QTS: A QOS-Guaranteed Transport System for Broad-Band Multimedia Communications

Maria C. Yuang, *Member, IEEE*, and Jen C. Liu

***Abstract*—In this paper, we propose a quality-of-service (QOS)-guaranteed transport system (QTS), which offers various QOS's at the transport layer for broad-band multimedia communications. The QTS, composed of a bandwidth allocator and a transport protocol module, supports three classes of applications requiring different bit rates, delay sensitivity, and loss sensitivity. The bandwidth allocator intelligently manages the allocation of transport-layer bandwidth at the expense of imposing inevitable blocking of delay-sensitive application connections. The transport protocol module of the QTS performs rate-based flow control for delay-sensitive applications based on transfer rates predetermined by the bandwidth allocator. In addition, the module accomplishes error control only for loss-sensitive applications. As a result, as will be shown, by providing guaranteed rates and reducing the error control overhead, the QTS offers satisfactory bounded delays and jitters for delay-sensitive applications, while incurring minimal throughput degradation for loss-sensitive applications. Finally, we demonstrate the superiority of the QTS over transmission control protocol (TCP) via simulation results in terms of maximum and mean system delays and delay jitter.**

***Index Terms*— Bandwidth allocation, broad-band multimedia communications, MPEG-I and MPEG-II encoding, quality of service, rate-based flow control, transmission control protocol, window-based flow control.**

I. Introduction

BROAD-BAND networks have been widely deployed to support broad-band multimedia applications [1]–[8], including file transfers, images, audio, and high-quality video. These applications immensely require the guarantee of quality of service (QOS), such as bounded end-to-end delay and jitter and/or error-free transmissions. Broad-band integrated services digital network (BISDN)/ asynchronous transfer mode (ATM) [1], [9]–[12] has been designed to meet these requirements at lower layers of the protocol stack. Consequently, this results in the shift of the performance bottleneck toward the transport layer [13], [14].

Traditional transport protocols, such as the transmission control protocol (TCP) [15] and International Standards Organization (ISO) TP4 [16], were designed for low-speed error-prone networks. These protocols perform end-to-end flow control by means of the sliding window mechanism. The mechanism provides low latency transmissions under light loads by increasing the window size and offers robust communications under heavy loads by decreasing the window size. However, on the one hand, the modification of the window size should be kept at a minimum for reducing the overhead being imposed to networks. On the other hand, the window size should be frequently modified to dynamically adapt to varying traffic characteristics. Consequently, the dilemma renders these protocols unviable for broad-band multimedia applications exhibiting diverse traffic characteristics in nature.

To alleviate the problem, versatile message transaction protocol (VMTP) [17], network block transfer (NETBLT) [18], and Xpress transport protocol (XTP) [19] employed rate-based flow control in lieu thereof. The major challenge becomes the determination of transfer rates. Moreover, the high-speed transport protocol (HSTP) [14] and multistream protocol (MSP) [21], [22] employed loss-free transmissions for loss-sensitive applications and low-latency transmissions for delay-sensitive applications. Real-time transport protocol (RTP) [20] adopted a light-weight protocol (e.g., user datagram protocol (UDP) [15]) to offer low latency for delay-sensitive applications. These protocols, which have been shown to be superior, unfortunately still result in severe performance degradation due to the lack of QOS guarantee should the transport layer be overloaded.

In this paper, we propose a QOS-guaranteed transport system (QTS), which offers diverse QOS's at the transport layer for broad-band multimedia and industrial applications. Potential industrial candidates include applications involving rolling and a capstan lathe factory. The QTS is composed of a bandwidth allocator and a transport protocol module. It supports three classes of applications demanding different bit rates, delay sensitivity, and loss sensitivity. These three classes are: constant-bit-rate (CBR)-based delay sensitive, variable-bit-rate (VBR)-based delay sensitive, and VBR-based loss sensitive.

The bandwidth allocator of the QTS intelligently manages the allocation of transport-layer bandwidth at the expense of imposing inevitable blocking of delay-sensitive application connections. The transport protocol module of the QTS performs rate-based flow control for delay-sensitive applications based on the transfer rate predetermined by the bandwidth allocator. Moreover, the module offers error control only for loss-sensitive applications. As a result, as will be shown, by providing guaranteed rates and reducing the error control (including checksum) overhead, the QTS offers satisfactory bounded delays and jitters for delay-sensitive applications,

Manuscript received October 15, 1996; revised July 20, 1997. This work was supported by the Institute for Information Industry (III) under Contract 86-0040.

The authors are with the Department of Computer Science and Information Engineering, National Chiao Tung University, Hsinchu, 30050 Taiwan, R.O.C.

Publisher Item Identifier S 0278-0046(98)00408-0.

TABLE I
EXAMPLES OF APPLICATIONS USED THROUGHOUT THIS PAPER

Application	Class	Traffic Property	Mean Load	Burstiness	QOS (MSD)
CBR audio	A	IPG: 400	0.0025 (64k bps) [23,24,25,26]	1	470 (150ms [27])
VBR audio	B	α : 0.5 β : 0.5 Talkspurt period: 400 Silence period: 400	0.00125 (32k bps) [9,28]	2	470 (150ms [27])
VBR video: VC using MPEG−I	B	IFG: 104 P(2)=0.6667 P(4)=0.0667 P(8)=0.2666[29]	0.036 (0.9M bps) [30]	27.78	625 (200ms [23,24,27])
VBR video: VC using motion JPEG	B	IFG: 104 P(9)=0.5 P(11)=0.5	0.096 (2.4M bps) [32]	10.4	625 (200ms [23,24,27])
VBR video: VC using DVI	B	IFG: 104 P(14)=0.3 P(15)=0.4 P(16)=0.3 [33]	0.144 (3.6M bps) [33]	6.87	625 (200ms [23,24,27])
VBR video: VOD using MPEG−I	B	IFG: 104 P(2)=0.6 P(6)=0.2 P(20)=0.2	0.06 (1.5M bps) [30,31,34,35]	16.67	781 (250ms [25,34,36])
VBR video: VOD using MPEG−II	B	IFG: 52 P(2)=0.25 P(10)=0.4167 P(25)=0.3333	0.25 (6.4M bps) [31,35]	4	781 (250ms [25,34,36])
Image using JPEG	B	IFG: 390 P(45)=0.5 P(90)=0.5 [31]	0.17 (4.25M bps) [31]	5.88	781 (250ms [25,34,36])
File Transfer	C	File length: 1000	1	1	−

Legend: CBR: constant bit rate; VBR: variable bit rate; VC: video conferencing; VOD: video on demand; IPG: interpacket gap; IFG: interframe gap; MSD: maximum system delay; P(l): the probability of frame size l.

while incurring minimal throughput degradation for loss-sensitive applications. Finally, we demonstrate the superiority of the QTS over TCP via simulation results in terms of maximum and mean system delays and delay jitter.

