A Modular Multirate Video Distribution System—Design and Dimensioning

Yiu-Wing Leung, *Member, IEEE,* and Tak-Shing Yum, *Senior Member, IEEE*

***Abstract*—A modular architecture is proposed for distributing broadcast and switched video. The architecture consists of a set of concentration buses (or input buses), a TDM-based bus matrix and a set of distribution buses (or output buses). The transmission time in each output bus is divided into fixed size frames. Dedicated time slots in a frame are reserved for broadcast video. The remaining time slots are allocated to switched video on a first-come–first-served basis. Videos are switched via time slot assignments which determine the connections within the bus matrix. Two slot assignment algorithms are designed, one for point-to-point transmissions and the other for point-to-multipoint transmissions. The advantages of this architecture include: 1) accommodation of multirate video, 2) support of video broadcasting and multicasting, and 3) modular growth at distributed locations.**

I. Introduction

PROPOSALS for broadband video services can be categorized into broadcast services and interactive services. For broadcast services, videos are distributed to all the customers. Customers do not usually have control over the content of the broadcasts (e.g., TV programme distribution). For interactive services, video communication session is initiated by customers and video is switched to those who have requested this session (e.g., videoconferencing and video on demand [1]).

With the advances in fiber optics technology, switching rather than transmission becomes the bottleneck in broadband video services. In Hong Kong as an example, large quantities of dark fibers are in place but good quality video communication is still a rarity because currently available switching facilities cannot accommodate them economically.

In recent years, considerable effort has been made in developing high speed packet switches for broadband communications. High speed packet switches are customarily being classified into three broad types: shared-memory based, shared-medium (or shared-bus) based and space-division based [2], [3]. In particular, several high speed packet switches designed by IBM, NEC and Fujitsu use shared-buses for high speed switching. IBM's PARIS switch [4]–[5] uses a high speed shared bus for switching and it adopts an efficient bus-access policy which can accommodate variable size packets. NEC's ATOM switch [6] also uses a single shared bus but it adopts the bit-slice organization to increase the speed of the bus. For still larger switches, a multistage organization was proposed and store and forward of packets is needed in every stage. Alternatively, Fujitsu's Bus Matrix Switch [7] uses multiple shared buses which are connected in matrix form with memory located at each crosspoint of the buses. Packets contending for access to the same bus are stored in the crosspoint memories. Arbiters scan the crosspoint memories and remove packets from them.

In a previous study [8], we have proposed a TDM based multibus packet switch where the inputs and outputs are grouped for the sharing of buses and packets are switched from the inputs to the outputs using a zero memory $M \times M$ bus matrix. Based on this multibus packet switch, we propose in this paper a modular architecture for distributing broadcast and switched video and design algorithms for establishing multirate point-to-point as well as point-to-multipoint connections. This architecture has three advantages. First, multirate video can be accommodated. This can accommodate a variety of video services that have different bit rate requirements. Second, video can be broadcast or multicast to the customers. Hence, multipoint communication services (e.g., video conferencing) can be provided. Third, it can be used as a building block for constructing large video distribution networks.

II. A Modular Architecture for Video Distribution

Fig. 1 shows a modular architecture for distributing broadcast and switched videos. It consists of a *video multiplexer*, a *video switch*, a *slot assignment processor*, M *output buses* and a set of output adaptors. All of these components need to be confined to a "single box" to ensure the length of any bus is short enough for high speed transmission.

We assume that each video source has a constant bit rate. For example, compressed HDTV video without degradation has a bit rate of 44 Mbps and compressed NTSC video with VCR quality has a bit rate of 1.5 Mbps [9]. Asynchronous transfer mode can support communication services having different traffic statistics and variable bit rate transmission, but it must guarantee a small delay and packet loss rate for real-time traffic and regulate the input rate of each communication session. In this paper, we considered a specialized service network for fixed rate video transmissions. Such specialization can eliminate the overhead of guaranteeing the delay, controlling packet loss and regulating the input rate of each communication session.

All video channels are packetized into the same fixed size packets before entering the network. The transmission time in

Manuscript received February 23, 1993; revised November 22, 1993; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor J. Turner. This work was supported by two Polytechnic Research Grants funded by the Hong Kong Polytechnic, Hong Kong.

Y.-W. Leung is with the Department of Computing, Hong Kong Polytechnic, Kowloon, Hong Kong (e-mail: csyleung @comp.hkp.hk).

T.-S. Yum is with the Department of Information Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong (e-mail: yum @ie.cuhk.hk).

