A Preemptive Protocol for Voice-Data Integration in Ring-based LAN: Performance Analysis and Comparison*†

Sarit Mukherjee‡
Dept. of Computer Science & Engg.
University of Nebraska
Lincoln, NE 68588

Debanjan Saha Satish K. Tripathi
Department of Computer Science
University of Maryland
College Park, MD 20742

Abstract

Recently we proposed a preemptive protocol handling multiple classes of traffic on a slotted ring LAN. In this paper we analyze the protocol for voice-data integration and present a comparative study of its performance with other well known ring LAN protocols. We derive approximate closed form expressions for response time distribution of both voice and data traffic in their transform forms. The analysis is validated through simulation. Computation of important performance metrics is discussed. Through extensive simulation we compare the perfomance of our protocol with token ring based FDDI, slotted ring bassed Cambridge Fast Ring (CFR), and register-insertion ring based TORNET2 for a wide range of workloads.

*This work is supported in part by contract DASG60-87-C-0066 from the U. S. Army Strategic Defense Command to the Department of Computer Science, University of Maryland. The views, opinions, and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of Defense position, policy, or decision, unless so designated by other official documentation. Computer facilities were in part provided by NSF grant CCR-8811954.

†The paper was presented in part at the 2nd International Conference on Computer Communications and Networks, June 1993.

‡The work was done when the author was a graduate student at the University of Maryland, College Park.

1 Introduction

One important design goal of high speed local/metropolitan area networks (LAN/MAN) is to support various quality of service requirements of different classes of traffic (e.g. data, voice, video) in an integrated environment. The LAN/MAN traffic can be broadly classified into two classes — isochronous (voice, video, etc.) and asynchronous (data). The quality of service requirements of isochronous traffic are low delay and delay variability (jitter), whereas, asynchronous traffic does not need any strict bound on delay. On the other hand, isochronous traffic can withstand some packet losses, which is intolerable for asynchronous traffic. Most of the proposed protocols supporting multiple classes of traffic in a LAN/MAN environment work on the basis of bandwidth reservation [4, 41, 3, 12, 30, 35]. Certain bandwidth is reserved for isochronous traffic and the rest is available to other classes. Packet-by-packet reservation increases the transmission latency of packets, resulting in a decrease in the throughput of the network [41]. Per-connection based allocation of bandwidth suffers from poor network utilization as bandwidth is allocated at the time of call setup, and the dynamic recovery of the unused bandwidth is very difficult [21, 5].

In [26] we presented a preemptive protocol for a slotted ring LAN[1] supporting multiple classes of traffic. The extension of the protocol to handle multiple classes of traffic in a hierarchically connected slotted ring MAN is discussed in [24]. The proposed protocol does away with the bandwidth (slot) reservation, provides distributed media access control, and efficiently adapts itself to traffic load fluctuations. Priorities are assigned to different classes of traffic. The lower priority stations always send as many packets as possible (without violating the fairness criteria of the bandwidth distribution) instead of letting a free slot go. But this does not affect the higher priority traffic as they can preempt a slot used by a lower priority station. Consequently, the utilization and the throughput increase without violating the quality of service requirements. In this paper we study the applicability and the performance of this protocol in a high speed LAN/MAN supporting both voice and data traffic.

Performance of ring protocols for voice-data integration has been widely studied [42, 16, 40, 28, 41, 7, 39]. The main problem faced in the analysis is the modeling of the voice source. A voice source behaves as having alternating talkspurts and silences, with packets generated at a constant rate during talkspurts only. Both talkspurts and silence periods can be approximated to be exponentially distributed [33]. Traditional queueing theoretic analysis becomes intractable for this arrival process. Mean delay analysis of each class of traffic (voice and data) has been performed in [16, 17, 40, 41, 32, 22]. However, this is not adequate since, for voice traffic, other important

[1] A slotted ring LAN consists of a number of stations connected to a ring in which one or more fixed length slots move in one particular direction. A station sends a packet to another by writing it into a free slot.

performance metrics (e.g., delay variability, clipping probability) become difficult to compute unless the distribution of the delay is available. Delay distributions for voice traffic have been obtained in some studies [7, 16], but their approximate models fail to capture the correlated behavior of a voice source. Poisson arrival has been assumed in [7, 39]. This model does not capture the on-off nature of the source. Poisson source with bulk arrival has been used to better approximate a voice source in [16, 41]. Since it cannot model the fixed inter-arrival time between voice packets, the delay seen by the first packet in a bulk is used to estimate the voice packet delay. Simulation studies have been carried out to study the performance metrics in [42, 28, 34].

In this paper we model the arrival stream from a voice source by a two-state Markov Modulated Poisson Process (MMPP) [27]. The traffic pattern from a voice station supporting multiple call connections can be approximated by a two-state MMPP with a high degree of accuracy [10]. A single voice source can also be approximated by a two-state MMPP with no arrivals in one state (silence period). The superposed source models arrival stream at a gateway handling multiple call connections in a MAN, whereas the single source model is appropriate in a LAN environment where an individual source can be connected to the network. Since standard queueing theoretic technique is hard to use for analyzing this kind of systems (which is shown later to be a MMPP/G/1 queue with server vacation), we use the matrix analytic method [20, 19]. We obtain a closed form expression for the approximate response time distribution of voice packets. A data source is modeled as a bulk Poisson process to reflect its burstiness and variable message length[2]. Its approximate response time distribution is derived using $M^{[X]}/G/1$ queue with server vacation [36]. We obtain closed form expressions for some important performance metrics, e.g., mean delay, delay jitter for voice traffic and mean delay of data messages. Extensive simulation is carried out to show that our analysis is valid for a wide range of workloads. We discuss the application of the analytical results in designing a LAN/MAN system.

We perform a comparative study of the protocol with slotted ring based Cambridge Fast Ring (CFR) [11], token ring based FDDI [30], and register-insertion ring [41] based TORNET2 [39] using discrete event simulation. Each of these protocols has been studied extensively in literature. However, to our knowledge, a comparative study of them for voice-data integration has not been reported. Our results show that for voice traffic TORNET2 and the premptive protocol show superior performance. Data traffic is best served by CFR and the performance of the preemptive protocol is very close to it.

The rest of the paper is organized as follows. In section 2, we describe the network topology and the protocol. Analytical models of voice and data sources and their service time distributions are

[2]Since superposition of several such sources produce traffic of the same characteristics, the analysis is valid for single as well as multiple data sources connected to a station.

discussed in sections 3 and 4, respectively. The response time distribution for each class of traffic is derived in section 5. In section 6, we show the computation of several important performance metrics. The analytical results are validated through simulation in section 7. In section 8 we show the application of the analysis in designing and tuning a LAN/MAN. The comparative study with CFR, FDDI and TORNET2 is presented in section 9. The paper is concluded in section 10.

2 Topology and Protocol

A slotted-ring network [9, 15] is composed of a ring in which bits flow circularly in one direction. The stations (or nodes) are connected to the ring, and bits are transferred from one station to another around the ring. The number of bits that can circulate in the ring depends on the ring latency and the aggregate delay produced by the stations. The bit stream is divided into a constant number of fixed length slots. Stations communicate by writing fixed size packets on the slots. The protocol for the ring handling multiple classes of traffic is described in [26]. Here we describe it for two classes of traffic — voice and data, where voice traffic has preemptive priority over data traffic. A prototype ring with eight stations is shown in figure 1. Each slot of the ring is formated in the

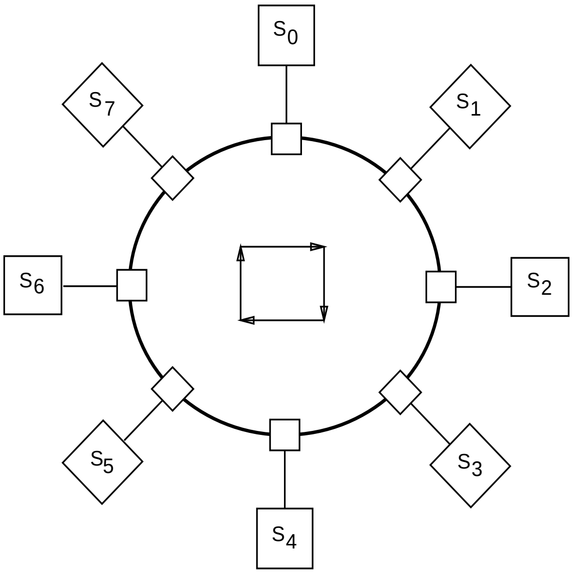

Figure 1: The slotted-ring architecture

way shown in figure 2. The V/D bit is used to check whether the slot is used by voice or data traffic. The F/E bit is used by data traffic to check if the slot is full or empty. The Source and Dest fields contain the addresses of the source and the destination of the packet, respectively. The Data field can hold a packet produced by any class of traffic, and the Ack bit is used to acknowledge

V/D	F/E	Source	Dest	Data	Ack

Figure 2: Format of a slot

the receipt of a data packet. Note that data packets are acknowledged because data traffic cannot tolerate any packet loss.

Packets arrive from external sources to the stations. Each station has separate buffers to store different classes of traffic. For the rest of the discussion, we call them V-station and D-station for voice traffic and data traffic, respectively[3]. A station gets a slot at regular intervals. It uses a slot to transfer a packet to any destination station in its class.

A slot is considered free for a V-station if the V/D bit is reset. The V-station writes the packet into an empty slot and sets the V/D bit. In the process it may overwrite any data which has already been written by a D-station. It also fills up source and destination fields of the slot. It cannot use another slot while waiting for the return of this slot. Since there is a constant number of slots, a station can identify the slot it used by counting the number of slots passed by. When a V-station receives a slot destined to it, it reads the data. A full slot is emptied by the source V-station (known as *source removal*) by resetting the V/D bit. In order to prevent *hogging*, the station is not allowed to use the same slot.

For data traffic, a slot is considered free if both V/D and F/E are reset. The station sets the F/E bit, writes data in it, fills out source and destination fields, and waits for the slot to return. It cannot use any other slot till it gets back the one it used, i.e., a D-station, like a V-station, can have only one outstanding packet on the ring. A destination D-station reads a full slot (V/D reset, F/E set) and sets the Ack bit. Note that the same slot is used for acknowledging the receipt. Since voice traffic does not modify the Ack, the acknowledgement is never overwritten. When the slot returns, the source station resets the F/E bit and checks if the slot has been successfully read by the destination station by examining the Ack bit. If the bit is not set[4], the packet is retransmitted using the same strategy. The Ack bit is also reset by this station. The station is not allowed to use the same slot.

In this protocol D-stations always try to send as many packets as possible, without violating the fairness of slot distribution, instead of letting a free slot go. However, this does not affect the voice traffic as they can reuse a slot used by a D-station. Consequently, utilization and throughput

[3]Note that a station could be a V-station, a D-station, or both.

[4]The slot has been overwritten by an intermediate voice source before it reaches its destination.

increase without affecting the quality of service requirements of the voice traffic. Although successful transmissions of data packets depend on the actual location of the active voice stations, the protocol gives fair share of slots to traffic within a particular class. Furthermore, data traffic can effectively utilize the bandwidth when voice sources become quiescent (silence period).

The protocol provides priority-based, distributed and adaptive control of slot usage among the stations. Since V/D bit can override F/E bit, higher priority to V-stations is ensured. The usability of a slot can be determined locally by examining the slot header. This achieves distributed control over the slots. The availability of slots depends on the loads of the stations. When V-stations become heavily loaded, they get a large share of the slots. When they are in low load condition, D-stations get free slots more quickly.

3 Modeling of Sources

For ease of analysis, we assume that arrival processes at each station are stochastically identical[5]. In the rest of the section we describe the modeling of the arrival processes for voice and data sources.