The remainder of this paper is organized as follows. Section II presents the architecture of the QTS. For the bandwidth allocator module of the QTS, the analytic computation of the minimum transfer rate is proposed in Section III. Section IV shows analytic and simulation results that validate the accuracy of the analytical model and draws performance comparisons between the QTS and TCP in terms of maximum and mean system delays and delay jitter. Section V then focuses on the operations and performance results of the other module, namely. the transport protocol module. Finally, concluding remarks are given in Section VI.

II. QTS ARCHITECTURE

The QTS supports three classes (A, B, and C) of applications, requiring different bit rates, delay sensitivity, and loss sensitivity. In particular, class-A applications are CBR- based delay sensitive, class-B applications are VBR-based delay sensitive, and class-C applications are VBR-based loss sensitive. Table I lists the characteristics of a number of applications which will be used throughout the remainder of this paper. In the table, these application examples are classified with respect to five characteristics: class, traffic property, mean load, burstiness, and QOS in terms of maximum system delay.

In terms of class, CBR audio and file transfers are class-A and class-C applications, respectively. VBR audio, VBR video, and image are examples of class-B applications. In terms of traffic property, CBR audio generates a fixed-length packet every interpacket gap [31], [32], and VBR video generates a variable-length frame every interframe gap [30]–[32], [35]. Moreover, VBR audio generates a fixed-length packet every interpacket gap only during the talkspurt period and generates no packet during the silence period. Without being explicitly specified, the length of time is measured in units of packet processing delay, i.e., 320 μs [39]. The length of packet is measured in units of 1 kbyte. Furthermore, in terms of mean load and burstiness, Motion Picture Experts Group (MPEG)-II-based [40] video on demand (VOD) generates immense mean traffic load, while MPEG-I-based [41] video conferencing (VC) exhibits the largest burstiness. Finally, as for QOS in terms of maximum end-to-end delay (referred to as system delay hereinafter), video applications accept larger maximum system delay than audio applications, while file transfers even tolerate unbounded system delay.

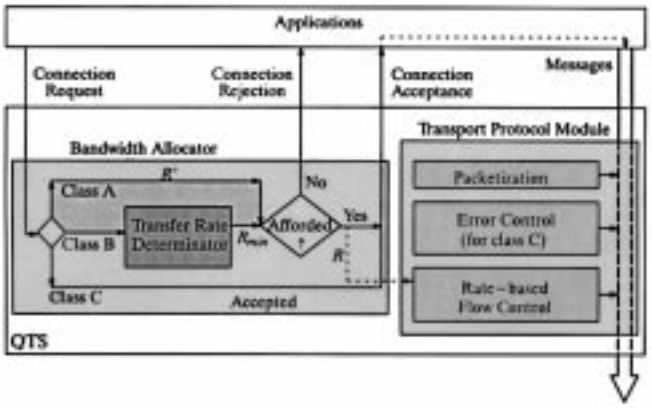


Fig. 1. Architecture of the QTS.

The architecture of the QTS is shown in Fig. 1. The QTS consists of two major components, a bandwidth allocator and a transport protocol module. Basically, the bandwidth allocator is responsible for making the connection acceptance or rejection decision by determining if the allocated transfer rate can be satisfied. In particular, the allocator accepts any class-A connection and sets its CBR value as the transfer rate should its CBR value not exceed the current available transport bandwidth. For any class-B connection, a minimum transfer rate $(R_{\min})$ is first computed based on a queueing model (described in Section III) aimed at offering the QOS guarantee. The connection is accepted only if the resultant rate is tolerable. Finally, the allocator unconditionally accepts all class-C connections. The logic of the bandwidth allocator is formally presented in Fig. 2.

The transport protocol module is responsible for transferring data via three processes: packetization, error control, and rate-based flow control. Application messages are first packetized into fixed-length packets. Error control is engaged only for loss-sensitive class-C packets. The rate-based flow control then regulates the departure of delay-sensitive packets on the predetermined rate basis and polices loss-sensitive packets on an available packet rate (APR) [42] basis.

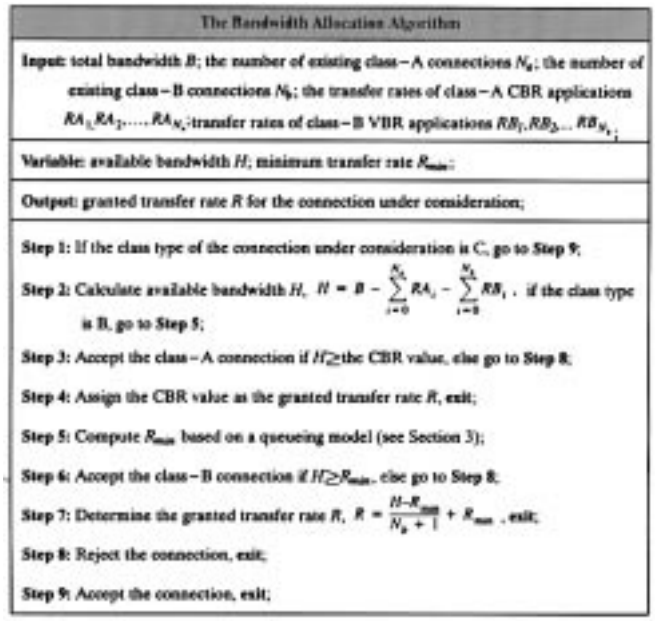


Fig. 2. Bandwidth allocator.

III. Bandwidth Allocator

In this section, we first introduce the VBR-traffic source models. Based on the source models, we derive the unfinished work distribution and, in turn, the maximum system delay (i.e., end-to-end delay). The corresponding minimum transfer rate is finally determined.

A. Traffic Source Models

We consider two types of VBR applications, compressed audio and compressed video [9], [11]. The source model of each type of application is described in the following.

1) VBR Audio: Any VBR-audio traffic can be modeled as a two-state Markov chain [28], alternating between the ON and OFF states, each of which is g slot in length. α denotes the probability of switching from the ON state to the OFF state, and β denotes the opposite probability. Any source stream is considered as a sequence of cycles, each of which consists of a talkspurt period, defined as a number of consecutive ON states each of which is g slot in length, followed by a silence period defined as a number of OFF states, each of which is also g slot long. Moreover, one packet is generated per g time slots in the ON state during the talkspurt period, and no packet is generated in the OFF state during the silence period. Accordingly, the single-step transition probability matrix $P^{(t,t+1)}$ is expressed as

$$P^{(t,t+1)} = \begin{cases} \begin{bmatrix} 1-\beta & \beta \\ \alpha & 1-\alpha \end{bmatrix}, & \text{if } (t-r) \bmod g = 0 \\ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, & \text{if } (t-r) \bmod g \neq 0 \end{cases} \quad (1)$$

where r is the time slot at which the first packet arrives. Let $S^{(0)} \equiv [1,0]$ at the initial time slot. Let $S^{(t)}$ be the state probability vector at time slot t. That is, $S^{(t)} \equiv [s_{\text{ON}}^{(t)}, s_{\text{OFF}}^{(t)}]$, where $s_{\text{ON}}^{(t)}$ and $s_{\text{OFF}}^{(t)}$ are the probabilities being in the ON and OFF states at time slot t, respectively. We can obtain $S^{(t)}$ from $S^{(0)}$ and $P^{(t'-1,t')}, 1 \leq t' \leq t$. With $S^{(t)}$, the probability mass function (PMF) $a^{(t)}(j)$ of the number of packets generated at time slot t becomes

$$a^{(t)}(j) = \begin{cases} s^{(t)} & j = 1, \quad \text{if } (t-r) \bmod g = 0 \\ s_{\text{OFF}}^{(t)}, & j = 0 \\ 1, & j = 0, \quad \text{if } (t-r) \bmod g \neq 0 \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

