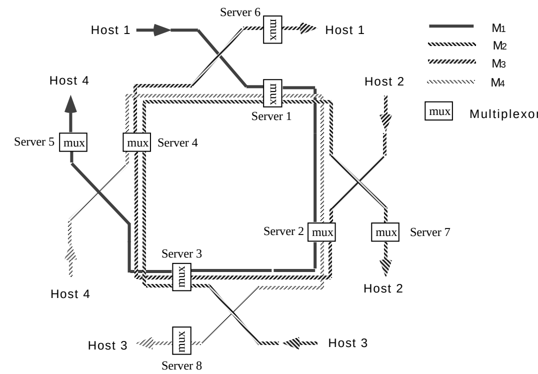

Figure 3: Connection-server graph representation.

and M_4.

Consider the connection M_1 from Host 1 to Host 4 shown in Figure 2. M_1 traverses 9 delay line servers (5 physical links and 4 switching fabrics) and 4 demultiplexor servers (input ports of 4 switches) all of which are constant servers. M_1 also traverses 4 multiplexor servers (output ports of 4 switches) which are variable servers. Recall that the constant servers serving M_1 only add a fixed amount of delay to M_1's cells and do not change M_1's traffic characteristics. Hence, their impact on M_1 can be accounted for by simply subtracting the total delay suffered by M_1 at these servers from M_1's delay requirement. The same holds for other connections. In the rest of the paper, we assume that the delay requirements of connections are modified in such a way. Consequently, we eliminate all the constant servers from further consideration and focus only on the variable servers in the remainder of the paper. Hence, now we can view a connection as being served by a sequence of variable servers only. We will often omit the prefix 'variable' when referring to variable servers to avoid repetitiousness. Further, we assume that each of these servers is given a unique identity which is an integer.

We use the above abstraction to construct a connection-server graph. A connection-server graph is constructed as a labeled, directed graph with the servers as its nodes. A directed edge is introduced from server m to server n if there is a connection that is served by server m followed by server n. The edge is labeled by the connection that uses the servers in immediate sequence. Figure 3 shows the connection-server graph corresponding to the system shown in Figure 2. The sources and destinations of connections are also shown in the connection-server graph. The connection-server graph is used to facilitate the discussion of network stability later.

We will use the following notations concerning the set of connections in the connection-server graph.

- N is the total number of connections in the system.
- M_i is the i^{th} connection in the system, where $1 \leq i \leq N$.
- $\mathcal{M}$ is the set of N connections competing for resources within the ATM network. That is,

$$\mathcal{M} = \{M_1, M_2, \ldots, M_i, \ldots, M_N\}. \qquad (1)$$

- K is the total number of servers in the system.
- L_j is the number of links which enter server j. We assume that a link into server j is uniquely identified by an integer in the range $[1, L_j]$.
- $\mathcal{F}_{k,j}(I)$ is the maximum number of cells that can arrive at server j over its k^{th} input link in any interval of length I.
- $m_{i,j,k}$ denotes the membership of connection M_i in the k^{th} link into server j. That is,

$$m_{i,j,k} = \begin{cases} 1, & \text{if the connection } M_i \text{ enters server } s_j \text{ by the } k^{th} \text{ link,} \\ 0, & \text{otherwise.} \end{cases} \qquad (2)$$

- $s(i,j)$ denotes the identity of the j^{th} server in the path of connection M_i.
- S_i is the total number of servers serving connection M_i.
- H_i is the sequence of servers serving connection M_i.

$$H_i = < s(i,1), \ldots, s(i,j), \ldots, s(i,S_i) > . \qquad (3)$$

- $\chi_{i,h,j}$ defines the precedence relation for the servers in M_i's path.

$$\chi_{i,h,j} = \begin{cases} 1, & \text{if in the connection } M_i\text{'s path} \\ & \text{server } h \text{ precedes server } j, \\ 0, & \text{otherwise.} \end{cases}$$