3.1 Voice Source

A voice source alternates between talk spurts and silence periods (see figure 3). The packet stream

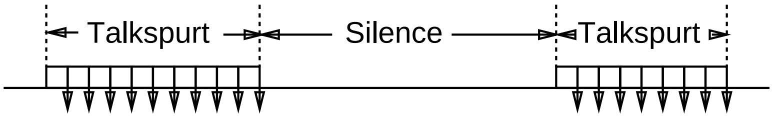

Figure 3: A voice source

from a single voice source is characterized by arrivals at fixed intervals during talkspurts and no arrivals during silence periods. Sriram and Whitt [33] have shown that the durations of talk spurts and silence periods can be well-approximated by exponential distribution with means of 352 ms and 650 ms, respectively. Inter-arrival time between packets (T_{pkt}) during talkspurt is constant and depends on the voice coding rate. Thus the arrival process forms a discrete time arrival of packets at T_{pkt} apart for exponentially distributed amount of time, and then a period of no packet arrival. We assume that a voice station supports multiple call connections. Hence it can be characterized by the superposition of voice sources. Heffes and Lucantoni [10] have shown that the superposition of voice sources can be modeled by a two-state Markov Modulated Poisson Process(MMPP) [27]

[5]This assumption can be relaxed. However, we do not discuss it in this paper.

with high degree of accuracy. This model preserves the correlated nature of the voice traffic. In the following analysis, we model the arrival process at a voice station by a two-state MMPP. In the special case, when a voice station supports one-call connection, the arrival process can still be approximated by a two-state MMPP, with no packet arrival in one state to model the silence period[6].

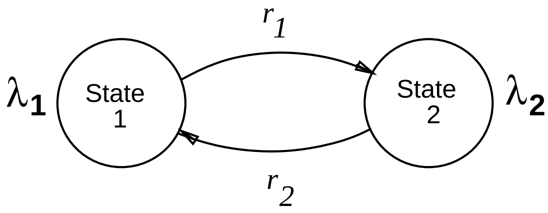

Figure 4: Superposed voice source model

In the MMPP approximation of voice packet arrival pattern (figure 4), the underlying modulating process is a two state Poisson process. Sojourn time in each state is exponentially distributed with means $1/r_1$ and $1/r_2$, respectively. Packet arrival patterns in either state of the modulating process follow Poisson distributions with different rates. We assume that arrival rates are λ_1 and λ_2, respectively. The parameters $r_1, r_2, \lambda_1, \lambda_2$ can be computed from the known parameters of the voice sources [10]. This superposed arrival process can be fully characterized by the matrices $\mathbf{R}$ and $\mathbf{\Lambda}$, where $\mathbf{R}$ is the generator matrix of the modulating Markov Process and $\mathbf{\Lambda}$ is the rate matrix. In our model they take the following forms,

$$\mathbf{R} = \begin{bmatrix} -r_1 & r_1 \\ r_2 & -r_2 \end{bmatrix}, \quad \mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}.$$

3.2 Data Source

Data sources are bursty in nature and can generate different sizes of messages (figure 5). In a slotted environment, a message is divided into a number of fixed size packets that can fit in a slot. We assume that a source generates messages at exponentially distributed intervals with mean $1/\lambda_d$. Each message consists of geometrically distributed number of packets[7] with a mean of $1/p_d$. Thus, the arrival process at a D-station can be characterized as bulk arrival of packets with exponentially distributed inter-arrival time between bulks (with mean $1/\lambda_d$). The bulk size, B, is geometrically

[6] This model over-estimates the variability of an actual voice source.

[7] It could be any other distribution, geometric assumption is made for simplicity.

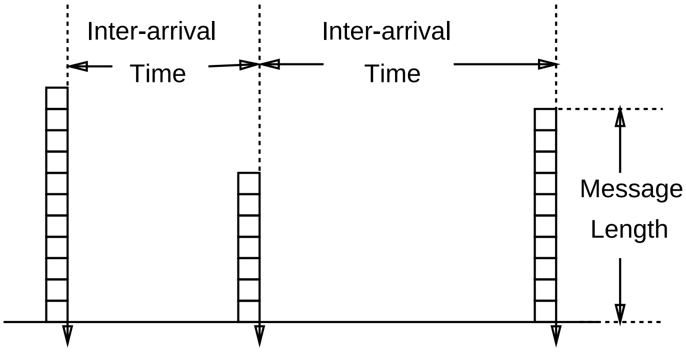

Figure 5: A data source

distributed with average bulk size $1/p_d$. The pdf and pgf of B are,

$$Prob(B = i) = p_d(1 - p_d)^{i-1}, \quad i = 1, 2, \ldots, \infty,$$

$$G_B(z) = \frac{p_d z}{1 - (1 - p_d)z}. \tag{1}$$

4 Modeling of Service Time

In the ring, slots are the unit servers that carry packets from one station to another. From a station's perspective, slots visit it at discrete time intervals that are x time units apart, where x is the slot transmission time[8]. Service is available only at the instants when a slot arrives. Therefore, the servers are non-work-conserving and they take a vacation of duration x when the queue is empty. We use server vacation [37] to model this phenomenon. The service time of a packet is defined as the aggregate of the time to get a free slot (since the packet is at the head of the queue) and the time to write a packet into a slot. The rate of service depends on the probability of finding a free slot. In the rest of the section, we derive the service time distributions of different classes of traffic. We assume that there are N stations (each station can handle both voice and data) connected to the ring and the ring latency is R time units.

4.1 Voice Traffic

For a V-station, the probability of arrival of a full slot is the same as the utilization, U_v, of a slot by the voice traffic. U_v can be approximated by the product of aggregate voice packet arrival rate as seen by a slot and the time required for a slot to flow from one station to the next one [2, 18] (which is simply $(x + R/N)$, assuming stations are uniformly distributed on the ring). The aggregate voice

[8]This implies that a slot takes x time units to pass a station.

packet arrival process seen by a slot is the superposition of N independent MMPP sources. It can be shown that the superposition of N identical two-state MMPP sources is an $(N+1)$ state MMPP source [27]. After some algebraic manipulation, we get the average rate of the superposed process as, $\lambda_v N$, where $\lambda_v = (r_2\lambda_1 + r_1\lambda_2)/(r_1 + r_2)$ is the fundamental arrival rate from a voice source. Hence,

$$U_v = \lambda_v N(x + R/N).$$

The service time of a packet is, $S_v = K_v x$, where the random variable K_v counts the total number of slots passing by the station up to and including the arrival of a free slot. Assuming independence among slots, K_v follows a modified geometric distribution with parameter U_v. Now the service time distribution and its Laplace-Stieltjes transform can be obtained as,

$$Prob(S_v = kx) = (1 - U_v)U_v^{k-1}, \; k = 1, 2, \ldots, \infty,$$

$$L_{S_v}(s) = \frac{(1 - U_v)e^{-xs}}{1 - U_v e^{-xs}}. \tag{2}$$

The traffic intensity at a voice station is, $\rho_v = \lambda_v \overline{S_v}$, where $\overline{S_v}$ is the average service time of a voice packet. For stability of the V-station, $\rho_v < 1$ should be maintained.

4.2 Data Traffic

The utilization of a slot by data traffic, U_d, can be approximated using the same argument as used for voice traffic, with the additional constraint that the slot is not used by voice traffic. The superposed data packet arrival rate from N sources is $\overline{B}\lambda_d N$, and the probability that a slot is not used by voice traffic is $(1 - U_v)$. Thus,

$$U_d = \overline{B}\lambda_d N(x + R/N)(1 - U_v).$$

A slot cannot be used by a D-station if it is used either by a V-station or by a D-station. So the probability that a slot cannot be used by a D-station is, $U_{vd} = U_v + U_d$.

A data packet may have to be transmitted more than once since it may get overwritten by a V-station. Each packet transmission consists of the following operations: the station waits for an empty slot, writes the packet into the slot and waits one round-trip time (R) for the acknowledgement to return. If the source station does not get the acknowledgement, it restarts another such sequence of operations. Let X be the random variable denoting the time required to complete one transmission. Then $X = K_d x + R$, where the random variable K_d counts the total number of slots passing by the station up to and including the arrival of a free slot. The pdf of the random variable K_d and its pgf are as follows,

$$Prob(K_d = k) = U_{vd}^{k-1}(1 - U_{vd}), \quad k = 1, 2, \ldots, \infty,$$

$$G_{K_d}(z) = \frac{(1-U_{vd})z}{1-U_{vd}z}.$$

So the Laplace-Stieltjes transform of X is

$$L_X(s) = \frac{(1-U_{vd})e^{-(x+R)s}}{1-U_{vd}e^{-xs}}.$$

Suppose that a D-station D_i sends a slot to a D-station D_j. There are $n = (j-i) \pmod N$ number of V-stations in between them. Assuming that a destination station is chosen randomly, n is uniformly distributed in $[1, N-1]$. If the slot sent by D_i is overwritten by any one of the n V-stations, then D_j will not be able to read that slot. Also note that it could be overwritten only once by any one of the n V-stations. The probability that a V-station overwrites is the probability that the V-station is busy, which is ρ_v. The probability that the slot will be overwritten by the k^{th} V-station starting from V_i is $(1-\rho_v)^{k-1}\rho_v, 1 \le k \le n$. Assuming all the stations are independent in operation, the conditional probability that a slot will be overwritten by any one of the n V-stations is $\sum_{k=1}^{n}(1-\rho_v)^{k-1}\rho_v$. So the unconditional probability of an overwritten slot is

$$p_{ow} = \frac{1}{N-1}\sum_{n=1}^{N-1}\sum_{k=1}^{n}(1-\rho_v)^{k-1}\rho_v = 1 - \frac{(1-\rho_v)\{1-(1-\rho_v)^{N-1}\}}{(N-1)\rho_v}.$$

Let L_p be the random variable that i transmissions will be required to send a packet. Since each transmission is independent, L_p is geometrically distributed with parameter p_{ow} and the pgf of L_p is

$$G_{L_p}(z) = \frac{(1-p_{ow})z}{1-p_{ow}z}. \tag{3}$$

The service time S_d can be written as, $S_d = X_1 + X_2 + \cdots + X_{L_p}$, where X_i's are iids with the transform $L_X(s)$. This yields the Laplace-Stieltjes transform of the distribution function of S_d as,

$$L_{S_d}(s) = G_{L_p}(L_X(s)) = \frac{(1-p_{ow})(1-U_{vd})e^{-(x+R)s}}{(1-U_{vd}e^{-xs}) - (1-U_{vd})p_{ow}e^{-(x+R)s}}. \tag{4}$$

Note that for stability a D-station should have $\rho_d = \lambda_d \overline{B}\ \overline{S_d} < 1$.

5 Derivation of Response Time Distribution

The response time of a packet of any class consists of two parts, namely the waiting time and the service time. We define the waiting time of a packet as the time a packet takes to reach the head of the queue at the station starting from its arrival instant. Service time is defined as the aggregate of the time to obtain a free slot and write into it (and to be received successfully for data traffic), once the packet is at the head of the queue. The Laplace-Stieltjes transforms of the service time distributions for both classes of traffic have been computed in the previous section. In this section, we derive the waiting time distribution for each class of traffic and hence obtain the response time distributions. For details, the reader is referred to [23].

As explained earlier, the queueing system for the stations is non-work-conserving and can be modeled by server vacation. The derivation of the response time distribution is carried out in the following fashion. The Laplace-Stieltjes transforms of the waiting time distribution of a packet for an equivalent queueing system without server vacation and the vacation time distribution are computed separately. The final expression of the transform of the waiting time distribution of a packet is then obtained by multiplying those two transforms since the arrivals are independent [6]. The response time distribution is computed by multiplying the transforms of waiting time and service time distributions. Matrix analytic method and standard queueing theoretic approach are used to derive the response time distributions of the voice and data traffic, respectively. From the response time distribution, the transfer time[9] distribution can be computed by adding the propagation delay to the response time. In the rest of the analysis, we assume that the behavior of the stations are independent of one another (this underestimates the actual response time) and they have infinite buffer capacity (we do not consider buffer overflow in this paper).