2) VBR Video: Any VBR-video traffic generates a frame (a positive number of fixed-size packets) every m time slots. A typical example is an MPEG-II encoded stream in which variable-size I, B, and P frames [40] are generated per 1/60 s. The PMF of the number of packets generated at time slot t becomes

$$b^{(t)}(j) = \begin{cases} p(j), & L \geq j \geq 1, \text{if } (t-e) \bmod m = 0 \\ 1, & j = 0, \text{if } (t-e) \bmod m \neq 0 \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where $p(j)$ is the probability that a frame of j packets long arrives, L is the maximum number of packets allowed in a frame, and e is the time slot at which the first frame arrives.

B. System Delay Analysis

Based on the above traffic models, we first derive the unfinished work distribution for an observed connection under a given transfer rate, followed by the analysis of the maximum system delay.

1) Unfinished Work Distribution: The unfinished work is the amount of packets waiting for transmission. Packets are served at a normalized transfer rate, denoted as R. R can be reformed as a proper fraction "$(x/y), x \leq y$ $x, y \in$ positive integer" representing that, at most x, units of packets are transmitted in y time slots. The y time slots constitute a cyclic interval. Accordingly, any absolute time index, for example, time slot t, can be transformed as the hth time slot of the fth cyclic interval. Thus, $a^{(t)}(j)$ and $b^{(t)}(j)$ defined in the above subsection can be rewritten as $a^{(f,h)}(j)$ and $b^{(f,h)}(j)$. The relationships among random variables are depicted in Fig. 3.

Let $U^{(n,i)}$ represent the amount of unfinished work for an observed connection at the beginning of the ith time slot of the nth interval. Meanwhile, the number of packets $A^{(n,j)}$ arrive from the observed connection. The amount of unfinished work becomes $\overline{U}^{(n,i)}$. During the ith time slot, the amount of work $V^{(i)}$ is served if $\overline{U}^{(n,i)}$ is not less than $V^{(i)}$. Consequently, $U^{(n,i+1)}$ unfinished work is left at the beginning of the next time slot. Notice that the number of arrival packets are allowed to be greater than one unit, but, at most, one unit of packets is served in a time slot. Therefore, $\Sigma_1^y V^{(i)} = x$ for $y \geq x$.

Assuming $U^{(0,1)} = \overline{U}^{(0,1)} = 0$. According to Fig. 3, we simply get

$$\overline{U}^{(n,1)} = \begin{cases} U^{(n,1)} + A^{(n,1)}, & \text{if } U^{(n,1)} + A^{(n,1)} \leq K \\ U^{(n,1)}, & \text{if } U^{(n,1)} + A^{(n,1)} > K \end{cases} \quad (4)$$

where K is the size of the unfinished work buffer. Since $U^{(n,1)}$ is equal to $\overline{U}^{(n,1)}$ plus the amount of served work in this time slot, $V^{(1)}$, one gets

$$U^{(n,2)} = \max(\overline{U}^{(n,1)} - V^{(1)}, 0). \quad (5)$$

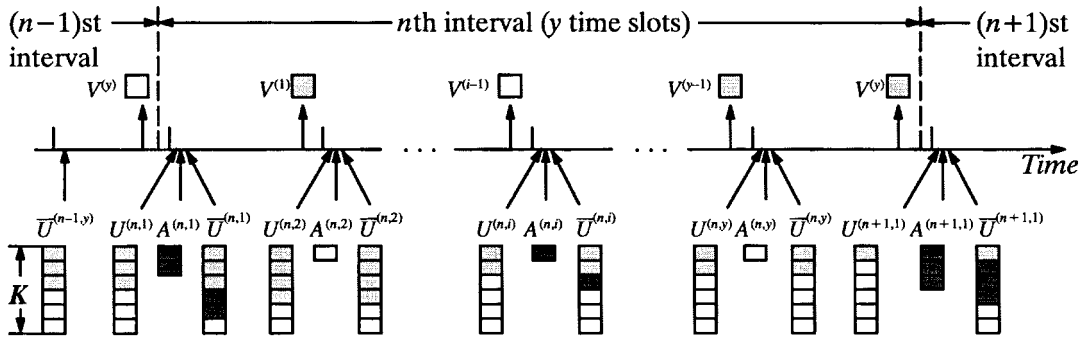


Fig. 3. The relationships among random variables.

By reasoning as above, we have the following equations:

$$\overline{U}^{(n,i)} = \begin{cases} U^{(n,i)} + A^{(n,i)}, \\ \quad 1 \le i \le y, & \text{if } U^{(n,i)} + A^{(n,i)} \le K \\ U^{(n,i)} \\ \quad 1 \le i \le y, & \text{if } U^{(n,i)} + A^{(n,i)} > K \end{cases} \tag{6}$$

$$U^{(n,i+1)} = \max\left(\overline{U}^{(n,i)} - V^{(i)}, 0\right), \qquad 1 \le i \le y-1. \tag{7}$$

Let $u^{(n,i)}(j)$ and $\overline{u}^{(n,i)}(j)$ be the PMF of $U^{(n,i)}$ and $\overline{U}^{(n,i)}$. That is, $u^{(n,i)}(j) = \text{Prob}[U^{(n,i)} = j]$ and $\overline{u}^{(n,i)}(j) = \text{Prob}[\overline{U}^{(n,i)} = j]$. In addition, let $a^{(n,i)}(j)$ and $b^{(n,i)}(j)$ be the PMF of $A^{(n,i)}$ for an audio and a video connection, respectively. Thus, the distributions for audio and video connections are given, according to (6), by

$$\overline{u}^{(n,i)}(j) = \sum_{j_1=0}^{K} \sum_{j_2=0}^{1} u^{(n,i)}(j_1) \cdot a^{(n,i)}(j_2), \qquad 1 \le i \le y$$
$$\begin{cases} j = j_1 + j_2, & \text{if } j_1 + j_2 \le K \\ j = j_1, & \text{if } j_1 + j_1 > K \end{cases} \tag{8}$$

$$\overline{u}^{(n,i)}(j) = \sum_{j_1=0}^{K} \sum_{j_2=0}^{1} u^{(n,i)}(j_1) \cdot b^{(n,i)}(j_2), \qquad 1 \le i \le y$$
$$\begin{cases} j = j_1 + j_2, & \text{if } j_1 + j_2 \le K \\ j = j_1, & \text{if } j_1 + j_1 > K \end{cases} \tag{9}$$