IEEE Log Number 9407062.

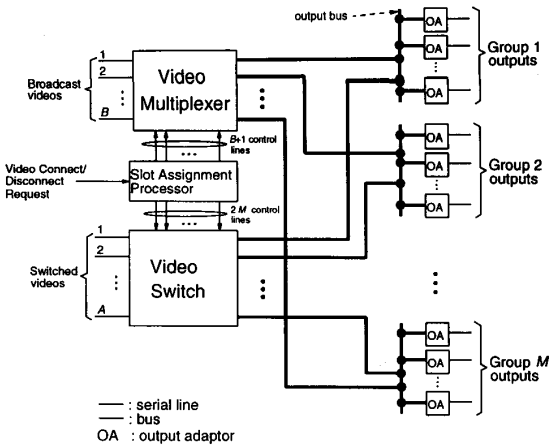


Fig. 1. A modular architecture for distributing broadcast and switched videos.

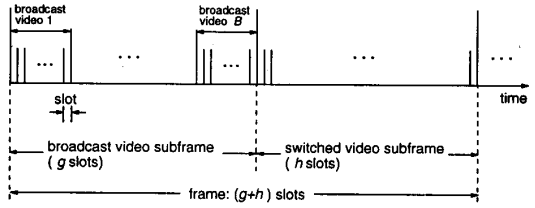


Fig. 2. Transmission frame.

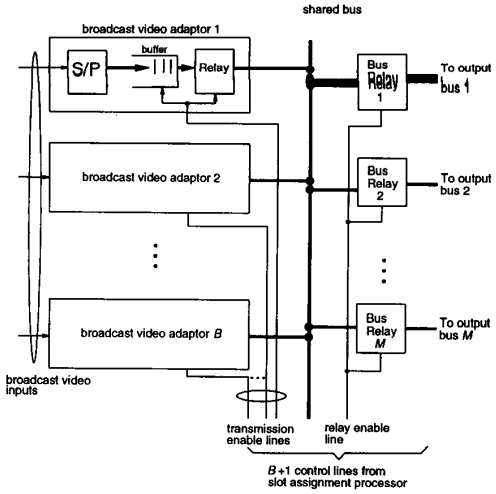


Fig. 3. Video multiplexer.

each output bus is divided into fixed size *frames* as shown in Fig. 2. A frame is divided into g slots of *broadcast video subframe* and h slots of *switched video subframe* where each slot can accommodate one video packet. The basic rate in this network is one packet per frame. A higher rate video can be accommodated by assigning multiple slots per frame. In the rest of this paper, the slot positions for broadcast video and switched video are in reference to the broadcast video subframe and the switched video subframe respectively.

A. Video Multiplexer

The video multiplexer combines up to B broadcast videos by time division multiplexing and feeds them onto the M output buses. Fig. 3 shows the block diagram of a video multiplexer. It consists of B *broadcast video adaptors* and M *bus relays*. Each broadcast video is connected to a broadcast video adaptor, which converts the serial input into parallel bit streams, stores the packets in the buffer and transmits a packet when its control input is HIGH (i.e., "1"). The bus relay is a high input impedance two-state device which copies the content in the input bus to the output bus when its control input is HIGH and disconnects the output bus from the input bus when its control input is LOW. Because these relays have high input impedance, they can ensure a certain fanout at the output buses.

The $B+1$ control lines from the slot assignment processor consist of B transmission enable lines and 1 relay enable line. Each transmission enable line is connected to the control input of a broadcast video adaptor and the relay enable line is connected to the control inputs of all the bus relays. In the broadcast video subframe, the relay enable line is HIGH so that all the broadcast video adaptors can transmit to all the output buses. Let the ith broadcast video require b_i slots per frame. Then, the first transmission enable line is set to HIGH for b_1 slots to enable broadcast video adaptor 1 to transmit packets to the output buses and then the second transmission enable line is set to HIGH for the next b_2 slots, etc.

B. Video Switch

Fig. 4 shows the video switch. The switched-video inputs are partitioned into M groups of L inputs each. There are two reasons for grouping the inputs. First, the data transfer rate of a bus is usually much higher than the data rate of a switched video channel (e.g., the bus in the IBM PARIS prototype is operated at 6 Gbps [10] and the data rate of a compressed video at VCR quality is 1.5 Mbps [9]). Therefore, a bus can serve many switched video channels. Second, very often, only some of the input ports have traffic. Many ports can therefore share a bus by statistical multiplexing. In section V, the group size dimensioning problem for a specified blocking performance is posed and solved. Similarly, a specific output bus can serve a group of customers.