5.1 Voice Traffic

We follow the matrix analytic technique described in [19] to derive the waiting time distribution of voice traffic. We first obtain the waiting time distribution of the MMPP/G/1 queue without server vacation. Let $\mathbf{W}(x) = [W_1(x), W_2(x)]$, where $W_i(x)$ is the joint probability that at any arbitrary time the voice source is in state i and a virtual arrival at that instant sees the unfinished work of at most x. Using the results obtained by Ramaswami [29], it can be shown that the Laplace-Stieltjes transform, $\mathbf{L_W}(s)$ of $\mathbf{W}(x)$, satisfies the following equation,

$$\mathbf{L_W}(s) = \begin{cases} s(1-\rho_v)\mathbf{g}[s\mathbf{I} + \mathbf{R} - \mathbf{\Lambda} + \mathbf{\Lambda} L_{S_v}(s)]^{-1} & s > 0, \\ \boldsymbol{\pi} & s = 0. \end{cases}$$

In the above equation, $\boldsymbol{\pi} = [r_2/(r_1+r_2), r_1/(r_1+r_2)]$, is the steady state probability vector of the modulating Markov process, and $\mathbf{g} = [g_1, g_2]$ is the invariant probability vector of the stochastic matrix $\mathbf{G}$ satisfying the following equation,

$$\mathbf{G} = \int_0^\infty e^{(\mathbf{R} - \mathbf{\Lambda} + \mathbf{\Lambda}\mathbf{G})x} dL_{S_v}(x).$$

[9]Transfer time of a packet is defined as the time taken to reach the destination starting from its arrival instant.

In general $\mathbf{G}$ cannot be obtained analytically from the above equation. However, in the special case when $\mathbf{G}$ is a 2×2 matrix, we can solve for $\mathbf{G}$ and obtain its steady state vector $\mathbf{g}$. Omitting some details and without loss of generality, assuming $\lambda_1 \geq \lambda_2$, we get,

$$\mathbf{g} = [\frac{r_2}{(r_1 + r_2) + (\lambda_1 - \lambda_2)y}, \quad \frac{r_1 + (\lambda_1 - \lambda_2)y}{(r_1 + r_2) + (\lambda_1 - \lambda_2)y}],$$

where y satisfies the equation, $y = 1 - \dfrac{r_2 y}{r_1 + (\lambda_1 - \lambda_2)y} - L_{S_v}(r_1 + r_2 + \lambda_1 y + \dfrac{\lambda_2 r_2 y}{r_1 + (\lambda_1 - \lambda_2)y})$.

After some tedious algebra we find $\mathbf{L}_W(s)$ to be

$$\mathbf{L_W}(s) = \frac{s(1 - \rho_v)}{D}\left[g_1(s + \lambda_2(L_{S_v}(s) - 1)) - r_2, \quad g_2(s + \lambda_1(L_{S_v}(s) - 1)) - r_1\right],$$

where $D = [s - r_1 + \lambda_1(L_{S_v}(s) - 1)][s - r_2 + \lambda_2(L_{S_v}(s) - 1)] - r_1 r_2$.

$\mathbf{L_W}(s)\mathbf{e}$ gives the transform of the waiting time distribution at any arbitrary time, where $\mathbf{e}$ is a unit column vector. The Laplace-Stieltjes transform of the waiting time distribution as seen by an arriving packet can now be expressed as

$$L_{W_v}(s) = \frac{\mathbf{L_W}(s)\boldsymbol{\lambda}}{\boldsymbol{\pi}\boldsymbol{\lambda}}, \quad \text{where } \boldsymbol{\lambda} = \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix}.$$

Note that this waiting time distribution is for a work-conserving server. But, as mentioned earlier, in our protocol the server is of non-work-conserving type. It takes a vacation, V_v, of duration x if the V-station queue is empty. The Laplace-Stieltjes transform of the distribution of the forward recurrence time of a vacation period is $L_{V_v}(s) = e^{-xs}$. From [20] we get the Laplace-Stieltjes transform of the waiting time of a packet as $L_{W_v}(s)[1 - L_{V_v}(s)]/(sx)$. Now the Laplace-Stieltjes transform of the response time (R_v) distribution can be written as

$$L_{R_v}(s) = \left[\frac{1 - L_{V_v}(s)}{sx}\right]\left[\frac{s(1 - \rho_v)[(g_1\lambda_1 + g_2\lambda_2)s + \lambda_1\lambda_2(L_{S_v}(s) - 1) - r_1\lambda_2 - r_2\lambda_1]}{\lambda_v([s - r_1 + \lambda_1(L_{S_v}(s) - 1)][s - r_2 + \lambda_2(L_{S_v}(s) - 1)] - r_1 r_2)}\right] L_{S_v}(s). \tag{5}$$

5.2 Data Traffic

We derive the response time of a packet in a message and then compute the response time distribution of a message. The response time of a packet consists of (see figure 6) the waiting time of the bulk that it belongs to, the waiting time of the packet within the bulk, and the service time (which includes the time to get a free slot). We briefly describe the derivation of each component.

The waiting time of a bulk in the queue is the same as the waiting time of a customer whose service time distribution is $G_B(L_{S_d}(s))$, which is $s(1 - \rho_d)/[s - \lambda_d + \lambda_d G_B(L_{S_d}(s))]$, using the

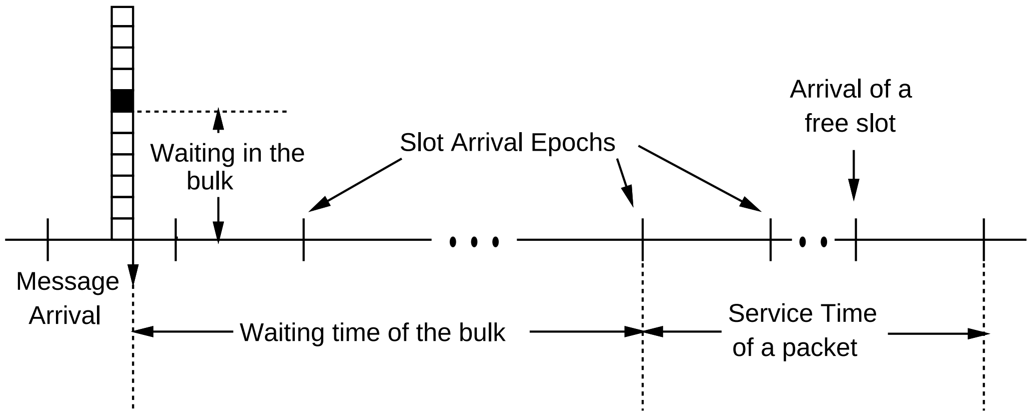

Figure 6: Response time of a data packet

standard results of M/G/1 queue. The waiting time of a packet within a bulk can be expressed as [37], $[1 - G_B(L_{S_d}(s))]/[\overline{B}(1 - L_{S_d}(s))]$. Since the server takes vacation V_d of duration x when the D-station is empty, the Laplace-Stieltjes transform of the distribution of the forward recurrence time of vacation is $(1 - L_{V_d}(s))/(sx)$, where $L_{V_d}(s) = e^{-xs}$. Now using [6] the distribution of the response time (R_d) of a packet can be expressed as

$$L_{R_d}(s) = \left[\frac{1 - L_{V_d}(s)}{sx}\right]\left[\frac{s(1-\rho_d)}{s - \lambda_d + \lambda_d G_B(L_{S_d}(s))}\right]\left[\frac{1 - G_B(L_{S_d}(s))}{\overline{B}(1 - L_{S_d}(s))}\right] L_{S_d}(s). \tag{6}$$

The response time distribution of a message can now be computed in the following way. Let W_m denote the waiting time of a message and W_b denote the waiting time of the first packet of the message. Then the waiting time of a message can be defined as the waiting time of the last packet of the message, which is,

$$W_m = W_b + S_{d_1} + S_{d_2} + \cdots + S_{d_{B-1}},$$

where S_{d_i} is the service time of a packet and B is the number of packets in the message. The Laplace-Stieltjes transform of W_b can be obtained as,

$$L_{W_b}(s) = \frac{s(1-\rho_d)}{s - \lambda_d + \lambda_d G_B(L_{S_d}(s))}.$$

We define $L = B - 1$. The pgf of L is $G_L(z) = G_B(z)/z$, and hence the waiting time distribution of a message is $L_{W_m}(s) = L_{W_b}(s) G_L(L_{S_d}(s))$. Therefore, the Laplace-Stieltjes transform of the response time distribution (R_m) of a message can be written as

$$L_{R_m}(s) = \left[\frac{1 - L_{V_d}(s)}{sx}\right]\left[\frac{s(1-\rho_d)}{s - \lambda_d + \lambda_d G_B(L_{S_d}(s))}\right] G_B(L_{S_d}(s)). \tag{7}$$

For geometric message size distribution, $L_{R_m}(s)$ is the same as $L_{R_d}(s)$ [8]. However, this analysis can be used for other distributions of message size as well.

6 Performance Metrics

6.1 Voice Traffic

From the response time distribution, the mean delay and the delay jitter (depends on the second moment) can be derived by successive differentiation. After some tedious algebra, the expressions for the first two moments can be written as

$$\overline{R_v} = \frac{(1-\rho_v)(g_1\lambda_1 + g_2\lambda_2 - \lambda_1\lambda_2\overline{S_v}) - \lambda_v(1-\lambda_1\overline{S_v})(1-\lambda_2\overline{S_v})}{(r_1+r_2)(1-\rho_v)\lambda_v} + \frac{(1+C_{S_v}^2)\rho_v^2}{2(1-\rho_v)\lambda_v} + \frac{x}{2} + \overline{S_v},$$

$$\begin{aligned}\overline{R_v^2} \;=\; & \frac{(1+C_{S_v}^2)^2\overline{S_v}^2\rho_v^2}{2(1-\rho_v)^2} + \frac{\lambda_v\overline{S_v^3}}{3(1-\rho_v)} - \frac{\lambda_1\lambda_2\overline{S_v^2}}{\lambda_v(r_1+r_2)} + \frac{(g_1\lambda_1 + g_2\lambda_2 + \lambda_1 + \lambda_2 - 3\lambda_1\lambda_2\overline{S_v})\overline{S_v^2}}{(r_1+r_2)(1-\rho_v)} \\ & + \frac{2(1-\lambda_1\overline{S_v})(1-\lambda_2\overline{S_v})}{\lambda_v(r_1+r_2)^2(1-\rho_v)^2}[\lambda_v(1-\lambda_1\overline{S_v})(1-\lambda_2\overline{S_v}) - \rho_v^2(1+C_{S_v}^2)(r_1+r_2) \\ & -(g_1\lambda_1 + g_2\lambda_2 - \lambda_1\lambda_2\overline{S_v})(1-\rho_v)] + \frac{x^2}{3} + \overline{S_v^2},\end{aligned}$$

where C_{S_v} is the coefficient of variation of the service time. Assuming that a destination is chosen randomly, and the ring latency, on an average, is $R/2$. Therefore, the average transfer time of a voice packet is

$$\overline{T_v} = \overline{R_v} + \frac{R}{2}.$$

The variance of the response time gives the jitter for n superposed voice sources at a station. It quantifies the variability in the inter-arrival time of consecutive voice packets at the receiver end. This quantity alone is not sufficient as the jitter for a particular call connection can be different from the value of the superposed source. The per-connection jitter is important to assure proper quality of voice reproduction at the receiver. For a connection, say the i^{th} packet departs the source station at time t_i and the $(i+1)^{th}$ packet leaves at t_{i+1}. We define $T_{i,i+1}$ as the inter-arrival time between the i^{th} and the $(i+1)^{th}$ packets of the same connection at the receiver. Note that the $(i+1)^{th}$ packet joins the source station T_{pkt} time after the i^{th} packet. Here we derive the distribution of $T_{i,i+1}$. Since $\pi'_j T_{pkt}$ is the amount of time spent in state j when the arrival rate is λ'_j, $(j=1,2)$, the pgf of the number packets that join the queue in T_{pkt}, K, is

$$G_K(z) = e^{\pi'_1\lambda'_1 T_{pkt}(z-1)} \times e^{\pi'_2\lambda'_2 T_{pkt}(z-1)},$$

where π'_j and λ'_j, $(j=1,2)$ are the parameters of $(n-1)$ superposed voice sources. The Laplace-Stieltjes transform of $T_{i,i+1}$ is

$$L_{T_{i,i+1}}(s) = G_K(L_{S_v}(s)) = e^{(\pi'_1\lambda'_1 + \pi'_2\lambda'_2)T_{pkt}(L_{S_v}(s)-1)}.$$

This gives the estimate of the per-connection-jitter distribution. Note that the ring propagation time is not included in the distribution since it is constant and same for all the packets of the connection.