respectively. $v^{(i)}(j)$ is the PMF of $V^{(i)}$, i.e., $v^{(i)}(j) = \text{Prob}[V^{(i)} = j]$.. Thus, the distributions are given, according to (7), by

$$u^{(n,i+1)}(j) = \pi_0(\overline{u}^{(n,i)}(j+l) * v^{(i)}(l)) \qquad 1 \le i \le y-1, 0 \le j \le K. \tag{10}$$

where $\pi_0(\cdot)$ is an operator representing the maximum function of probability distributions [37]. The steady-state probability of the amount of unfinished work observed from the arrival packet in the ith time slot of a cyclic interval, denoted as $u^i(j)$, is expressed as

$$u^i(j) = \lim_{n_i \to \infty} u^{(n,i)}(j), \qquad 1 \le i \le y, 0 \le j \le K. \tag{11}$$

2) System Delay Distribution: We now derive the system delay distribution taking the VBR video as an example. The system delay is composed of three delays incurred at the sender, the network, and the receiver. In a delay-guaranteed network and a receiver with all its capacity preallocated, the network and receiver delays can be assumed to be constants. In the following analysis, we first derive the sender delay distribution based on the unfinished work distribution given in the previous subsection. Moreover, the service discipline is assumed to be first come first served (FCFS).

After the system reaches the steady state, the observed frame arrives at the beginning of the ith time slot in a cyclic interval, as shown in Fig. 4. All arriving packets behind the observed frame are ignored owing to the fact that the sender delay of the observed frame is unaffected. Let $U'^{(i)}$ and $\overline{U'}^{(i)}$ denote the amount of unfinished work at the ith time slot in a cyclic interval before and after the observed frame has arrived, respectively. The relationship can be expressed as

$$\overline{U'}^{(i)} = U'^{(i)} + l \tag{12}$$

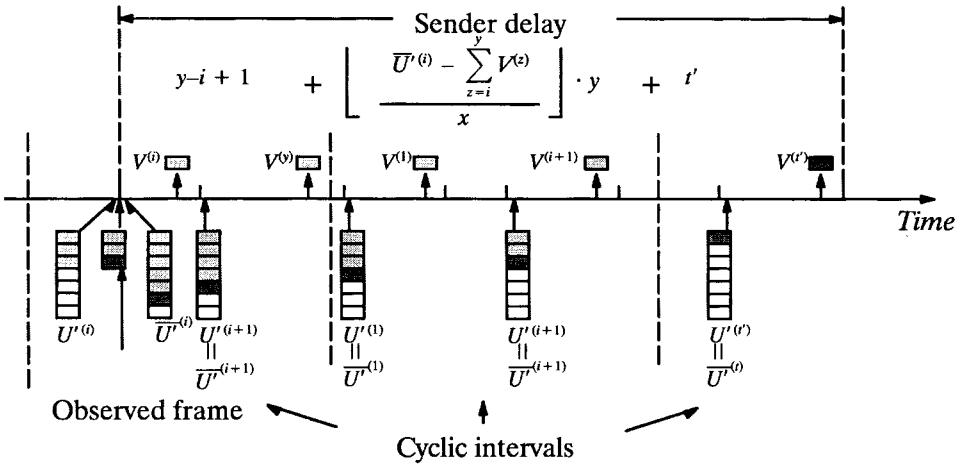
where l is the number of packets in the observed frame. The sender delay distribution is now separately derived, considering whether or not the unfinished work can be consumed within the current cyclic interval. In the first case [Fig. 4(a)], in which the unfinished work cannot be consumed within the current interval, the sender delay $W'^{(i)}$ is composed of three time durations. They are the time period between the tested packet arrival and the end of the current interval, the time duration of a number of complete cyclic intervals spending to serve the remaining work, and the service time serving the final leftover work. That is,

$$W'^{(i)} = (y - i + 1) + \left\lfloor \frac{\overline{U'}(i) - \sum_{z=i}^{y} V^{(z)}}{x} \right\rfloor \cdot y + t'. \tag{13}$$

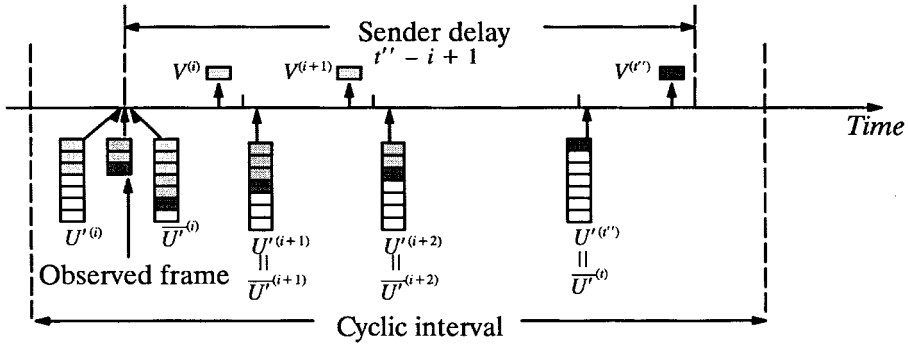
where t' is the smallest integer satisfying the following equation:

$$\sum_{z=0}^{t'} V^{(z)} = \left(\overline{U}'^{(i)} - \sum_{z=i}^{y} V^{(z)}\right) \bmod x. \tag{14}$$

In the second case [Fig. 4(b)], in which the unfinished work can be consumed within the current interval, the waiting time



(a)



(b)

Fig. 4. Sender delay. (a) Unfinished work cannot be consumed within the current cyclic interval. (b) Unfinished work can be consumed within the current cyclic interval.

$W''^{(i)}$ can be given as

$$W''^{(i)} = t'' - i + 1 \tag{15}$$

where t'' is the smallest integer satisfying the following equation:

$$\sum_{z=i}^{t''} V^{(z)} = \overline{U}'^{(i)}. \tag{16}$$

As a whole, the sender delay W is summarized as

$$W = \begin{cases} W', & \overline{U}'^{(i)} > \sum_{z=i}^{y} V^{(z)} \\ W'', & \overline{U}'^{(i)} \le \sum_{z=i}^{y} V^{(z)}. \end{cases} \tag{17}$$