- $G_{i,j}$ is the set of servers traversed by a cell of M_i before arriving at server j.
- $F_{i,j}(I)$ is the maximum number of M_i's cells that can arrive at server j in any interval of length I.
- $d_{i,j}$ is the worst case delay experienced by a cell from connection M_i at server j.
- d_i is the worst case end-to-end cell delay experienced by a cell of connection M_i and is given by

$$d_i = d_{i,s(i,1)} + d_{i,s(i,2)} + \ldots + d_{i,s(i,S_i)}. \quad (4)$$

For many applications deployed over ATM networks end-to-end cell delay is an important QoS parameter. Therefore, determination of bounds on end-to-end cell delay is a pivotal issue in ATM network analysis. To efficiently analyze the end-to-end delays in ATM networks two important issues must be addressed.

1. System Stability: An ATM network is *stable* if and only if for every connection M_i, $M_i \in \mathcal{M}$

$$d_i \le D, \quad (5)$$

   where D is a non-negative real number. ATM networks with arbitrary topology can have cycles in their connection server graphs. The presence of cycles in the connection server graph may lead to feedback dependency loops in the system. Due to the presence of these loops the system may be potentially unstable, i.e., connections in such a system can have unbounded cell delays. Obviously, it is not a fruitful exercise to determine delay bounds in a potentially unstable system. Therefore, determination of system stability is a critical step in the determination of bounds for connection's cell delay.

2. End-to-end cell delay bounds: This addresses the issue of deriving the end-to-end delays experienced by cells of every connection in a stable network. Since, we represent the connection as a sequence of servers, the end-to-end cell delay of a connection is the sum of the worst case delay it suffers at every server along its route. Therefore, in order to derive the end-to-end cell delay of a connection we should derive the delay upper bound at the servers.

Our goals are to first develop the criteria of stability in an ATM network and then to derive delay bounds at every server in the stable system. We analyze the system of servers represented by the connection-server graph. However, such an analysis requires

1. an uniform characterization of the traffic at the input of every server, and
2. a systematic study of the scheduling policies employed at the servers.

In this paper, we assume that the connection M_i's cell traffic at the input of any server is characterized by $F_{i,j}(I)$. $F_{i,j}(I)$ specifies the maximum number of cells that can arrive at server j in any interval of length I. We further assume that the cell traffic at the input of connection M_i is characterized by the piecewise linear model with parameters β_i, and ρ_i[1]. Hence, for any connection M_i, we have

$$F_{i,s(i,1)}(I) = \min(I, \beta_i + \rho_i * I). \quad (6)$$

The cell traffic of many applications can be characterized by the linear model. Nevertheless, in a later section we are going to extend our results to encompass applications with any other reasonable traffic description. The following theorem gives $F_{i,j}(I)$, M_i's traffic characterization at the input of server j, when the source traffic of M_i is specified by (6).

Theorem 1 *For any connection M_i, if $F_{i,s(i,1)}(I) = \min(I, \beta_i + \rho_i * I)$ and $j \in H_i, j \neq s(i,1)$, then*

$$F_{i,j}(I) = \begin{cases} I, & I \le \zeta_{i,j}, \\ \beta_i + \rho_i * \sum_{g \in G_{i,j}} d_{i,g} \\ +\rho_i * I, & \zeta_{i,j} \le I. \end{cases} \quad (7)$$

where

$$\zeta_{i,j} = \frac{\beta_i + \rho_i * \sum_{g \in G_{i,j}} d_{i,g}}{1 - \rho_i}. \quad (8)$$

The proof of the theorem is given in [8].

Theorem 1 implies that if the traffic entering the network is constrained by a continuous piecewise linear function, so is the traffic flowing inside the network.

Since in an ATM network, cells from different connections are multiplexed at the multiplexor and transmitted over its output link, it is useful to characterize the aggregate cell traffic over a single link. The description of the aggregate cell traffic over a link is given by the following theorem.