Every switched video is connected to a *switched video adaptor*. Group i adaptors are connected to the ith horizontal bus, and group j output ports are connected to the jth vertical bus. The M horizontal buses are connected to the M vertical buses in bus matrix form, with a total of M^2 *switching elements*. The switching element placed at the crosspoint of the ith horizontal bus and the jth vertical bus is denoted by S_{ij}. A common clock line is connected to all the switched video adaptors and switching elements (not shown in Fig. 4). Video channels are switched to the outputs via the switching elements which are controlled by the slot assignment processor.

When the current slot assignment needs to be changed, the slot assignment processor sends the new slot assignment at the beginning of a frame to all the switched video adaptors and switching elements through the $2M$ control lines (Fig. 4).

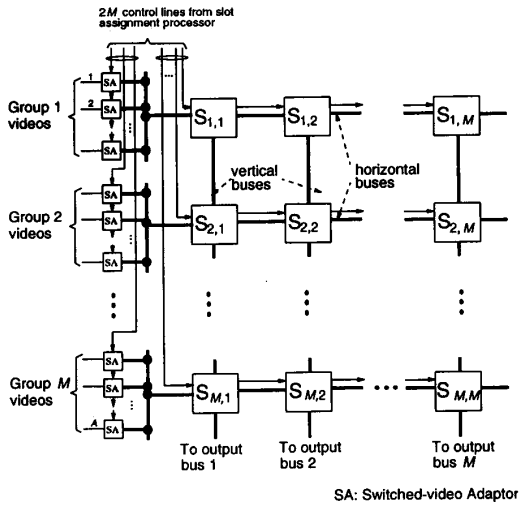


Fig. 4. Video switch.

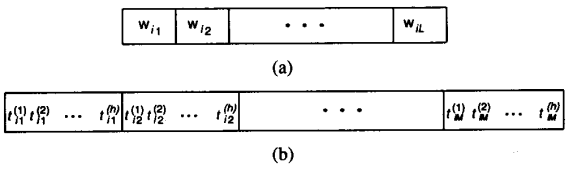


Fig. 5. Slot assignment signal unit formats.

These new slot assignments will be used in the next frame. Fig. 5(a) shows the slot assignment signal unit format for the group i switched video adaptors. w_{ij} is a bit map of the slot positions assigned to the jth input port of group i. Thus $h = 10$ and $w_{34} = 0001001000$ means that the 4th input port of group 3 can transmit at slots 4 and 7. When a new slot assignment arrives at a switched video adaptor, it removes the first h bits and passes the remaining bits to the next switched video adaptor. Fig. 5(b) shows the slot assignment signal unit format for the switching elements on the ith horizontal bus. Here $t_{ij}^{(k)} = 1$ indicates that S_{ij} is to connect the ith horizontal bus to the jth vertical bus in slot k; $t_{ij}^{(k)} = 0$ means otherwise. When a new slot assignment arrives at a switching element, it removes the first h bits and passes the remaining bits to the successive switching elements. $\{w_{ij}\}$ and $\{t_{ij}^{(k)}\}$ are determined in Sections III and IV.

Fig. 6(a) shows a switched video adaptor. It converts the serial input into parallel bit streams and stores the packets in the buffer. When the controller receives a slot assignment from the slot assignment processor, it removes and stores the first h bits and passes the remaining bits to the next switched video adaptor. It then initiates packet transmissions according to the bit patterns received.

Fig. 7 shows the realization of a switching element. The relay is a high input impedance two-state element. When its control input is HIGH, it passes the signal from the input to the output; when its control input is LOW, it disconnects the output from the input. The high input impedance of these relays can ensure a certain fanout at the output buses. The shift

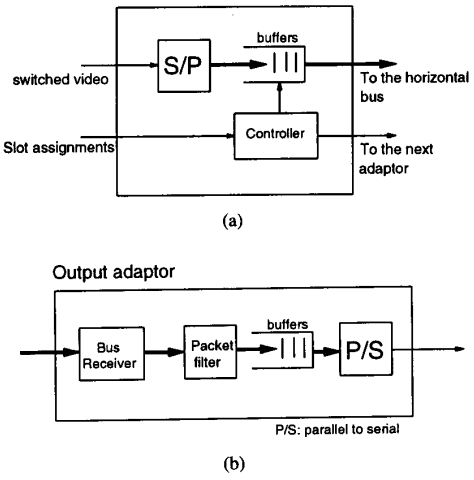


Fig. 6. Switched video adaptor and output adaptor.