Let $w'^{(i)}(k)$ and $w''^{(i)}(k)$ be the PMF's of $W'^{(i)}$ and $W''^{(i)}$, respectively. That is, $w'^{(i)}(k) = \text{Prob}[W'^{(i)} = k]$ and $w''^{(i)}(k) = \text{Prob}[W''^{(i)} = k]$. In addition, let $u^{(i)}(j)$ and $\overline{u'}^{(i)}(j)$ be the PMF's of $U^{(i)}$ and $\overline{U'}^{(i)}$. That is, $u^{(i)}(j) = \text{Prob}[U'^{(i)} = j]$ and $\overline{u'}^{(i)}(j) = \text{Prob}[\overline{U'}^{(i)} = j]$, where $u^{(i)}(j)$ is given by (11). Thus, the distribution is given, according to (12), by

$$\overline{u'}^{(i)}(j) = u^{(i)}(j) * p(j). \tag{18}$$

The sender delay distribution $w^{(i)}(k)$ for the tested packet arrival at the ith time slot of a cyclic interval, according to (17), becomes

$$w^{(i)}(k) = \begin{cases} w'(k) = \overline{u'}^{(i)}(j), \\ \quad \text{where } k = (y - i + 1) + \left\lfloor \dfrac{j - \sum_{z=i}^{y} v^{(z)}(1)}{x} \right\rfloor \\ \quad \cdot y + t', \text{if } j > \sum_{z=i}^{y} v^{(z)}(1); \\ w''^{(i)}(k) = \overline{u'}^{(i)}(j), \\ \quad \text{where } k = t'' - 1 + 1, \text{if } j \le \sum_{z=i}^{y} v^{(z)}(1). \end{cases} \tag{19}$$

The sender delay distribution $w(k)$ in the sender for the observed connection is

$$w(k) = \sum_{i=0}^{y-1} w^{(i)}(k) \cdot q_i \tag{20}$$

where q_i is the probability of the observed frame arrival at the ith time slot of a cyclic interval. q_i is derived by the

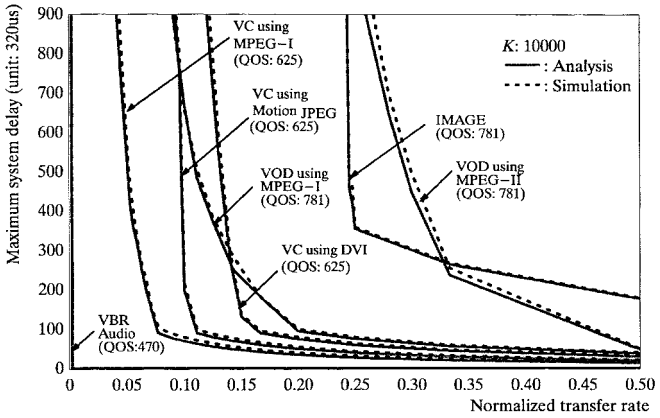


Fig. 5. Maximum system delay for VBR applications.

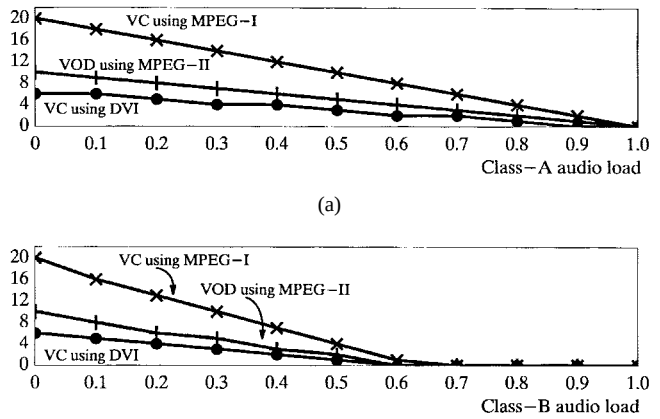


Fig. 6. Impact of applications on class-B video connections. (a) Under class-A loads. (b) Under class-B loads.

following equation:

$$q_i = \begin{cases} \dfrac{\gcd(m,y)}{y}, & 1 \le i \le y, \quad \text{if } i \bmod \gcd(m,y) = 0 \\ 0, & \text{otherwise} \end{cases} \tag{21}$$

where $\gcd(m,y)$ is equal to the greatest common division of m and y [38].

The maximum system delay distribution $w_{\text{system}}(k)$ is

$$w_{\text{system}}(k+2) = w(k) \tag{22}$$

assuming that each of the network and receiver delays is assumed to one time slot in length. The maximum system delay is the smallest integer k_{system} to satisfy the following equation:

$$\sum_{k=0}^{k_{\text{system}}} w_{\text{system}}(k) > 0.999999, \qquad 0 \le k \le K. \tag{23}$$

We have so far shown the derivation of the maximum system delay for VBR video. The same analysis can be applied for a VBR audio by replacing $p(1) = 1$ in (18) and m by g in (21). Consequently, according to the above analysis, any required maximum system delay, as will be shown next, corresponds to a minimum transfer rate $R_{\min}$, guaranteeing the achievement of such maximum delay for any newly arrived connection.

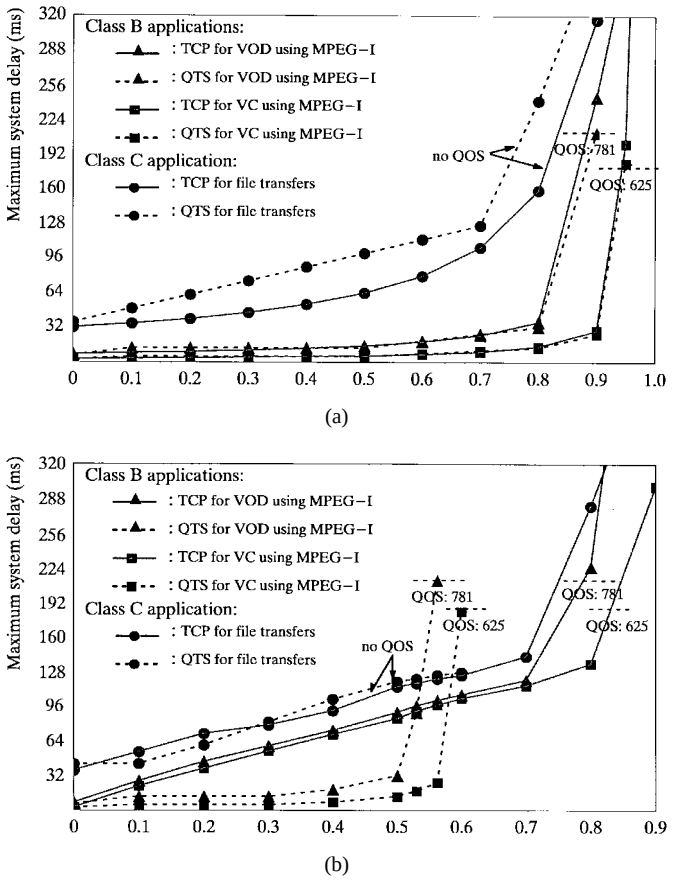


Fig. 7. Maximum system delay comparisons. (a) Under class-A loads. (b) Under class-B loads.