To preserve the quality of service requirements of voice traffic, voice packets must be delivered within some specific time bounds, which is typically 150-200 ms [34]. The clipping probability of a voice packet can be defined as the probability that its transfer time overshoots the delay bound. Since in the case of a ring, the propagation delay from the source to the destination is fixed, the variable factor is the response time in the station. A packet will be clipped if ($R_v > D_{max}$), where D_{max} is the delay bound minus the propagation delay between the source-destination pair. Thus the probability of clipping is $Prob(R_v > D_{max})$ and can be obtained by inverting $(1 - L_{R_v}(s))/s$ at D_{max}. Analytical inversion of this expression is difficult, to say the least. However, it can be inverted numerically [1]. Usually 0.5% to 2% voice packet loss does not deteriorate the regenerated voice quality at the receiver end.

6.2 Data Traffic

The average delay in delivery of a packet can be computed from its distribution (equation (6)). The mean response time can be expressed as

$$\overline{R_d} = \frac{\lambda_d \overline{B}\ \overline{S_d^2}}{2(1-\rho_d)} + \frac{\overline{B^2}\ \overline{S_d}}{2\overline{B}(1-\rho_d)} + \frac{x}{2} + \overline{S_d}.$$

Assuming that a destination is chosen randomly and the ring latency, on an average, is $R/2$. Therefore, the average transfer time of a data packet is

$$\overline{T_d} = \overline{R_d} + \frac{R}{2}.$$

Similar quantities for a message are the following,

$$\overline{R_m} = \frac{\lambda_d[\overline{B^2}\ \overline{S_d}^2 + \overline{B}\ \mathrm{Var}(S_d)]}{2(1-\rho_d)} + \overline{B}\ \overline{S_d} + \frac{x}{2}, \quad \text{and} \quad \overline{T_m} = \overline{R_m} + \frac{R}{2}.$$

In equation (3) we have derived the distribution of L_p, the number of transmissions required to deliver a data packet. The number of transmissions required to deliver a message can be written as $L_m = L_{p_1} + L_{p_2} + \cdots + L_{p_B}$, where B is the number of packets in the message. The pgf of the distribution of L_m can be written as $G_{L_m}(z) = G_B(G_{L_p}(z))$. When bulk size is geometrically distributed with mean $1/p_d$, we get

$$G_{L_m}(z) = \frac{p_d(1-p_{ow})z}{1 - zp_d(1-p_{ow})z}.$$

7 Validation and Numerical Results

A prototype ring running the preemptive protocol was simulated using *csim*[10] [31]. The system simulated has the following parameters: there are 20 stations each carrying both voice and data traffic, four slots (of size 32 bytes [11]) in the ring with a total ring delay of 12.8 time units[11]. A voice station is modeled as a two-state MMPP source with different packet arrival rates in each state. A data station is modeled as a bulk Poisson source with geometrically distributed bulk size. The simulation models the actual flow of slots in the ring (not the geometric service time distribution as approximated in the analysis). We validate our analysis by varying the load of the voice and data traffic. The load of the voice traffic is varied by changing the arrival rate of packets at a particular state of the MMPP and the mean sojourn time in that state. The data traffic load is varied by changing the bulk inter-arrival time and the mean bulk size. The results involving time are normalized to the time required to transmit a slot (which is 3.2 time units). All the graphs are plotted against the percentage of load offered by all the stations to the network. The percentage is computed as the fraction of the total load offered by all the stations to the transmission capacity of the network (which is taken to be 80 Mb/s). Simulation results are obtained at 95% confidence level with the size of the confidence interval less than 2% of the mean.

Figures 7 (a) and (b) show the mean transfer time and the standard deviation of the response time of voice packets against percentage of voice traffic load on the ring. The load due to data traffic cannot affect the performance of the voice traffic since the protocol gives preemptive priority to the voice traffic. Hence, the data traffic load is immaterial for these graphs. The analysis and simulation show very close agreement (the maximum difference between simulation and analysis is less than 2%).

In figure 8 we plot the mean transfer time of data packets, with the load of the voice traffic as a parameter. The data load is varied by changing the mean inter-arrival time of bulks and the mean bulk size. The load offered by voice traffic is kept at fixed levels for each of these graphs. The mean bulk size is varied across figure 8 (a) and figures 8 (b) and (c), with the former being twice of the latters. Hence, the mean transfer time of packets in figure 8 (a) is more than that of figures 8 (b) and (c), even with lower voice traffic load. At low voice traffic load (figure 8 (b)), the analytical and simulation results show very close agreement. At higher voice traffic load (figure 8 (c)), the inter-arrival time of free slots at a station becomes correlated and the analysis underestimates the simulation. The difference between the simulation and analytical results widens with an increase

[10] *csim* is a simulation package distributed by Microelectronics and Computer Technology Corporation (MCC).

[11] The actual unit of time is not important here as we verify the analytical results normalized to the time needed to transmit a slot.

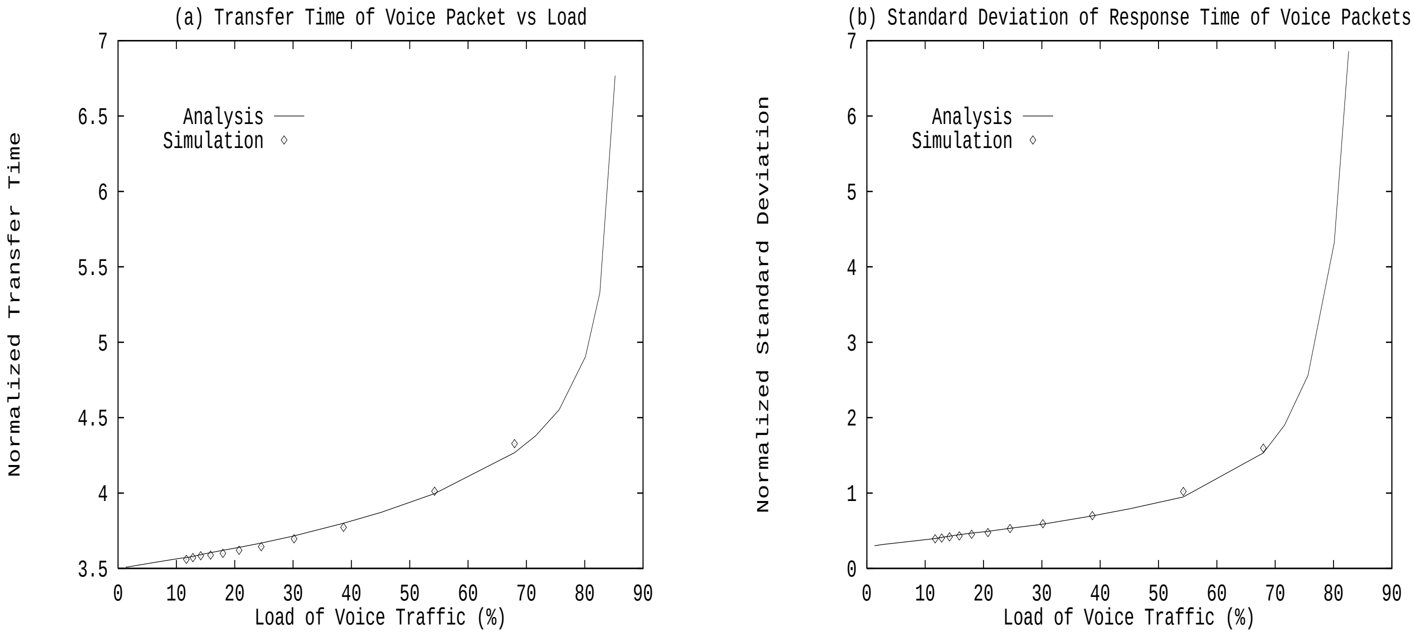

Figure 7: (a) Transfer time of voice packets (b) Standard deviation of response time of voice packets vs Load of voice traffic

in bulk size. This is because our estimate of slot utilization (U_d) treats all slots uniformly and does not capture the fact that all the packets belonging to a particular bulk go to the same destination, and thus creates a "deterministic" pattern of slot flow in the ring. This phenomenon becomes more prominent with the increase in bulk size. This explains the larger difference between analytical and simulation results in figure 8 (a) than in figures 8 (b) and (c). The maximum difference between simulation and analytical results for data traffic is less than 2.5%.

8 Application

The performance metrics derived in section 6 can be used to quantify the "behavior" of a practical system analytically. The need is further felt for big networks that are very expensive to study through simulation. Knowing the physical parameters of the network (e.g., the length of the cable, transmission rate, number of stations to be connected, statistical characteristics of the sources), the moments of response times for voice and data traffic, clipping probability for voice traffic, etc. can be computed. These measures help determine whether the system can satisfy its specific quality of service requirements (e.g. maximum delay for voice packets should be within some specified bound). They are also useful in designing new systems as well as in tuning and improving the