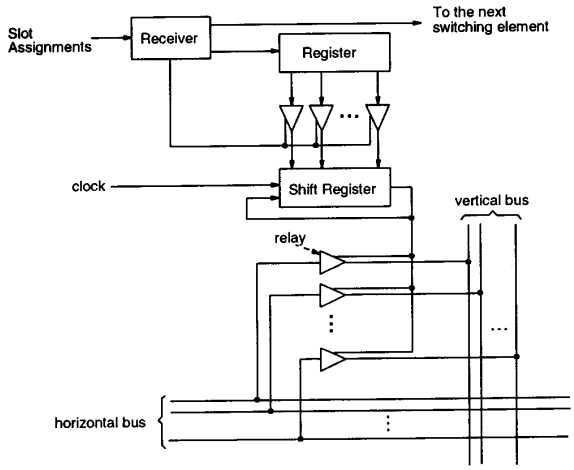


Fig. 7. Switching element.

register stores a set of bits that determines when to connect the horizontal bus to the vertical bus. When a clock pulse arrives, all the bits shift to the left by one position and the least significant bit is shifted to the most significant bit position. If the current least significant bit is HIGH, the relays copy the content in the horizontal bus to the vertical bus. When a new slot assignment arrives, the receiver extracts the first h bits, stores them in the register and passes the remaining bits to the successive switching elements. At the end of the current frame, the receiver enables the relays connecting the two registers, allowing the updating of the slot assignment in the shift register.

By controlling the switching elements, a switched video can be sent to one or more output queues. Fig. 8 shows an illustrative example in which group 2 videos are connected to output buses 2, 4, and 5, group 4 videos are connected to output bus 3 and group 5 videos are connected to output bus 1.

By incorporating a video multiplexer and a video switch in the same architecture, the network only needs to provide

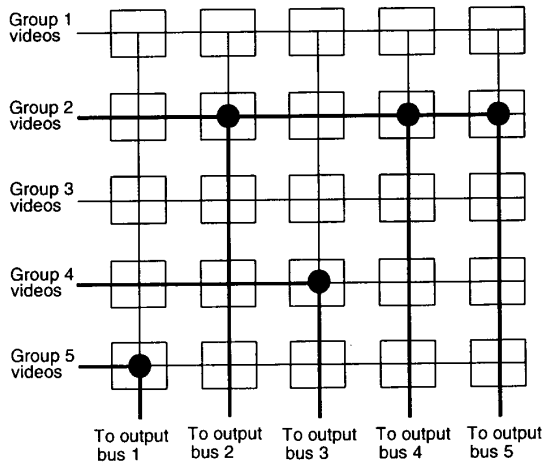


Fig. 8. An example of multicasting.

one type of output interface (i.e., the output adaptors) for the customers to receive both broadcast and switched videos. An added advantage is that the number of slots allocated for broadcast video and the number of slots allocated for switched video can be changed with the change of traffic composition during different times of the day.

C. Slot Assignment Processor

The slot assignment processor records the number of slots required by every broadcast and switched video. It also records the output group numbers of the switched videos. It coordinates broadcast video distribution through the $B+1$ control lines that run to the video multiplexer. When a customer requests a new switched video session, the slot assignment processor determines whether it can be admitted. If this new session can be admitted, it sends the new slot assignment to the video switch through the $2M$ control lines.

D. Output Adaptor

Fig. 6(b) shows the block diagram of an output adaptor. The bus receiver is a high input impedance device that copies the content in its input to its output. The packet filter inspects the circuit number contained in the header of every incoming packet and accepts packets destined for it. The received packets are stored in the buffer and then converted into serial bit streams for transmission.

The fanout problem [11] limits the maximum number of output adaptors that can be connected to each output bus. This problem can be solved by connecting the output bus to a set of bus relays of high input impedance and each bus relay drives a subgroup of output adaptors in an output group.

III. Slot Assignment for Point-to-Point Transmissions

In this section, we consider the case where every switched video packet is sent to one output group.