Theorem 2 *The aggregate cell traffic over the k^{th} link at server j is given by*

$$\mathcal{F}_{k,j}(I) = \begin{cases} I, & I \le \eta_{j,k}, \\ \sum_{i=1}^{N} m_{i,j,k} * (\beta_i + \rho_i * \sum_{g \in G_{i,j}} d_{i,g} \\ +\rho_i * I), & \eta_{j,k} \le I. \end{cases} \quad (9)$$

where

$$\eta_{j,k} = \frac{\sum_{i=1}^{N} m_{i,j,k} * [\beta_i + \rho_i * \sum_{g \in G_{i,j}} d_{i,g}]}{1 - \sum_{i=1}^{N} m_{i,j,k} * \rho_i}. \quad (10)$$

The proof of the theorem is given in [8].

The results of Theorems 1 and 2 are important in analyzing the delays at the servers in the network. We make use of Theorems 1 and 2 to derive the main results developed in this paper.

The scheduling policy at a server determines the order in which cells from a connection are transmitted at the output of the server. Hence, the server scheduling policy has a direct impact on the delays experienced by a connection's cell at a server as well as on the distortion of the connections traffic within the network, i.e., the connection's traffic may become more bursty. The increase in burstiness may perturb the traffic flow of other connections in the network, resulting in an increase in the cell delays and traffic burstiness of those connections. This scenario may be aggravated if the connection-server graphs contains feedback loops, and may lead the system to an unstable state in which the delays of the connections become unbounded. In the rest of the paper, we study two popular scheduling policies: First come first serve (FCFS) and priority driven scheduling. We develop criteria to ensure system stability and give expressions for the delay bounds with the above scheduling policies.

4 FIFO DRIVEN Scheduling

In this section, we establish the stability criteria and delay bounds for an FCFS based ATM network. Due to their implementation efficiency and cost, ATM networks with FCFS servers are widely prevalent in the market. An FCFS server transmits cells on its output link in the order they arrive at its input. Therefore, the worst case delay experienced by any cell at the server is the same for any connection traversing it.

We need some notations and definitions which will be used in this section. For an FCFS based ATM network let $d_{*,j}$ be the worst case cell delay at server j. Let $\vec{d}_{FCFS}$ be the delay vector for all servers in the system, i.e.,

$$\vec{d}_{FCFS} = (d_{*,1},\ d_{*,2},\ \cdots,\ d_{*,K})^{\top}_{K \times 1}. \quad (11)$$

For any vector $\vec{x}$ of size K, i.e., $\vec{x} = (x_1,\ x_2,\ \ldots,\ x_K)^{\top}_{K\times 1}$ we define

$$\|\vec{x}\| = \max_{i=1,2,\ldots,K} |x_i|. \quad (12)$$

Let μ be the maximum of average link utilizations in the network. For a system of N connections whose input traffic is described by the piecewise linear function, we have

$$\mu = \max_j \sum_{k=1}^{L_j} \sum_{i=1}^{N} m_{i,j,k} * \rho_i. \quad (13)$$

At server j, let $Z_j(\vec{d}_{FCFS})$ be a function of $\vec{d}_{FCFS}$ such that

$$Z_j(\vec{d}_{FCFS}) = C_{0,j} + \sum_{s=1, s\ne j}^{s=K} C_{s,j} * d_{*,s}, \quad (14)$$

where

$$C_{0,j} = \sum_{k=1}^{L_j} \sum_{i=1}^{N} m_{i,j,k} * \beta_i + \left(\sum_{k=1}^{L_j} \sum_{i=1}^{N} m_{i,j,k} * \rho_i - 1 \right) * \frac{\sum_{i=1}^{N} m_{i,j,l_j} * \beta_i}{1 - \sum_{i=1}^{N} m_{i,j,l_j} * \rho_i}, \quad (15)$$

and

$$C_{s,j} = \sum_{k=1}^{L_j} \sum_{i=1}^{N} m_{i,j,k} * \rho_i + \left(\sum_{k=1}^{L_j} \sum_{i=1}^{N} m_{i,j,k} * \rho_i - 1 \right) * \frac{\sum_{i=1}^{N} m_{i,j,l_j} * \rho_i * \chi_{i,s,j}}{1 - \sum_{i=1}^{N} m_{i,j,l_j} * \rho_i}. \quad (16)$$