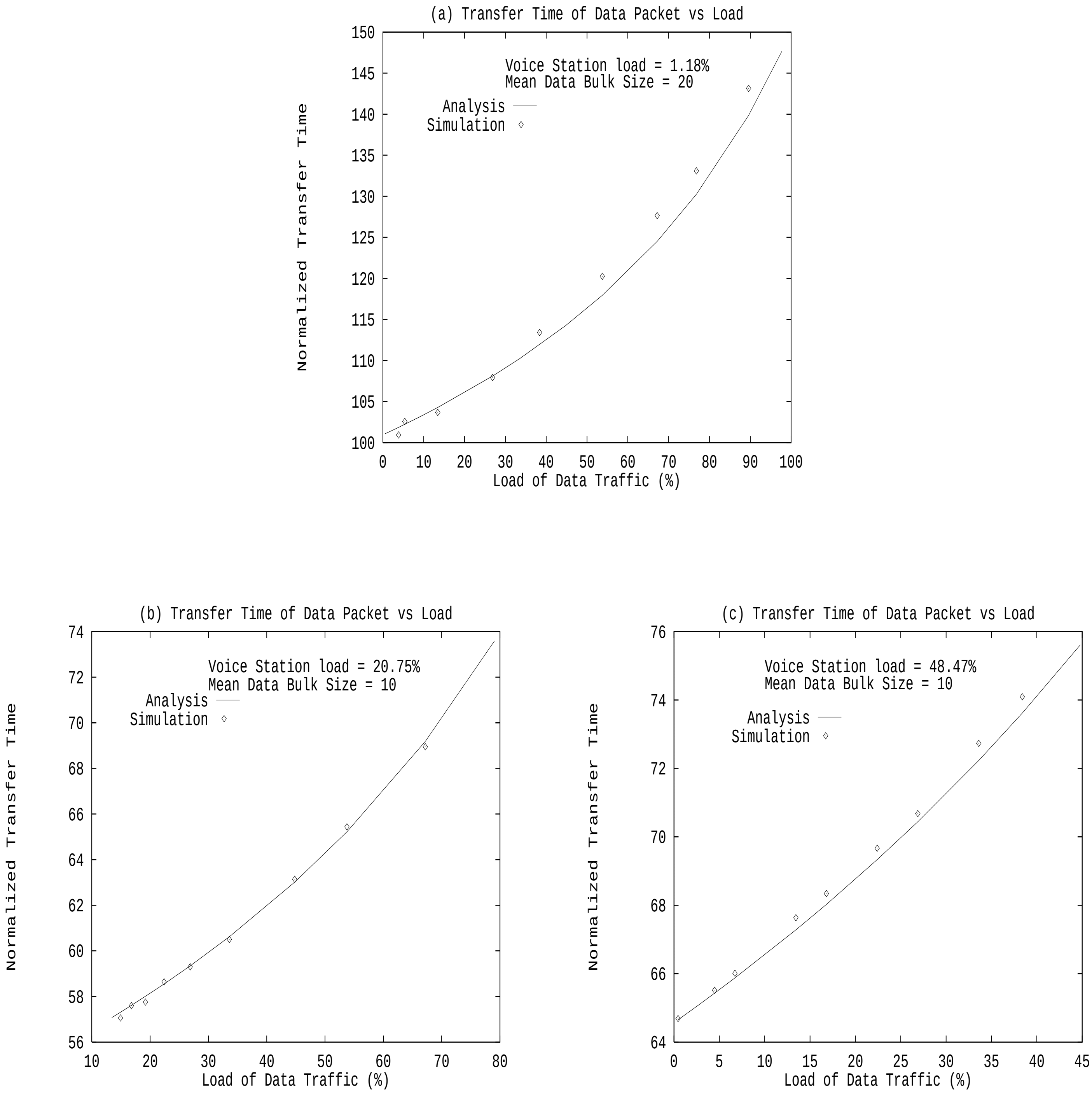

Figure 8: Transfer time of data packets vs Load of data traffic

performance of existing systems. In this section we present an application of our analysis to an interesting design problem.

Consider the problem of connecting M voice (data) sources to a ring. They can be connected individually (a station will support one source), or in a cluster (a station will support several sources) to the ring. Since, a station connected to the ring introduces a delay due its processing overhead of slots, ring latency in the first alternative will be higher than that in the second. On the other hand, in the second alternative, each station supports more than one voice (data) source, leading to higher waiting times in the station queues. Our objective in this example is to compute the optimal number of voice (data) sources per station so that transfer time is minimized[12]. For simplicity, we consider uniform *group-size* (the number of sources connected per station) throughout the network. The group-size is denoted by n. The number of stations that need to be connected to the network is $N = M/n$. As we will see shortly, the distribution of number of sources per station (i.e. the group-size n) plays a major role in determining the transfer time of voice and data traffics.

The ring latency (R) can be divided into two components, $R = R_{fixed} + R_{variable}$. The fixed component is the propagation delay in the ring which is determined by the length of the cable. The variable component is contributed by stations connected to the ring. Each station processes the passing slots and there by introduces a delay. The amount of delay introduced depends on the processing speed of the station. In the present scenario, the aggregate delay introduced by all the stations is responsible for the variable part of the ring latency. Assuming similar processing capability at each station, $R_{variable}$ is directly proportional to N. We write $R_{variable} = N\beta$, where β is the delay introduced by a station. We express β as the fraction of the slot size[13]. So, ring latency can be expressed as

$$R = R_{fixed} + N\beta.$$

In figures 9 and 10 we plot the mean transfer times of voice ($\overline{T_v}$) and data ($\overline{T_d}$) traffic, respectively, against the group-size (n) for different values of β. The mean transfer time is normalized to the transmission time of a slot (x) and n is expressed as the percentage of total number of sources (M). Transfer time consists of ring latency and response time. Variability in response time for different values of n is due to the variability in waiting time only. Service time depends on the total load and does not change with the change in group size. For small values of n, sources per station are few and traffic intensity at a station is low [14]. Therefore, the waiting time of a packet in a station is small. However, when N is big this introduces large ring latency. The ring latency

[12] Both ring latency and waiting time are parts of transfer time.

[13] It can be multiplied by the transmission time of a slot to get the time latency.

[14] The parameters of the MMPP for n superposed voice sources are computed as in [10]. The data sources are considered as Poisson sources whose superposition is easy to compute.

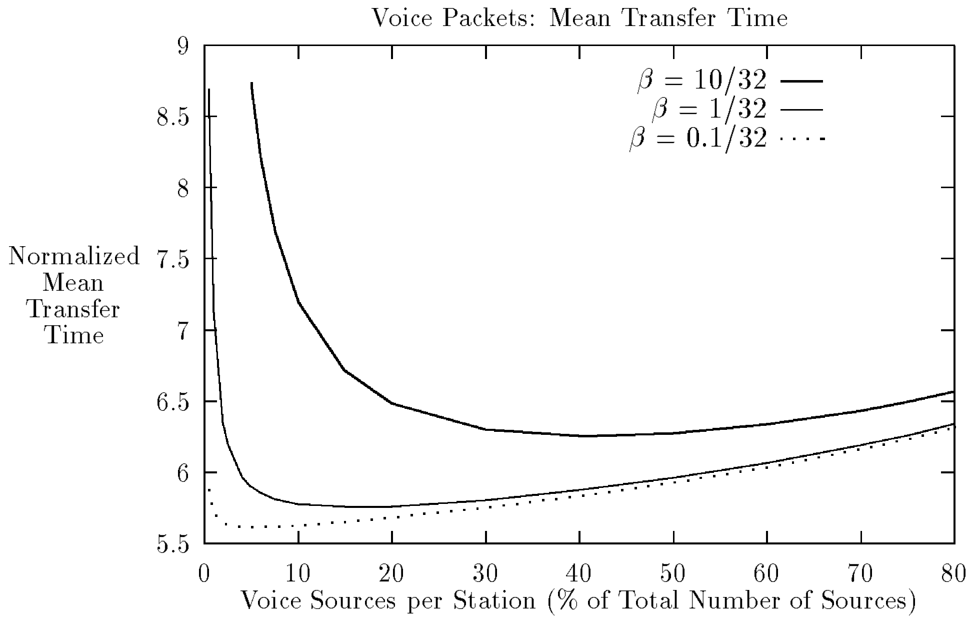

Figure 9: Mean transfer time of a voice packet vs Number of voice sources per station

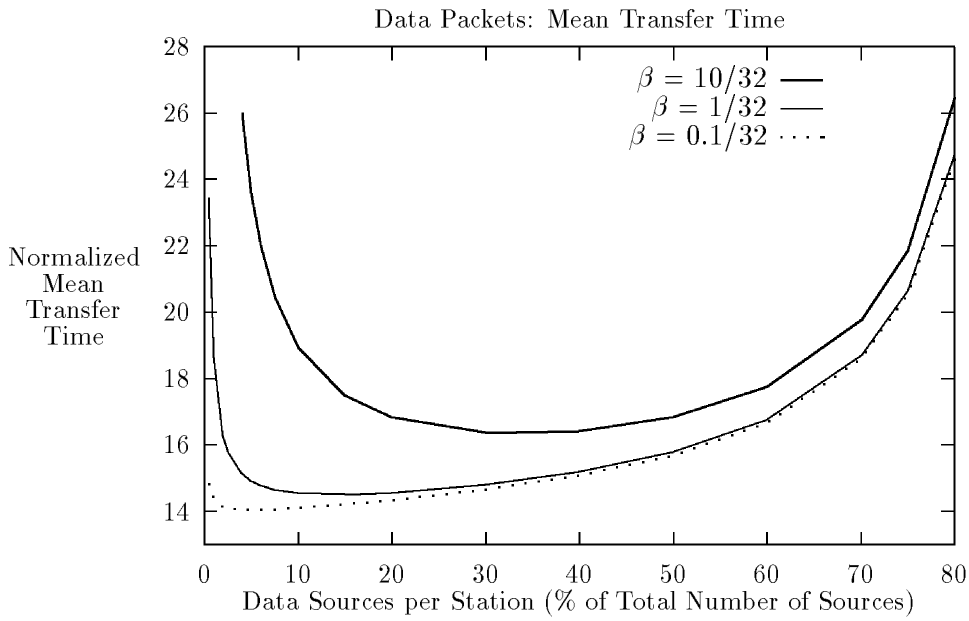

Figure 10: Mean transfer time of a data packet vs Number of data sources per station

becomes the predominant factor in the transfer time for small values of n. So as n increases (and N decreases), ring latency decreases and so do the mean transfer times. After a certain value of n the scenario changes. A station now handles large number of sources (large n) but very few stations are connected to the ring (small N). The superposed source increases the traffic intensity at a station and so the waiting time in the station increases. However, ring latency decreases because of small value of N. Now the waiting time becomes predominant in the transfer times. Consequently, the mean transfer times start increasing. It should also be noted that as β decreases (i.e. processing at a station becomes faster), the transfer time decreases, which is what we expect. Since the transfer time of data traffic depends on the load of both the voice and data traffic, its mean transfer time increases faster than that of the voice traffic. The optimal value of the group-size with respect to the transfer time of a packet is the value of n at the minima of the curve (call it n_{opt}). For best performance the group-size should be chosen as n_{opt}.

9 Comparative Study

In this section, we compare the performance of the protocol with token ring based FDDI [30], register-insertion ring based TORNET2 [39], and slotted ring based CFR [38, 11] for voice-data integration through an extensive simulation [25]. FDDI implements timed token rotation to guarantee bounded ring access time. In TORNET2 voice traffic uses register-insertion protocol and data traffic follow token ring protocol. In CFR voice traffic is assigned head of the line priority over data traffic in a slotted ring environment. In the rest of the discussion we refer to the preemtive protocol as PREEMPT.

The simulation is performed using *csim*. We choose a ring network of length 2.56 km, the length of a typical LAN [13]. Variation in ring length does not change the relative qualitative behavior of the protocols for the metrics considered. With the signal speed of 2×10^8 m/s, the propagation delay in the ring is 12.8 μs. We assume that the delay introduced by stations due to their processing time are roughly the same for the protocols studied and is not taken into account in the simulation model. Link speed is assumed to be 80 Mb/s.

A voice source is modeled as alternating between talkspurt and silence period. The durations of talkspurt and silence period are exponentially distributed with means 352 ms and 650 ms, respectively [33]. We assume that voice traffic is packetized into 32-byte packets. This means that during the talkspurt voice packets are generated 4 ms apart, assuming a coding rate of 65 Kb/s. Data stations are assumed to be Poisson sources. Data messages are assumed to be integral multiples of 32 bytes. Each station can support voice traffic (referred to as voice station), data traffic (referred to as data station) or both. A station can support single or multiple voice and/or data streams. We assume all the stations of a class behave in the same fashion and the destination of a packet is chosen randomly from all the other stations in its class.

In FDDI, the value of Target Token Rotation Time(TTRT) is adjusted to accommodate the bandwidth requirements of all the voice stations [13]. Keeping it constant compares FDDI unfavorably with the others. We select the value of TTRT that gives the best performance satisfying all the requirements, and tune it with the change in voice traffic load. We also assume that the registers in TORNET2 and slots in PREEMPT and CFR are 32 bytes each. In CFR, all the slots are considered to be *normal* slots. This is because we observed that a voice station cannot keep on holding a *channel* slot as the inter-arrival time of voice packets is greater than the slot rotation time. *Channel* slots are not very useful for data traffic either because of the bursty nature of this traffic. Therefore, these slots would behave like *normal* slots, even if they were present in the ring.