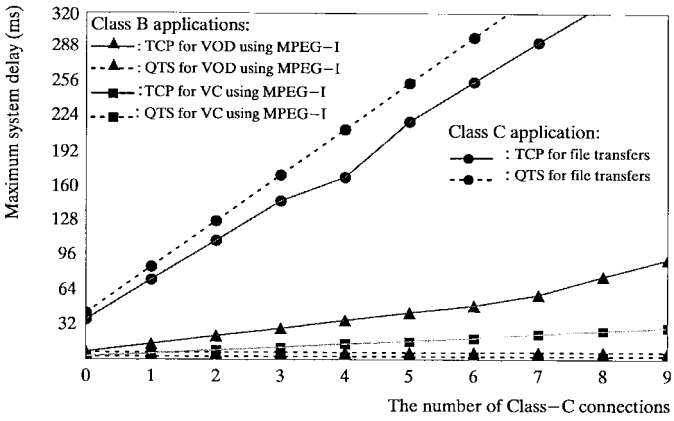


Fig. 8. The maximum system delay under a variety of class-C connections.

IV. Analytic and Simulation Results

We are now at the stage of determining $R_{\min}$ for VBR applications based on analytic computational results. The analytic computation terminates if all entries of matrix $|u^{(n_i,i)}(j) - u^{(n_1-1,i)}(j)|$ are smaller than 10^{-6}. We also ran time-based simulation using the same set of parameters as analysis, such as the traffic model, the scheduling discipline, and the buffer size. Simulation terminates if the maximum system delay remains

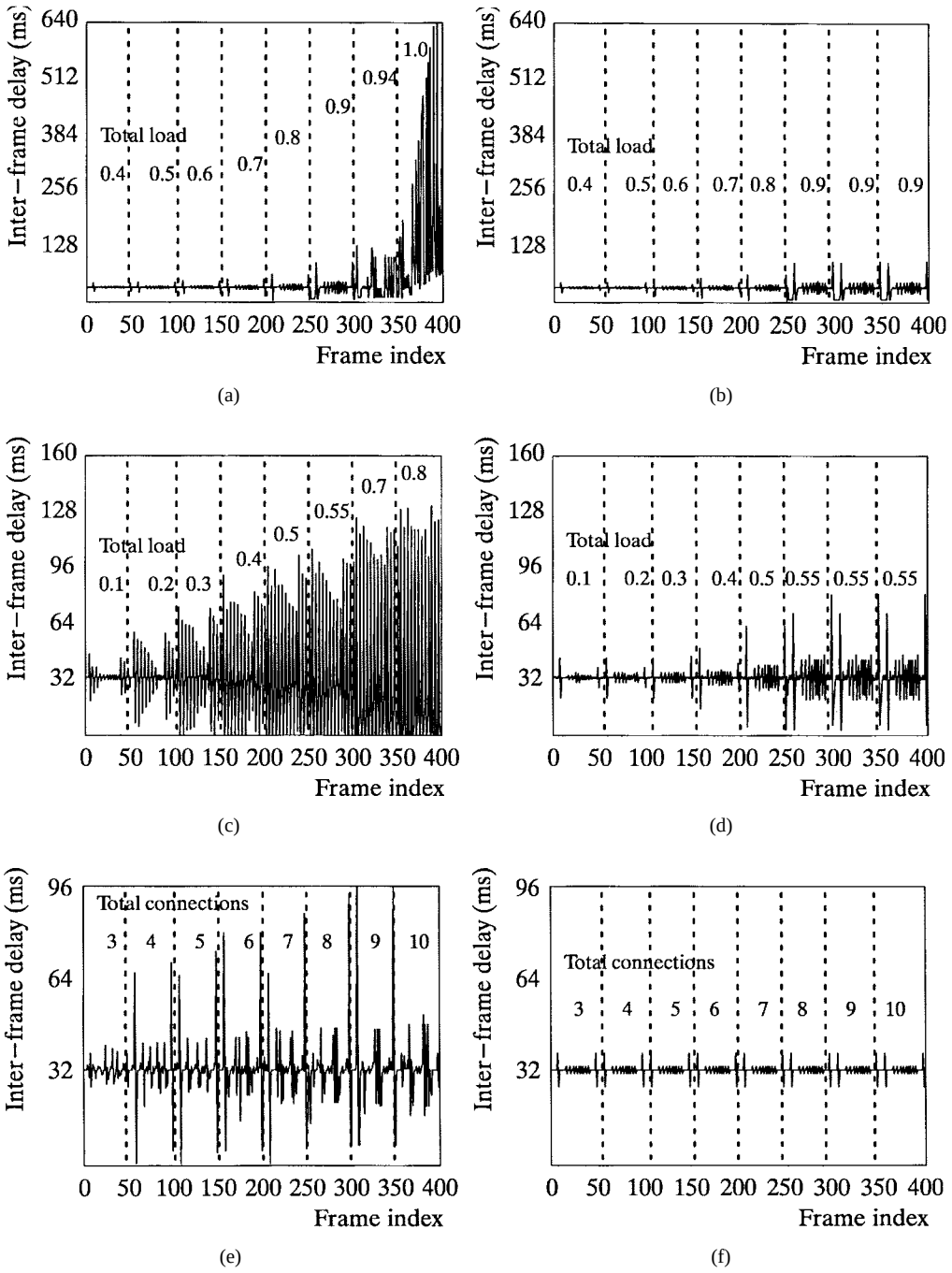


Fig. 9. Interframe delay comparison. (a) TCP: under class-A loads. (b) QTS: under class-A loads. (c) TCP: under class-B loads. (d) QTS: under class-B loads. (e) TCP: under class-C connections. (f) QTS: under class-C connections.

the same for 10^6 time ticks. Fig. 5 shows the maximum system delay as a function of the normalized transfer rate for VBR applications specified in Table I. The figure demonstrates that analytic results profoundly agree with simulation results. Moreover, the maximum system delay declines as the transfer rate increases. Based on these results, $R_{\min}$ can be determined as follows. For instance, $R_{\min}$ can be allocated as a rate of 1/20 for MPEG-I-based VC achieving a QOS delay of 625 time slots, as shown in the second curve from the left in Fig. 5.

Fig. 6 demonstrates the impact of granted applications (class A or class B) on the number of accepted class-B video connections. Due to the QOS guarantee for both class-A and class-B applications, the number of accepted class-B video connections declines as the load of class-A or class-B applications increases. More significantly, in comparison with Fig. 6(a), Fig. 6(b) shows that the number of class-B connections allowed under a given amount of the VBR class-B load is less than that under the same amount of the CBR class-A load. This is because an increase in traffic burstiness results in a decrease in statistical multiplexing gain [43].