In (15) and (19) l_j, $1 \le l_j \le L_j$, is the index of a link at server j such that

$$\eta_{j,l_j} = \max(\eta_{j,1}, \eta_{j,2}, \ldots, \eta_{j,L_j}), \quad (17)$$

where $\eta_{j,k}$ is given by (10). Furthermore, we define

$$\mathcal{E}_{s,j} = \min_{k=1,\cdots,L_j} \{ \frac{\sum_{i=1}^{N} m_{i,j,l_j} * \rho_i * \chi_{i,s,j}}{1 - \sum_{i=1}^{N} m_{i,j,l_j} * \rho_i} \}, \quad (18)$$

and

$$\tilde{C}_{s,j} = \sum_{k=1}^{L_j}\sum_{i=1}^{N} m_{i,j,k} * \rho_i + \left(\sum_{k=1}^{L_j}\sum_{i=1}^{N} m_{i,j,k} * \rho_i - 1\right) * \mathcal{E}_{s,j}. \quad (19)$$

4.1 Networks with Arbitrary Topology

In this subsection, we consider an FCFS based ATM network with arbitrary topology. Due to the arbitrariness of the topology, a connection-server graph can contain cycles even if the individual connection paths are acyclic. These cycles may cause cyclic dependencies among the traffic of different connections, which may lead the system to be unstable. The main result is given in the following theorems.

Theorem 3 *For an ATM network with arbitrary topology and FCFS based servers if*

$$\mu < 1, \quad (20)$$

and

$$\nu = \max_j\left(\sum_{s=1, s\neq j}^{s=K} \tilde{C}_{s,j}\right) < 1, \quad (21)$$

then

- *the system is stable, and*
- $\vec{d}_{FCFS}$ *satisfies the following equation*

$$\vec{d}_{FCFS} = \vec{Z}(\vec{d}_{FCFS}), \quad (22)$$

where

$$\vec{Z}(\vec{d}_{FCFS}) = (Z_1(\vec{d}_{FCFS}), \ \ldots, \ Z_K(\vec{d}_{FCFS}))^{\top}_{K\times 1}. \quad (23)$$

- *Furthermore,* $\vec{d}_{FCFS}$ *is bounded by*

$$\|\vec{d}_{FCFS}\| \leq \frac{\sum_{i=1}^{N}\beta_i}{1-\nu}. \quad (24)$$

The proof of the theorem is given in [8]. Theorem 3 gives the criteria for stability and an equation for the worst case delay in an FCFS based ATM network with arbitrary topology. Equation (22) can be solved by using a simple iterative procedure. Let $\vec{d}^{[0]}_{FCFS}$ represent a vector at the beginning of the first iteration, and let $\vec{d}^{[n]}_{FCFS}$ the vector at the end of the n^{th} iteration. Before the first iteration, vector $\vec{d}^{[0]}_{FCFS}$ is initialized as follows.

$$\vec{d}^{[0]}_{FCFS} = (1, \ 1, \ \ldots, \ 1)^{\top}_{K\times 1}. \quad (25)$$

In the n^{th} iteration, $\vec{d}^{[n]}_{FCFS}$ is computed as follows.

$$\vec{d}^{[n]}_{FCFS} = \vec{Z}(\vec{d}^{[n-1]}_{FCFS}). \quad (26)$$

The question remaining is if $\vec{d}^{[n]}_{FCFS}$ converges to $\vec{d}_{FCFS}$. In order to demonstrate the convergence of the procedure we need to determine the error between $\vec{d}_{FCFS}$ and $\vec{d}^{[n]}_{FCFS}$, the vector at the end of the n^{th} connection. That is, for server j we need to establish the difference between the value of $d_{*,j}$ computed at the end of the n^{th} iteration and the real value of $d_{*,j}$. For the iteration procedure to converge, this difference must become zero for large value of n.