In the rest of the section, we present the comparative study of the four protocols. The load in the following plots is the total load offered to the network by all the stations of a class and is expressed as a percentage of the total transmission capacity of the link. Simulation results are obtained at 95% confidence level with the size of the confidence interval less than 2% of the mean. Unless otherwise mentioned, a voice station supports a single source and a data station generates fixed size packets.

We perform three sets of experiments. In the first two sets we study the impact of the change in load due to one class of traffic keeping the load due to the other class fixed. In the final experiment we vary both voice and data traffic load. Before starting the discussion on the observations from the experiments we define some terms which will be used subsequently. *Ring access time* is defined as the time a packet waits at the head of the queue. This is the time between the end of the previous transmission and the beginning of a new transmission. *Ring latency* is defined to be the time needed to travel around the ring. It is the sum of the ring propagation delay and the delays introduced by the stations. *Transfer time* is the time between the arrival of a packet at the source and its receipt at the destination. It is the sum of the waiting time at the source, the ring access time and the propagation delay from the source to the destination.

9.1 Effect of Variation in Voice Load

In this section we study the effect of change in the voice load on the performance of different protocols. We conduct two experiments. In the first one we vary the load due to voice traffic by changing the number of voice stations connected to the ring. In the second experiment we vary the number of voice sources per station.

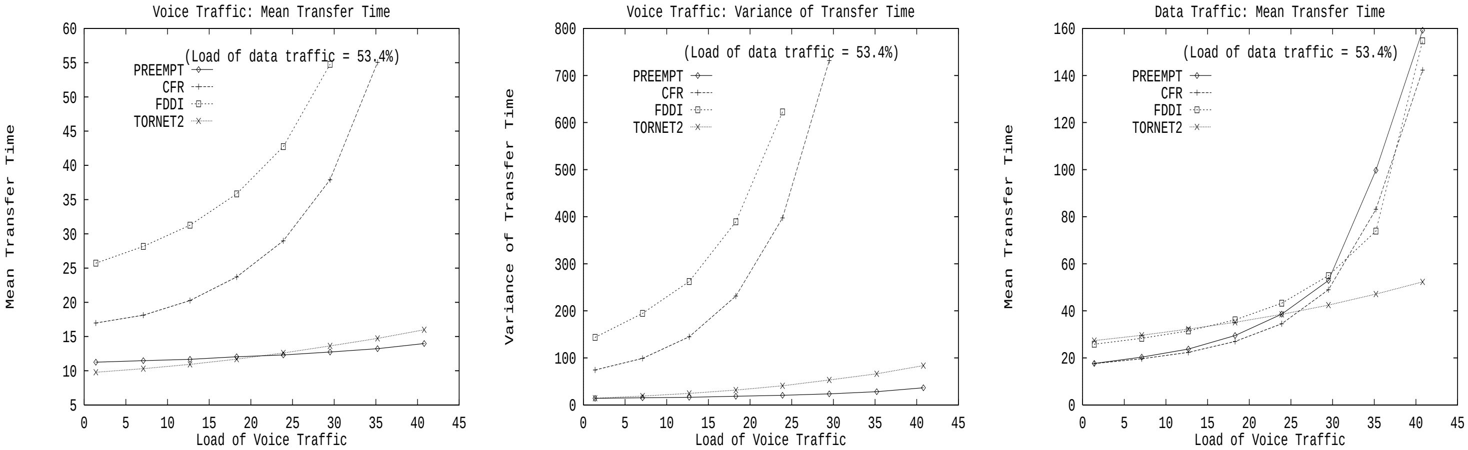

Figure 11: The ring supports a fixed number of data stations. The load of voice traffic is varied by changing the number of voice stations.

Varying the number of voice stations: Number of voice stations connected to the ring is varied, keeping the load due to data traffic fixed. The results are plotted in figure 11. The mean transfer time of voice traffic (figure 11(a)) in FDDI is high because of its high ring access time. CFR performs worse than PREEMPT since, unlike in PREEMPT, voice traffic in CFR gets affected by data traffic from upstream which takes away some slots. Thus voice packets face higher ring access time and that is reflected in higher transfer time. TORNET2 outperforms others at low load but its performance deteriorates with increase in load. This is because in TORNET2, voice packets are inserted into the ring almost as soon as they arrive. Moreover, at low load, the ring latency is close to the propagation delay since there are not many registers inserted in the ring. So both ring access time and ring latency are low, resulting in low transfer time. In PREEMPT, the ring latency is at most that in TORNET2, but its ring access time is higher as packets have to wait to get a free slot[15]. With the increase in the load, the number of registers inserted in TORNET2 increases. This has twofold effect on the transfer time. Firstly, the ring latency increases due to the increase in number of inserted registers. Secondly, since a station can have at most one outstanding register in the ring, a new insertion waits till the previous one is removed after a complete ring rotation. Thus ring access time also increases. The increase in transfer time in FDDI, CFR and PREEMPT are due to higher ring access time only. This is why transfer time in TORNET2 increases very rapidly with load and becomes higher than all the others.

The variance of transfer time (figure 11(b)) in FDDI is higher than all the others since the value of TTRT is increased to accommodate the requirements of all the voice stations. The holding time

[15] TORNET2 waits at most one register (which is the same as a slot) whereas PREEMPT waits at least one slot.

of the token becomes more dependent on the load of the individual stations for higher values of TTRT. This introduces more variability in the Token Rotation Time (TRT) and thus the variance in transfer time increases. CFR performs worse than PREEMPT and TORNET2 since the voice traffic in CFR gets affected by the data traffic, whereas, both PREEMPT and TORNET2 are independent of data traffic load.

In case of data traffic (figure 11(c)), at low load CFR and PREEMPT are very close since very few data packets get overwritten in PREEMPT. Furthermore PREEMPT tries to send more data packets than CFR since a station supporting both voice and data can have one outstanding slot for each class of traffic in the former. CFR does not allow a station to have more than one outstanding slot. Transfer time of data traffic in both CFR and PREEMPT is lower than that in FDDI and TORNET2 since the ring access time at low load in slotted ring is lower than that in token ring [42]. As the voice load increases, more and more data packets get overwritten and the transfer time in PREEMPT becomes higher than that in CFR. At a very high load, mean transfer time in TORNET2 becomes lower than that in CFR, PREEMPT and FDDI. This is because data transmission faces "starvation" in CFR and PREEMPT as CFR gives head of the line priority to voice traffic and data packets are left at the end of the queue, and in PREEMPT data packets get overwritten at a higher rate[16]. Mean transfer time in FDDI also increases since allocated bandwidth is used up by the voice traffic and hardly any data packet gets transmitted. The token ring protocol of TORNET2 allows some (although very low) data communication. The transfer time increases because of high ring latency as many registers are inserted.

Varying the number of voice sources per station: We simulate stations supporting both voice and data traffic and keep the number of stations fixed. The load at a voice station is changed by changing the number of voice sources per station. The results are shown in figure 12. For voice traffic (figures 12(a) & (b)) the qualitative behavior of the protocols is very similar to the previous case (figures 11(a) & (b)). However, increases in mean and variance in TORNET2 are sharper than in figure 11(a) & (b). High network load in figure 11 is due to large number of lightly loaded voice stations, whereas here it is due to small number of heavily loaded voice stations. When the stations are heavily loaded, register-insertion ring acts like a slotted ring with slightly lower ring access time and much higher ring latency. This is why TORNET2 shows sharper increase in both mean and variance than PREEMPT. Data traffic (figure 12(c)) shows similar behavior as in figure 11(c) since it is affected by the total load of voice traffic on the system rather than the load at the individual stations.

Summary: For voice traffic, TORNET2 is the best at low load. However, PREEMPT outperforms

[16] Because of this phenomena we proposed a variation to the original protocol in [26] which allows flow of data traffic even at high voice load. This feature is not simulated in our present model.

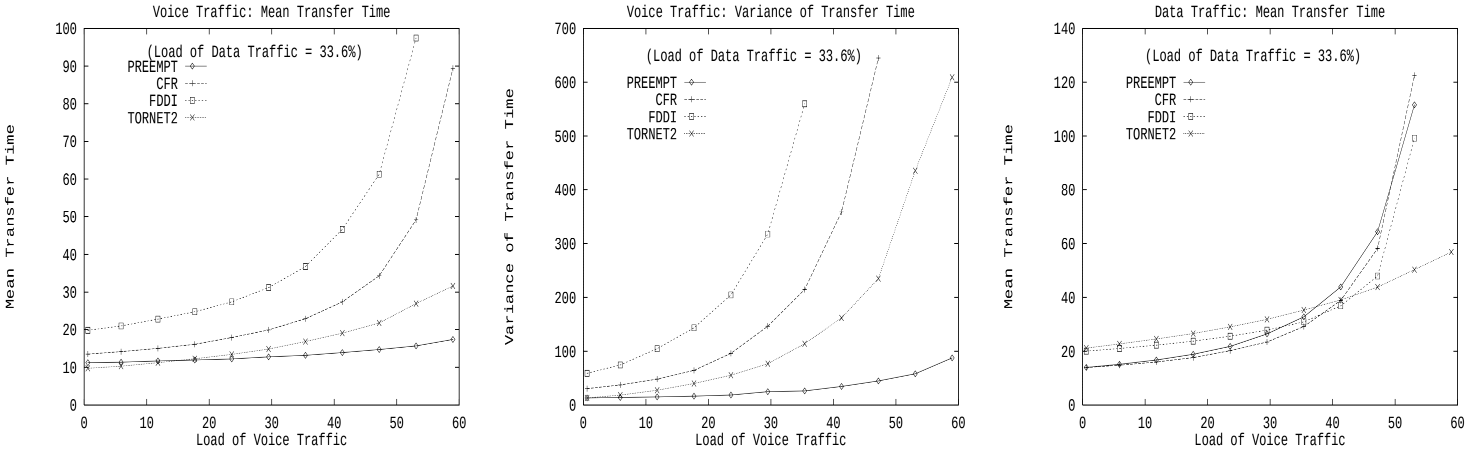

Figure 12: The ring supports fixed number of voice and data stations. The load of voice traffic is varied by changing the number of voice sources per station.

TORNET2 as load increases and performs the best in providing low jitter. TORNET2 is more sensitive to individual station load than the total load on the ring (compare figures 11(a) & (b) with 12(a) & (b)). So its performance will be significantly affected by non-uniform load distribution. PREEMPT shows very similar behavior in both scenarios. CFR and FDDI perform worse than both TORNET2 and PREEMPT. For data traffic, CFR performs the best. PREEMPT exhibits trends very close to CFR. TORNET2 performs the worst among all at low to moderate load. However, when voice traffic load is very high, data traffic suffers starvation in CFR, PREEMPT and FDDI while TORNET2 allows some data transfer.

9.2 Effect of Variation in Data Load

In this section we study the effect of change in the data load on the performance of different protocols. We simulated two scenarios — one with varying the load at individual data stations, and the other in which the number of data stations connected to the ring is changed.