We now draw performance comparisons between the QTS and TCP in terms of the maximum system delay, throughput, and the interframe delay via simulation results. In the simulation of TCP, the processing time of a packet was set as 370 μs,

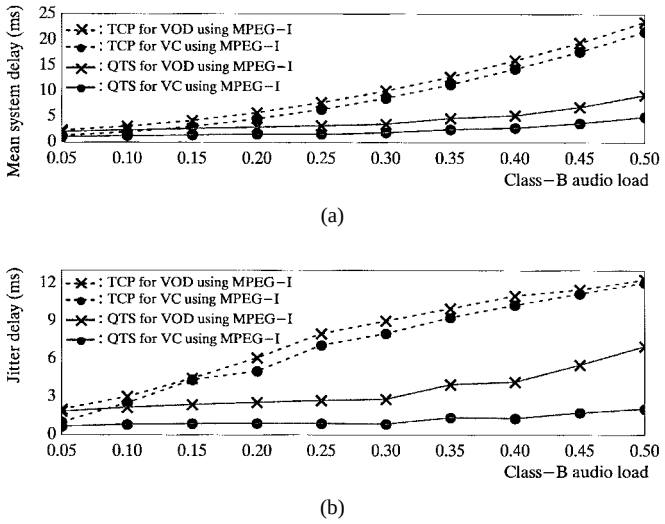


Fig. 10. Comparison of rate-based (QTS) and window-based (TCP) flow control. (a) Mean system delay. (b) Delay jitter.

including the checksum delay of 65 μs [39]. In the simulation of the QTS, the processing time of a packet for class-A and class-B applications was set as 320 μs due to the omission of the checksum processing. As for class-C applications, the processing time of a packet was set as 390 μs, owing to the inclusion of the checksum processing and bandwidth allocation overhead in the QTS. Furthermore, the window size in TCP was set to 4 kbytes [44] in length.

Fig. 7 illustrates the maximum system delay under various loads of class-A and class-B audio applications. As shown in Fig. 7(a), the QTS performs as well as TCP for class-B applications if the class-A audio load is within the allowance of the QTS. It is worth noting that the QTS stringently imposes a limit in the class-A audio load in an effort to guarantee the maximum system delay for accepted class-B applications. In contrast, TCP unlimitedly accepts class-A audio applications, resulting in an unbounded maximum system delay for class-B applications. Moreover, as was expected, the class-C applications in the QTS suffer higher maximum system delay than TCP owing to the best-effort transferring nature for class-C packets in the QTS. Fig. 7(b) shows the maximum system delay under class-B audio loads. As shown in the figure, TCP incurs higher maximum system delay than the QTS for class-B applications under light and medium loads of class-B audio applications. More significantly, the QTS performs as well as TCP for the class-C application in this case.

Fig. 8 shows the maximum system delay under a variety of class-C connections. As shown in the figure, the maximum system delay for class-B applications in TCP increases with the number of granted class-C connections. However, the QTS provides a near-constant maximum system delay for class-B applications, regardless of the number of accepted class-C connections. In comparison with TCP, the QTS achieves superior performance for class-B applications due to the preallocation of the transport bandwidth. Owing to the extra processing overhead, the QTS imposes higher system delay for the class-C applications than TCP.

Fig. 9 displays the interframe delay of an MPEG-I-based VOD application under various class-A and class-B loads and the number of class-C connections. As shown in Fig. 9(b), (d), and (f), the QTS guarantees a bounded interframe delay and variance under diverse traffic loads. In contrast, TCP exhibits unbounded interframe delay and large delay variance under heavier traffic loads, as depicted in Fig. 9(a), (c), and (e). Notice that, as shown in Fig. 9(b) and (d), the QTS exhibits minor delay fluctuation resulting from the sharing of the bandwidth with class-A and class-B applications.

V. Transport Protocol Module

The transport protocol module of the QTS performs flow control, in addition to traditional transport layer functions [15], such as error control, connection management, addressing, and multiplexing. Since the discussion of these traditional transport functions is beyond the scope of the paper, we only focus on the design of the rate-based flow control mechanism. In essence, the QTS performs rate-based flow control for class-A and class-B traffic based on $R_{\min}$ predetermined by the bandwidth allocator. Fig. 10 depicts performance comparisons between rate-based flow control of the QTS and window-based flow control of TCP, in terms of mean system delay and delay jitter. As shown in the figure, although both system delay and delay jitter increase with the load of class-B audio under any flow control mechanism, the QTS results in bounded and considerably low delay compared to TCP.

As was previously described, the transport protocol module of the QTS offers error control (including checksum) only for loss-sensitive applications. The lack of error control for delay-sensitive applications yields the reduction of the system delay and the increase of system throughput.

VI. Conclusions

This paper has presented a QTS which offers diverse QOS's for broad-band multimedia and industrial applications. The QTS is composed of a bandwidth allocator and a transport protocol module. It supports three classes of applications demanding different bit rates, delay sensitivity, and loss sensitivity. The bandwidth allocator of the QTS intelligently manages the allocation of transport-layer bandwidth. The transport protocol module of the QTS performs rate-based flow control for delay-sensitive applications based on the transfer rate predetermined by the bandwidth allocator. As a result, the QTS stringently guarantees the maximum system delay for accepted class-A and class-B applications at the expense of imposing inevitable blocking of such connections. Furthermore, the paper has shown that the QTS guarantees a bounded inter-frame delay and delay variance under diverse traffic loads, while TCP exhibits severe delay fluctuation under heavier loads.

References

[1] H. Armbrüster and K. Wimmer, "Broad-band multimedia applications using ATM networks: High-performance computing, high-capacity storage, and high-speed communication," *IEEE J. Select. Areas Commun.*, vol. 10, pp. 1382–1396, Dec. 1992.