The next theorem gives an estimation of $\vec{d}_{FCFS}$ at the end of the n^{th} iteration.

Theorem 4 *For an FCFS based ATM network in which $\mu < 1$ and $\nu < 1$, if the iterative procedure defined by (25) and (26) is used to solve (22) then at the end of the n^{th} iteration*

$$\|\vec{d}_{FCFS} - \vec{d}^{n}_{FCFS}\| \leq \frac{(\nu)^n}{1-\nu} * \|\vec{d}^{[1]}_{FCFS} - \vec{d}^{[0]}_{FCFS}\|. \quad (27)$$

The proof of the theorem is given in [8].

Note in (27)

$$\lim_{n\to\infty} \frac{\nu^n}{1-\nu} = 0. \quad (28)$$

Therefore, as $n \to \infty$, the right hand side of (27) tends to 0. Hence, the iterative procedure converges.

4.2 Cruz-Gallager-Parekh Ring

In this subsection, we study ATM networks with a specialized ring topology. This topology has been used as an representative benchmark by Cruz, Gallager, and Parekh to study the problem of stability in ATM networks [1, 10]. Henceforth, we shall refer to this topology as the *Cruz-Gallager-Parekh* (C-G-P) ring.

The architecture of the C-G-P ring is described as follows. The system consists of K 2×2 switches and K connections, $M_1, M_2, \ldots, M_i, \ldots, M_K$. Each server has a distinct identity id, where $id = 1, 2, \ldots, 2*K$ and every connection has an acyclic path which traverses K servers. For connection M_i, $s(i,1) = i$ and

$$s(i,j) = \begin{cases} 1 + (i+j-2) \bmod K, & 1 \leq j \leq K-1, \\ K + i, & j = K. \end{cases} \quad (29)$$

Figure 1 shows an example of a C-G-P ring with 4 switches.

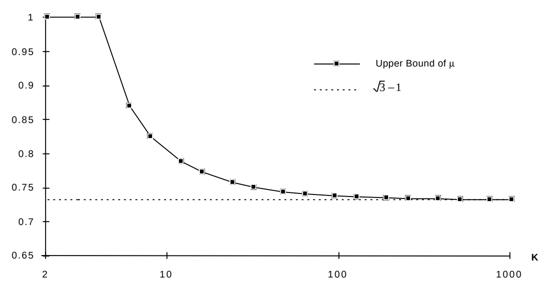

Figure 4: Plot of the upper bound of μ given by (29).

The source traffic in the C-G-P ring is constrained by a piecewise linear function. For connection M_i in the C-G-P ring, the traffic at the source is given by

$$F_{i,s(i,1)}(I) = \min(I, \beta_i + \rho * I), \tag{30}$$

where β_i and ρ are positive real-numbers. Note that the value of ρ is the same for all the connections.

The following theorem gives the criteria for stability in an FCFS based ATM network with a C-G-P ring topology.

Theorem 5 *A C-G-P ring with FCFS servers is stable if*

$$\mu < \begin{cases} 1, & \text{if } 2 \leq K \leq 4, \\ \sqrt{1 + \frac{2*(K-1)}{K-2}} - 1, & \text{if } K \geq 5. \end{cases} \tag{31}$$

The proof of the theorem is given in [8]. Figure 4 shows the plot of the upper bound of μ given by (31). We observe that as K is increased the upper bound of μ decreases. Further, for large values of K, the upper bound of μ converges to $\sqrt{3} - 1$. This observation is formalized in the following corollary.