Varying the data packet arrival rate: Stations support both voice and data traffic. The load due to data traffic is changed by varying the mean inter-arrival time of data packets. We plot the results in figure 13. In FDDI, token rotation time approaches twice of TTRT with the increase in data traffic load as the unused part of the bandwidth (THT) is used up by data stations and causes overrun [14]. So the mean and variance of voice packet transfer time increase very rapidly (figures 13(a) & (b)). Voice traffic in CFR gets affected since the active upstream data stations seize considerable number of slots, and the ring access time increases with increase in data traffic

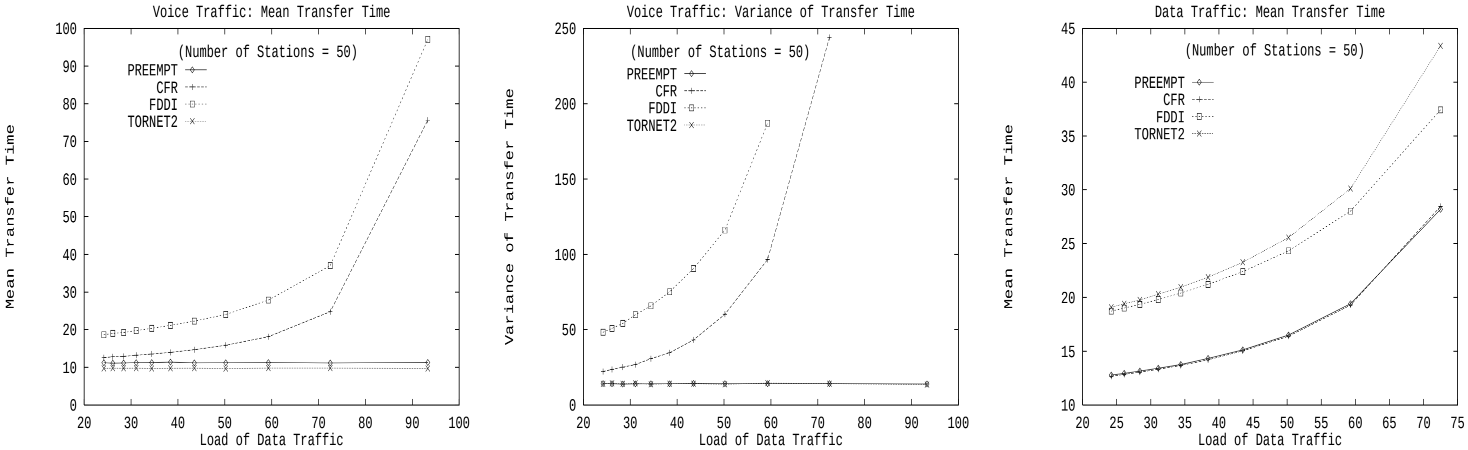

Figure 13: The ring supports fixed number of voice and data stations. The load of data traffic is varied by changing the inter-arrival time of data packets.

load. Transfer time of voice traffic in TORNET2 and PREEMPT are independent of data load and their mean and variance remain unaffected due to change in the data traffic load. TORNET2 has lower ring access time than PREEMPT and shows marginally better performance in mean transfer time (figure 13(a)).

For data traffic (figure 13(c)), CFR and PREEMPT show very similar behavior. Although data packets get overwritten in PREEMPT, it sends more packets than CFR. Moreover, the number of preemptions goes down when a voice station becomes quiescent (silence period). So the mean transfer times are very close. In FDDI, the bandwidth distribution among data traffic from different stations becomes "uneven" with increase in data load. Since a station has high data traffic intensity, it will use as much of its THT as possible, depriving other stations for that token rotation. So the access time as a whole increases and so does the mean transfer time. TORNET2 performs worse than FDDI because it completely ignores the data traffic flow.

Varying the number of data stations: We keep the number of voice stations fixed and change the number of data stations connected to the ring. We observe the qualitative behavior (figure 14) similar to the previous case (figure 13). The increase in mean and variance of transfer time of voice traffic (figures 14(a) & (b)) in CFR is sharper than that in figure 13(a) & (b). This is because the receipt of free slots from upstream data stations depends more on the number of such active stations than on the load at the stations, since each station can potentially hold only one slot. The mean transfer time of data traffic (figure 14(c)) for FDDI shoots up because its TTRT cannot provide service to all the data stations present in the ring. They undergo a long waiting period.

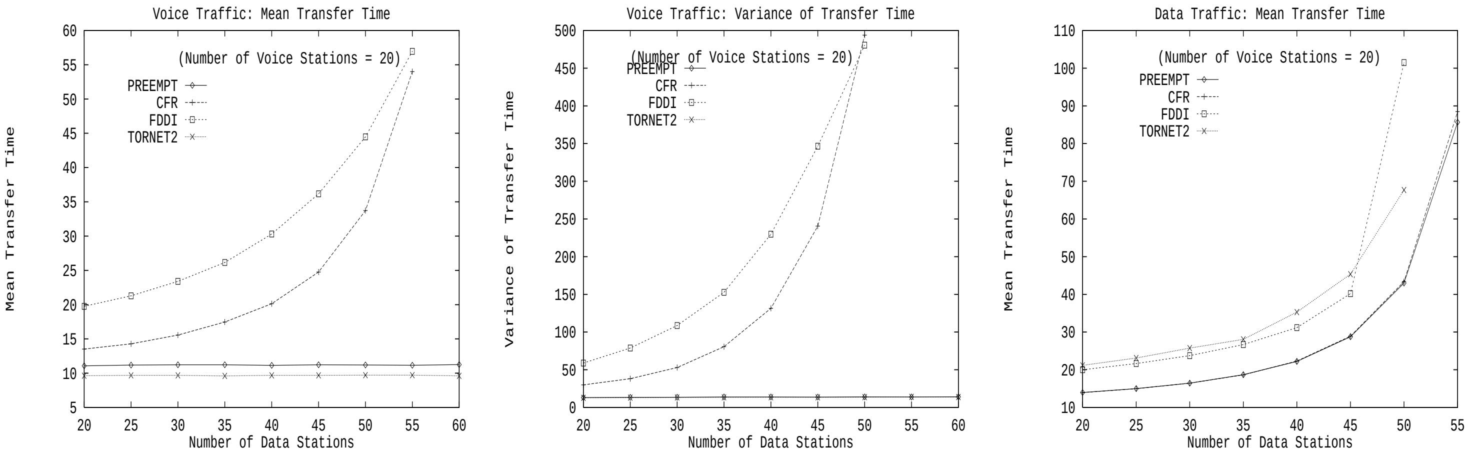

Figure 14: The ring supports fixed number of voice stations. The load of data traffic is varied by changing the number of data stations.

Summary: The performance metrics of voice traffic in PREEMPT and TORNET2 are independent of data traffic load. The mean transfer time of voice traffic in TORNET2 is marginally lower than that in PREEMPT, but they both provide similar variance. Transfer time of voice traffic in FDDI and CFR suffers heavily from data traffic load. For data transfer, CFR performs the best. PREEMPT is very close to it and TORNET2 is the worst.

9.3 Effect of Variation in Voice and Data Load

In this section we study the effect of change in both the voice and data load. The traffic load is varied by changing the number of stations connected to the ring. Two types of data sources are considered — one generating fixed size packets and the other generating data messages of two different sizes.

Fixed size data packets: Mean transfer time of voice traffic in TORNET2 and PREEMPT shows small increase as compared to CFR and FDDI, with increase in the number of stations 15(a). This is due to the fact that the performance of voice traffic in TORNET2 and PREEMPT depends only on the voice load, whereas in CFR and FDDI it depends on both voice and data load. Mean transfer time in TORNET2 is slightly lower than that in PREEMPT since the ring access time in the former is lower than that in the latter.

The variance of transfer time of voice traffic is shown in figure 15(b). TORNET2 shows lower variability than PREEMPT at low load. However, as load increases the relative order is reversed since large number of inserted registers in the ring increases the variability of ring latency in TORNET2.

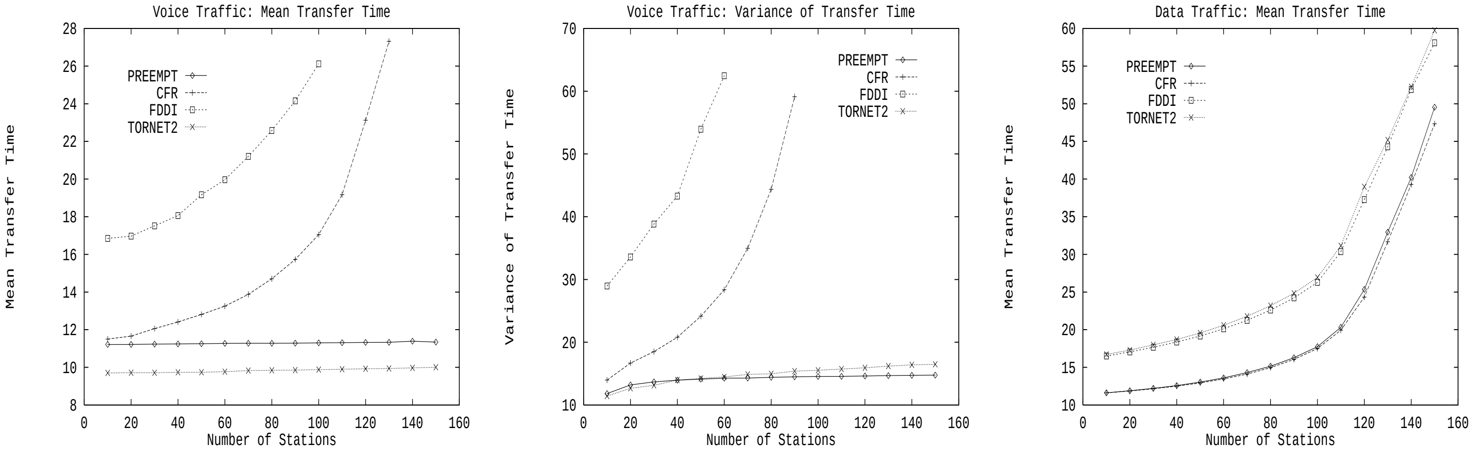

Figure 15: The ring supports equal number of voice and data stations. The load to the network is varied by changing the number of stations of each class.

Variability in PREEMPT is due to variability in the ring access time.

In data transfer (figure 15(c)), PREEMPT performs almost as good as the best performer CFR. But with the increase in load the difference between them increases since more data packets get overwritten and are retransmitted in PREEMPT. TORNET2 performs poorly as it devotes most of its bandwidth to serve the voice traffic. The transfer time remains high in FDDI as a station sends data packets only when it has some bandwidth available after sending the voice packets.

Mixed size data messages: This experiment differs from the previous one in that the data sources are mixed sources as used in [42]. There are two sizes of messages generated: 13% of them are small that can fit in one slot, 87% are big that needs 8 slots to transmit. The results are shown in figure 16.

Performance of the protocols for voice traffic (figures 16(a) & (b)) shows behavior similar to the previous case. Their performance is different for data traffic (figure 16(c)). Both the token rings (FDDI and the data transfer protocol of TORNET2) perform better than the slotted rings (CFR and PREEMPT) for variable size packets. This is because in a slotted ring a message is segmented into packets that can fit in a slot. Each packet incurs ring access delay that is summed up for a message. In a token ring the message can be transmitted in a single attempt.