[2] J. Rosenberg, R. E. Kraut, L. Gomez, and C. A. Buzzard, "Multimedia communications for users," *IEEE Commun. Mag.*, vol. 30, pp. 20–36, May 1992.
[3] I. W. Habib and T. N. Saadawi, "Multimedia traffic characteristics in broad-band networks," *IEEE Commun. Mag.*, vol. 30, pp. 48–54, July 1992.
[4] P. V. Rangan, H. M. Vin, and S. Ramanathan, "Designing an on-demand multimedia service," *IEEE Commun. Mag.*, vol. 30, pp. 56–64, July 1992.
[5] S. Ramanathan and P. V. Rangan, "Adaptive feedback techniques for synchronized multimedia retrieval over integrated networks," *IEEE J. Select. Areas Commun.*, vol. 1, pp. 246–259, Apr. 1993.
[6] S. Ramanathan and P. V. Rangan, "Feedback techniques for intra-media continuity and inter-media synchronization in distributed multimedia systems," *Comput. J.*, vol. 36, no. 1, pp. 19–31, Feb. 1993.
[7] C. Nicolaou, "An architecture for real-time multimedia communication systems," *IEEE J. Select. Areas Commun.*, vol. 8, pp. 391–400, Apr. 1990.
[8] D. Shepherd and M. Salmony, "Extending OSI to support synchronization required by multimedia applications," *High Speed Networks*, vol. 13, no. 7, pp. 399–406, Sept. 1990.
[9] R. Handel and M. N. Huber, *Integrated Broadband Networks—An Introduction to ATM-Based Networks*. Reading, MA: Addison-Wesley, 1991.
[10] M. Prycker, *Asynchronous Transfer Mode Solution for Broadband ISDN*, 3rd ed. Englewood Cliffs, NJ: Prentice Hall, 1995.
[11] D. E. Mcdysan and D. L. Spohn, *ATM-Theory and Application*. New York: McGraw-Hill, 1994.
[12] I. W. Habib and T. N. Saadawi,"Controlling flow and avoiding congestion in broad-band networks," *IEEE Commun. Mag.*, vol. 29, pp. 46–53, Oct. 1991.
[13] M. C. Yuang, J. C. Liu, and C. L. Shay, "BATS: A high-performance transport system for broadband applications," in *Proc. Local Computer Networks*, Oct. 1994, pp. 448–455.
[14] A. Netravali, W. D. Roome, and K. Sabnani, "Design and implementation of a high-speed transport protocol," *IEEE Trans. Commun.*, vol. 38, pp. 2010–2024, Nov. 1990.
[15] D. Comer, *Internetworking with TCP/IP, Volume 1: Principles, Protocols, and Architecture*. Englewood Cliffs, NJ: Prentice-Hall, 1991.
[16] W. Stallings, W., *Handbook of Computer Communications Standards, Volume 1: The Open Systems Interconnection (OSI) Model and OSI-Related Standards*. Indianapolis, IN: Sams, 1989.
[17] D. Cheriton,"VMTP: A transport protocol for the next generation of communication systems," in *Proc. ACM SIGCOMM'86*, 1986, pp. 406–415.
[18] D. D. Clark, M. L. Lambert, and L. Zhang, "NETBLT: A high throughput transport protocol," in *Proc. ACM SIGCOMM'87 Workshop*, 1987, pp. 353–359.
[19] W. T. Strayer, B. J. Dempsey, and A. C. Weaver, *XTP: The Xpress Transport Protocol*. Reading, MA: Addison-Wesley, 1992.
[20] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, *"RTP: A transport protocol for real-time applications"*, Internet draft RFC 1889, Jan. 1996.
[21] T. F. La Porta and M. Schwartz, "The multiStream protocol: A highly flexible high-speed transport protocol," *IEEE J. Select. Areas Commun.*, vol. 11, pp. 519–530, May 1993.
[22] T. F. La Porta and M. Schwartz, "Performance analysis of MSP: Feature-rich high-speed transport protocol," *IEEE/ACM Trans. Networking*, vol. 1, pp. 740–753, Dec. 1993.
[23] M. E. Anagnostou, M. E. Theologou, K. M. Vlakos, D. Tournis, and E. N. Protonotarios, "Quality of service requirements in ATM-based B_ISDN's," *Comput. Commun.*, vol. 14, no. 4, pp. 197–204, May 1991.
[24] D. Dubois, N. D. Georganas, and E. Horlait, "A QOS selector for multimedia applications on ATM networks," in *Proc. IEEE Int. Conf. Communications*, 1994, pp. 160–164.
[25] D. B. Hehmann, M. G. Salmony, and H. J. Stuttgen, "Transport services for multimedia applications on broadband networks," *Comput. Commun.*, vol. 13, no. 4, pp. 197-203, May 1990.
[26] L. Wilson, "Requirements for pervasive multiparty desktop video collaboration," in *Multimedia Handbook*. New York: McGraw–Hill, 1994, pp. 49.1–49.23.
[27] J. A. Zebarth, "Let me be me," in *Proc. IEEE GLOBECOM*, 1993, pp. 389–393.
[28] M. Evans, "Networks for multimedia and collaborative computing," in *Multimedia Handbook*. New York: McGraw–Hill, 1994, pp. 41.1–41.13.
[29] R. Harris, "Real-world applications for MPEG digital video," in *Multimedia Handbook*. New York: McGraw–Hill, 1994, pp. 31.1–31.6.
[30] W. Tawbi, F. Horn, E. Horlait, and J. B. Stefani, "Video compression standards and quality of service," *Comput. J.*, vol. 36, no. 1, pp. 43–54, 1993.
[31] J. Keyes, "A discussion of standards," in *Multimedia Handbook*. New York: McGraw–Hill, 1994, pp. 15.1–15.12.
[32] P. Uppaluru, "Networking multimedia on standard data networks," *Multimedia Handbook*. New York: McGraw–Hill, 1994, pp. 42.1–42.7.
[33] J. C. Liu, C. L. Shen, Y. P. Tseng, J. N. Yang, and M. C. Yuang, "Design and prototype of real-time multiparty teleconference system over high performance transport platform," Dep. Comput. Sci. Inform. Eng., Nat. Chiao Tung Univ., Hsinchu, Taiwan, R.O.C., Tech. Rep. C84056, 1994.
[34] W. Tawbi and E. Horlait, "A model for expression and support of multimedia applications requirements at end-systems," in *Proc. 1994 IEEE Region 10 9th Annu. Int. Conf.*, Aug. 1994, pp. 842–846.
[35] S. J. Su, "MPEG player," Acer Sertek Inc., Taiwan, R.O.C., *Service for Technology*, 1994, pp. 34–37.
[36] A. Campbell, G. Coulson, and F. Garcla, "Integrated quality of service for multimedia communications," in *Proc. IEEE INFOCOM*, 1993, pp. 732–739.
[37] M. Murata, Y. Oie, T. Suda, and H. Miyahara, "Analysis of a discrete-time single-server queue with bursty input for traffic control in ATM networks," *IEEE J. Select. Areas Commun.*, vol. 8, pp. 447–458, Apr. 1990.
[38] H. R. Kenneth, *Elementary Number Theory and Its Applications*, 2nd ed. Reading, MA: Addison-Wesley, 1988.
[39] C. Papadopoulos and G. M. Parulkar, "Experimental evaluation of SUNOS IPC and TCP/IP protocol implementation," *IEEE/ACM Trans. Networking*, vol. 1, pp. 199–216, Apr. 1993.
[40] *Information Technology—Generic Coding of Moving Pictures and Associated Audio*, ISO/IEC 13818-1,2,3, Recommendation H.262.
[41] *Information Technology—Coding of Moving Pictures and Associated Audio for Digital Storage Media at Up to About 1.5 Mbits/sec*, ISO/IEC 11172-1,2,3.
[42] T. M. Chen, S. S. Liu, and V. K. Samalam, "The available bit rate service for data in ATM networks," *IEEE Commun. Mag.*, vol. 34, pp. 56–71, May 1996.
[43] H. Saito, *Teletraffic Technologies in ATM Networks*. Norwood, MA: Artech House, 1994.
[44] J. H. Huang and C. W. Chen, "On performance measurements of TCP/IP and its device driver," in *Proc. 17th Conf. Local Computer Networks*, Oct. 1992, pp. 568–575.

Maria C. Yuang (M'91), for a photograph and biography, see this issue, p. 3.

Jen C. Liu was born in Taiwan, R.O.C., in 1968. He received the B.S. degree in computer science and information engineering in 1991 from the National Chiao Tung University, Hsinchu, Taiwan, R.O.C., where he is currently working toward the Ph.D. degree.

His research is focused on high-speed networks, multimedia communications, high-performance transport systems, and performance modeling and analysis.