Corollary 1 *An ATM network with the C-G-P topology and FCFS servers is stable if*

$$\mu < \sqrt{3} - 1 \approx 0.732. \tag{32}$$

The formal proof of the corollary is given in [8].

The results of Corollary 1 means that any ATM network with the C-G-P ring topology is stable if the maximum link utilization in the network is less than 73.2%. This is an efficient criteria to determine stability.

5 Priority Driven Scheduling

In this section, we study the stability problem in ATM networks with priority driven scheduling for the servers. We also establish delay bounds in such networks. In some ATM networks servers with priority driven scheduling policies are used to provide different levels of services. In priority driven scheduling, every connection traversing a server is assigned a priority. The server transmits the cells waiting in its queue in an order given by the priority of the connections associated with the cells. For example, if connection M_1 has a higher priority than connection M_2, then M_1's cell will always be transmitted before connection M_2's cell. Thus, the worst case cell delay of a connection at the server depends only on the traffic of the connections with a higher priority.

We first present the following notations which will help in the analysis of priority based ATM networks. Let $P_{i,j}$ be the priority assigned to connection M_i at server j. If $P_{i,j}$ is independent of time t, then the priority assignment is said to be a *static* one. A priority assignment is said to be *fixed* if for $j \neq j'$,

$$P_{i,j} = P_{i,j'}. \tag{33}$$

In this paper, we assume a *static and fixed priority* assignment. Given that the priority assignment is fixed, let P_i be the priority for M_i at all the servers. $\vec{P}$ is the priority assignment vector for the set of N connections. $\vec{P}$ is given by

$$\vec{P} = (P_1,\ P_2,\ \ldots,\ P_i,\ \ldots,\ P_N)^{\top}_{N\times 1}. \tag{34}$$

We further assume that the priority assigned to each connection is *globally distinct*, i.e., for $i \neq i'$

$$P_i \neq P_{i'}. \tag{35}$$

It must be noted that if the connection priorities in a static fixed system are not distinct then in the worst case the performance of the static fixed priority based system can reduce to that of a system using FCFS servers. In that case, the results presented in Section 4 are directly applicable. Without loss of generality, we assume that in the system of N connections, $M_1, M_2, \ldots, M_i, \ldots, M_N$, we have

$$P_1 > P_2 > \ldots > P_i > \ldots > P_N. \tag{36}$$

The following theorem, gives the criteria for stability and the end-to-end delay in ATM network with static, fixed, and globally distinct priority assignment (SFGDP).

Theorem 6 *For an ATM network with arbitrary topology and SFGDP based servers, if $\mu < 1$, then*

- *the system is stable and*
- *the delay of M_i at server $s(i,j)$, for $M_i \in \mathcal{M}$ and $1 \le j \le S_i$, is given by*

$$d_{i,s(i,j)} = \frac{\sum_{h=1}^{L_{s(i,j)}} \sum_{k=1}^{i-1} m_{k,s(i,j),h} * [\beta_k}{1 - \sum_{h=1}^{L_{s(i,j)}} \sum_{k=1}^{i-1} m_{k,s(i,j),h} * \rho_k}$$
$$\frac{+\rho_k * \sum_{g=1}^{j_k-1} d_{k,s(k,g)} + \rho_k * \frac{\beta_i+\rho_i*\sum_{g=1}^{j-1} d_{i,s(i,g)}}{1-\rho_i}]}{}. \quad (37)$$

The proof of Theorem 6 are not given here due to space limitations. An interested reader is referred to [8].

Theorem 6 establishes the criteria for stability and gives the expression for the worst case delay experienced by a connection in an ATM network with SFGDP servers. Because of our assumption (36), the delay of connection M_i at its j^{th} server given by equation (37) is dependent only on the delays experienced by connections $M_1, \ldots, M_{i-1}$ and the delays experienced at previous servers. Therefore, equation (37) can be solved in a sequential order for $i = 1$ then $i = 2$ and so on. For a given value of i, the delays at the servers are also computed in a sequential order, i.e., $j = 1$ then $j = 2$ and so on.