Summary: TORNET2 and PREEMPT exhibit very slow increase in mean and variance of transfer time of voice traffic compared to CFR and FDDI. Although TORNET2 shows slightly lower mean than PREEMPT, their order gets reversed in the variance of transfer time. For data traffic, TORNET2 and FDDI show similar behavior and outperform CFR and PREEMPT when messages are long. For short messages CFR perfroms the best followed by PREEMPT. This observation is in agreement with

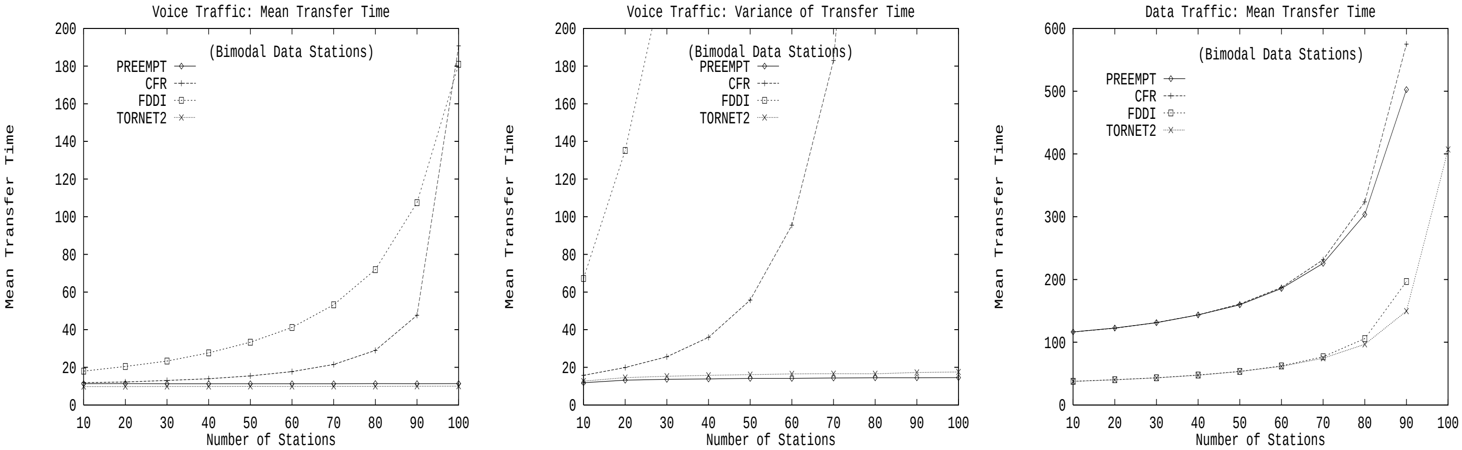

Figure 16: The ring supports equal number of voice and data stations. The load to the network is varied by changing the number of stations of each class. Data stations are mixed sources with 13% small messages and 87% big messages.

the conventional wisdom that the slotted rings are good for short packets whereas the token rings are better suited for long packets.

10 Conclusion

In this paper we have studied the performance of a preemptive protocol (as opposed to reservation-based protocols) for integrating voice and data in a high speed LAN/MAN environment. Our analysis takes into account the correlated nature of the voice traffic and the burstiness of the data traffic. With some independence assumptions, closed form expressions of transforms of response time distributions are derived for both classes of traffic. The analysis shows close agreement with the simulation. We discuss the computation of some important performance metrics (mean delay, delay jitter, clipping probability for voice traffic, mean packet and message delay for data traffic, etc.) and an application of analytical results in designing LAN/MAN. Last but not the least, a performance comparison of the preemptive protocol with FDDI, CFR and TORNET2 is presented.

In the analysis, we approximated the utilization of a slot by different classes of traffic, assuming uniform behavior of slots at all loads. Although it gives very good approximation at low to moderate load, it is not so accurate at high load. A more accurate modeling of this parameter is under investigation.

In a LAN environment it is likely that a station will support only one voice connection. A single voice source cannot be modeled accurately by a MMPP source as the packet arrival is deterministic during talkspurts. Arrival process can be better approximated by a phase type distribution and exponential sojourn time for silence period. Though this model can be analyzed numerically using matrix analytic technique, closed form expression for response time cannot possibly be derived.

Our simulation study shows that priority schemes used in CFR and FDDI do not perform well for voice traffic. Making voice traffic independent of data traffic shows considerable improvement in both TORNET2 and the preemptive scheme. TORNET2 looks very promising as far as the mean transfer time of voice traffic is considered, but it fails to provide low jitter. The preemptive protocol on the other hand ensures lower delay jitter at the expense of marginally higher mean transfer time. Moreover, the delay jitter exhibits very slow increase with increase in load in the preemptive protocol. For data traffic, CFR performs the best for fixed size packets. The preemptive protocol exhibits mean data transfer time close to CFR. In TORNET2 data traffic is completely ignored and it performs poorly.

References

[1] J. Abate and W. Whitt. The Fourier-series Method for Inverting Transforms of Probability Distributions. *Queueing Systems*, 10:5–88, 1992.

[2] L. N. Bhuyan, D. Ghosal, and Q. Yang. Approximate Analysis of Single and Multiple Ring Networks. *IEEE Transactions on Computers*, 38(7), July 1989.

[3] J. R. Brandsma, A. A. M. L. Bruekers, and J. L. W. Kessels. PHILAN: a Fiber-Optic Ring for Voice and Data. *IEEE Communications Magazine*, 24(12):16–22, 1986.

[4] L. M. Casey, R. C. Dittburner, and N. D. Gamage. FXNET: a Backbone Ring for Voice and Data. *IEEE Communications Magazine*, 24(12):23–28, 1986.

[5] R. Cohen and A. Segall. A New Scheme for Dynamic Management of Isochronous Channels in Integrated Rings. *Computer Networks and ISDN Systems*, 24:131–144, 1992.

[6] E. Gelenbe and R. Iasnogorodski. A Queue with Server of Walking Type. *Annales de l'Institut Henri Poincare, Calcul des Probabilites et Statistique*, XVI(1), 1980.

[7] W. L. Genter and K. S. Vastola. Delay Analysis of the FDDI Synchronous Data Class. In *Proceedings of the IEEE INFOCOM*, pages 766–773, 1990.

[8] S. Halfin. Batch Delays Versus Customer Delays. *The Bell System Technical Journal*, 62(7):2011–2015, 1983.

[9] J. L. Hammond and P. J.P. O'Reilly. *Performance Analysis of Local Computer Networks*. Addison-Wesley Publishing Company, 1986.

[10] H. Heffes and D. M. Lucantoni. A Markov Modulated Characterization of Packetized Voice and Data Traffic and Related Statistical Multiplexer Performance. *IEEE Journal on Selected Areas in Communications*, 4(6):856–868, September 1986.

[11] A. Hopper and R. M. Needham. The Cambridge Fast Ring Networking System. *IEEE Transactions on Computers*, 37(10):1214–1223, October 1988.

[12] O. C. Ibe and D. T. Gibson. Protocol for Integrated Voice and Data Local Area Networks. *IEEE Communications Magazine*, 24(7):30–36, 1986.

[13] R. Jain. Performance Analysis of FDDI Token Ring Networks: Effect of Parameters and Guidelines for Setting TTRT. In *Proceedings of the ACM SIGCOMM 90 Conference*, pages 264–275, 1990.

[14] M. J. Johnson. Proof that Timing Requirements of the FDDI Token Ring Protocols are Satisfied. *IEEE Transactions on Communications*, 35(6), June 1987.

[15] A. E. Kamal and V. C. Hamacher. Utilizing Bandwidth Sharing in the Slotted Ring. *IEEE Transactions on Computers*, 39(3), March 1990.

[16] D. Karvelas and A. Leon-Garcia. Performance of Intergrated Packet Voice/Data Token-Passing Rings. *IEEE Journal on Selected Areas in Communications*, 4(6), September 1986.

[17] D. Karvelas and A. Leon-Garcia. Transmission of Time Critical Information over FDDI and FDDI-II Networks. In *Proceedings of the GLOBECOM*, pages 1045–1049, 1992.

[18] W. M. Loucks, V. C. Hamacher, B. R. Preiss, and L. Wong. Short-Packet Transfer Performance in Local Area Ring Network. *IEEE Transactions on Computers*, 34(11), November 1985.

[19] D. M. Lucantoni. New Results on the Single Server Queue with a Batch Markovian Arrival Process. *Communications in Statistics — Stochastic Models*, 7(1):1–46, 1991.

[20] D. M. Lucantoni, K. S. Meier-Hellstern, and M. F. Neuts. A Single-server Queue with Server Vacations and a Class of Non-renewal Arrival Processes. *Advances in Applied Probability*, 22:676–705, 1990.

[21] K. J. Maly, E. C. Foudriat, R. Mukkamala, C. M. Overstreet, and D. Game. Dynamic Allocation of Bandwidth in Multichannel Metropolitan Area Network. *Computer Networks and ISDN Systems*, 25:203–223, 1992.

[22] T. B. Maples and T. Suda. Voice and Data Integrated Transmission on an FDDI Network. In *Proceedings of the GLOBECOM*, pages 1050–1056, 1992.

[23] S. Mukherjee, D. Saha, and S. K. Tripathi. Performance Evaluation of a Preemptive Protocol for Voice-Data Integration in Ring-based LAN/MAN. Technical Report CS-TR-3035, UMIACS-TR-93-15, University of Maryland, February 1993.

[24] S. Mukherjee and S. K. Tripathi. A Multiclass Preemptive Protocol for a Hierarchical Slotted-Ring MAN and its Performance Analysis. Technical Report UMIACS-TR-92-99, CS-TR-2960, University of Maryland, September 1992.

[25] S. Mukherjee and S. K. Tripathi. A Comparative Study of Media Access Control Protocols for Integrated Services Ring LANs. In *Proceedings of the International Conference on Computer Communications and Networks*, pages 74–78, 1993.

[26] S. Mukherjee, S. K. Tripathi, and D. Ghosal. A Multiclass Priority-based Slotted-Ring LAN and its Analysis. *IEEE Transactions on Computers*, 42(8), August 1993.

[27] M. F. Neuts. *Structured Stochastic Matrices of M/G/1 Type and Their Applications.* Marcel Dekker, Inc., 1989.

[28] A. Pattavina and G. Pacifici. Performance Evaluation of Ring Networks Supporting a Packet Voice Service. *Computer Networks and ISDN Systems*, 15:31–45, 1988.

[29] V. Ramaswami. The N/G/1 Queue and its Detailed Analysis. *Advances in Applied Probability*, 12:222–261, 1980.

[30] F. E. Ross. An Overview of FDDI: The Fiber Distributed Interface. *IEEE Journal on Selected Areas in Communications*, 7(7):1043–1051, September 1989.

[31] H. D. Schwetman. CSIM Reference Manual (Revision 14). Technical Report ACA-ST-257-87 Rev 14, Microelectronics and Computer Technology Corporation, 1990.

[32] E. M. Spiegel. Performance Analysis of the Timed-Token Protocol: A Vacation Model. In O. Spaniol and A. Danthine, editors, *High Speed Networking, III*, pages 109–124. IFIP, Elsevier Science, 1991.

[33] K. Sriram and W. Whitt. Characterizing Superposition Arrival Processes in Packet Multiplexers for Voice and Data. *IEEE Journal on Selected Areas in Communications*, 4(6):833–846, September 1986.

[34] T. Suda and T. T. Bradley. Packetized-Voice/Data Integrated Transmission on a Token Passing Ring Local Area Network. *IEEE Transactions on Communications*, 37(3):238–244, 1989.

[35] D. T. W. Sze. A Metropolitan Area Network. *IEEE Journal on Selected Areas in Communications*, 3(6), November 1985.

[36] H. Takagi. *Analysis of Polling Systems.* The MIT Press, 1986.

[37] H. Takagi. *Queueing Analysis A Foundation of Performance Evaluation*, volume 1. Elsevier Science Publishers B.V., 1991.

[38] S. Temple. The Design of the Cambridge Fast Ring. In I. N. Dallas and E. B. Spratt, editors, *Ring Technology Local Area Networks*, pages 79–88. Elsevier Science Publishers, 1984.

[39] Z. G. Vranesic, V. C. Hamacher, A. K. Sanwalka, and S. G. Zaky. A Hybrid Token/Insertion Ring LAN. In *Proceedings of the IEEE INFOCOM*, pages 211–220, 1991.

[40] J. W. Wong and P. M. Gopal. Analysis of a Token Ring Protocol for Voice Transmission. *Computer Networks*, 8:339–346, 1984.

[41] Q. Yang, D. Ghosal, and S. K. Tripathi. Performance Study of Two Protocols for Voice/Data Integration on Ring Networks. *Computer Networks and ISDN Systems*, 23(4):267–286, January 1992.

[42] M. Zafirovic-Vukotic, I. G. Niemegeers, and D. S. Valk. Performance Analysis of Slotted Ring Protocols in HSLAN's. *IEEE Journal on Selected Areas in Communications*, 6(6), July 1988.