6 Extensions

Recall that the stability criteria established in the previous sections were based on the following two assumptions.

1. The source traffic description function of all the connections in the system is piecewise linear.
2. The scheduling policy used in the servers within the network is either FCFS or SFGDP.

In this section, we relax the above two assumptions and establish the stability criteria for a general ATM network. Specifically we extend our results to encompass systems in which the source traffic description function is not piecewise linear. We also extend our results to ATM networks with servers employing work conserving scheduling policies other than FCFS and SFGDP. The proofs are not given in this paper due to space limitation. An interested reader is referred to [8].

6.1 General Source Traffic

Recall that in Theorems 3, 5, and 6 we assumed that the source traffic of the connections were constrained by the piecewise linear traffic description function. Although the source traffic of many connections can be characterized by such piecewise linear functions, it is useful to establish the criteria for stability in a system without this constrain. The following theorem establishes the stability criteria for a general source traffic description function.

Theorem 7 *The criteria for stability given in Theorems 3 and 5 hold if the source traffic satisfies the following condition: $\forall i$, $M_i \in \mathcal{M}$, there are positive real numbers ρ_i, and T such that for $I > T$,*

$$F'_{i,s(i,1)}(I) \le \rho_i, \quad (38)$$

where $F'_{i,s(i,1)}(I)$ is the derivative of $F_{i,s(i,1)}(I)$ above the variable I.

This theorem says that as long as the long term average rate of the source traffic is bounded, all the results for stability established using the piecewise linear function are also applicable for the general source traffic function.

6.2 Work Conserving Scheduling

Here, we consider an ATM network which consists of servers using any work conserving scheduling policy. A server employing a work conserving scheduling policy always transmits a cell if its buffer is not empty. The FCFS and priority driven scheduling policies are examples of work conserving scheduling policy. In the following theorem we establish the stability criteria for an ATM network with servers employing a work conserving scheduling policy.

Theorem 8 *The criteria for stability given in Theorems 3 and 5 hold if the scheduling policy at the servers are work conserving.*

This theorem can be easily proved by observing that for a given source traffic the length of the maximum busy interval for any work conserving server is the same. Now if the system using FCFS servers is stable then the length of the maximum busy interval at the FCFS servers is bounded. Therefore, the length of the maximum busy interval at the servers is also bounded if the system were to have used some other work conserving scheduling policy at its servers. Further, since for any work conserving server the maximum queue length at the server is no more than the length of its maximum busy interval, the queue length at the work conserving servers is bounded when the length of its maximum busy interval is bounded. Therefore, when the system with FCFS servers is stable then the system with any other work conserving servers is also stable. Hence the criteria for stability given in Theorems 3, 5, and 6 hold if the scheduling policy at the servers are work conserving. Theorems 5 and 8 together generalize the claim made by Gallager and Parekh [10] that that any C-G-P ring of 4 switches with work conserving servers is stable when $\mu < 1$. Our result indicates that when the ring size is large (≥ 5), the network with any work conserving scheduling policy is stable if $\mu < \sqrt{3} - 1$.

7 Summary and Conclusions

In this paper we addressed the stability problem in ATM networks. We have focused on the development of criteria for testing the stability of an ATM network and the determination of the delay bounds in a stable network. The problem of stability in ATM networks was studied by many researchers [1, 10, 12] However, our work differs from the previous work by making the following contributions:

We introduced two important results to analyze ATM networks. The first result, presented in Theorem 1, allowed us to express a connection's traffic at the input of the server in terms of the source traffic of the connection as well as the delays suffered in the previous servers. The second result, presented in Theorem 2, allowed us to cahracterize the aggregate cell traffic over an ATM link, and utilize it to accurately analyze the input traffic at the servers.

For FCFS based networks with arbitrary topology, we develop the criteria for network stability and an iterative method to derive the delay bounds in a stable network. We show that for a stable network the iteration procedure converges. We also generalized the result of stability in an FCFS based ATM network to the one using any work conserving scheduling policy.

In previous work, the Cruz-Gallager-Parekh ring has been a "benchmark" architecture to study the stability problem. For example, Gallager and Parekh claimed a C-G-P ring is stable if the total number of switches is no more than 4 [10]. We validated this result. Furthermore, we found that a large size ring is stable if the total utilization of the links is less than or equal to 73%.

For ATM networks with priority driven scheduling policies we found that one class of priority assignment mechanism, namely a static fixed globally distinct priority assignment, will guarantee the stability as long as the utilization of the individual links is less than 100%.

We also showed that the main results on stability holds not only for piecewise linear source traffic model (as assumed in most previous work) but also for general source traffic as long its long term average rate exists.

This work can be extended in several ways. It would be ineteresting to consider the stability problem in connection based heterogeneous networks. To establish the criteria of stability in such networks it will be necessary to investigate characterizations of the traffic within the network. Utilizing a consistent traffic characterization function over a series of network segments is a key step in this process.

Acknowledgments

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, the Office of Naval Research, US Navy, under grant number N00014-95-1-0238, 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 Office of Naval Research, the U.S. Government, Texas Higher Education Coordinating Board, the Texas State Government, or Texas A&M University.

References

[1] R. L. Cruz. A calculus for network delay. *IEEE Trans. on Infor. Theory*, 37(1):114–131, Jan. 1991.

[2] R. L. Cruz. A calculus for network delay, part II: Network analysis. *IEEE Trans. on Infor. Theory*, 37(1), Jan. 1991.

[3] R. L. Cruz. Quality of service guarantees in virtual circuit switched networks. *IEEE Select. Areas Commun.*, 13(6), Aug. 1995.

[4] N. Dunford and J. Schwartz. Linear operators, Part I: General theory. *Interscience publishers.* New York (1958).

[5] S. J. Golestani. A framing strategy for congestion management. *IEEE J. on Select. Areas in Commun.*, 9(7), Sept. 1991.

[6] Charles R. Kalmanek and Hermant Kanakia and Srinivasan Keshav, Rate Controlled Servers for Very High-Speed Networks. *Proc. of IEEE Global Telecom. Conf.*, Dec. 1990.

[7] L. Kleinrock. Queueing Systems, volume II: Computer Applications , *Addison-Wesley*, 2rd, 1994.

[8] Chengzhi Li, Amitava Raha and Wei Zhao, Stability in ATM networks, *Department of Computer Science, Texas A&M University*, june, 1996.

[9] J. Liebeherr, D. Wrege and D. Ferrari. Exact admission control in networks with bounded delay services. *to appear in IEEE/ACM Trans. Networking*

[10] A. K. J. Parekh. *A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks.* PhD thesis, Department of Electrical Engineering and Computer Science, M.I.T., 1992.

[11] A. Parekh and R. Gallager. A generalized processor sharing approach to flow control in integrated services network: The multiple node case. *Proc. of the IEEE Infocom'93.*

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

[13] A. Raha, S. Kamat, and W. Zhao. Admission control for hard real-time connections in ATM LAN's. In *Proc. of the IEEE Infocom'96*, Mar. 1996.

[14] H. Zhang. Providing End-to-End Performance Guarantees Using Non-Work-Conserving Disciplines. *Computer Commu.: Special Issue on Sys. Supp. for Multimedia Computing* , 18(10),Oct. 1995.

[15] Hui Zhang and Domenico Ferrari. Rate-Controlled Static Priority Queueing. *Proc. of IEEE Infocom'93"*,Mar. 1993.