A. Data Structure

Let $\gamma(i,j;d)$ be the number of packets to be sent from the jth input port in group i to output group d in each frame. The total number of packets sent from input group i to output group d is $c_{id} = \sum_{j=1}^{L} \gamma(i,j;d)$. We call the $\{c_{id}\}$ arranged in matrix form a *group traffic matrix* $\mathbf{C}$. Since only h slots per frame are available for switched videos, all the row sums and column sums of $\mathbf{C}$ cannot be larger than h. The mth slot *transmission matrix* $\mathbf{T}_m = [t_{id}^{(m)}]_{M\times M}$ describes the transmitting and receiving group pairs in slot m. Specifically, $t_{id}^{(m)} = 1$ indicates that group i videos can transmit a packet to group d outputs in slot m and $t_{id}^{(m)} = 0$ otherwise. Since each vertical or horizontal bus can transmit at most one packet in a slot, all the column sums and row sums of $\mathbf{T}_m$ cannot be larger than one.

B. Point-to-Point Slot Assignment Algorithm

The point-to-point slot assignment (PPSA) problem is to determine $\{\mathbf{T}_i\}$ such that $\mathbf{T}_1 + \mathbf{T}_2 + \cdots \mathbf{T}_h = \mathbf{C}$ for a given $\mathbf{C}$ and h. A related System of Distinct Representative (SDR) problem [12] is to find a transmission matrix with a maximum number of non-zero elements or maximum rank. To find one transmission matrix, the SDR algorithm [12] has time complexity $O(M^4)$ [13]. The PPSA problem studied here, however, does not require the transmission matrices to have maximum rank. If we use the SDR algorithm to obtain the transmission matrices one by one, the resulting time complexity is $O(hM^4)$. As we shall see below, the time complexity of our proposed PPSA algorithm is only $O(hM^2)$. Another slot assignment problem considered in [14]–[18] is to minimize the number of slots required to switch the traffic defined by $\mathbf{C}$. In other words, the problem is to find $\mathbf{T}_1, \mathbf{T}_2, \ldots$ and $\mathbf{T}_h$ such that h is minimum. Thus, the PPSA problem is a simpler version of the SDR problem and the "minimum h slot assignment" problem and the corresponding PPSA algorithm has a smaller time complexity.

The following algorithm called the PPSA algorithm computes $\mathbf{T}_1, \mathbf{T}_2, \ldots$ in turns. In the computation of $\mathbf{T}_1$, let matrix $\mathbf{B}$ be a running "matrix variable" recording the part of $\mathbf{C}$ still to be processed. For the computation of $\mathbf{T}_2, \mathbf{B}$ is updated by subtracing $\mathbf{T}_1$ from $\mathbf{C}$. For $\mathbf{T}_3, \mathbf{B}$ is set to $\mathbf{C} - \mathbf{T}_1 - \mathbf{T}_2$, etc. Let r_i and C_i be the number of non-zero elements in row i and column i of $\mathbf{B}$ respectively, and let $\mathbf{r} = [r_1 \quad r_2 \quad \cdots \quad r_M]$ and $c = [c_1 \quad c_2 \quad \ldots \quad c_M]$. The idea of the PPSA Algorithm is based on the fact that in any iteration, group i videos can be connected to one of the r_i possible output groups, and output group j can be connected to one of the c_j video groups. Row i has the smallest number of nonzero elements means group i videos has the smallest number of connection choices. Therefore, we should connect group i video channels to one of the r_i output groups specified by $\mathbf{B}$ in the present slot. Similarly, column j has the smallest number of nonzero elements means that output group j has the smallest number of connection choices. We should therefore connect output group j to one of the c_j video groups specified by $\mathbf{B}$ in the present slot. When group i video channels are connected to group j outputs in a slot, the former cannot be connected to

other output groups and the latter cannot accept packets from other video groups in the present slot. Hence, the elements in row i and column j of $\mathbf{B}$ are set to zero. The above steps are executed h times to determine a transmission matrix. The details of the PPSA algorithm are given below:

PPSA Algorithm:

$$\begin{bmatrix}\text{Inputs:} & M;\ h;\ \mathbf{C} \\ \text{Outputs:} & \mathbf{T}_1, \mathbf{T}_2, \ldots, \mathbf{T}_h\end{bmatrix}$$

1) FOR $m = 1$ TO h DO
2) $\quad \mathbf{T}_m \leftarrow \mathbf{0}$; $\mathbf{B} \leftarrow \mathbf{C}$;
3) $\quad$ FOR $n = 1$ TO M DO
4) $\quad\quad$ compute arrays $\mathbf{r}$ and $\mathbf{c}$ from $\mathbf{B}$;
5) $\quad\quad$ find the smallest nonzero entry r_p in $\mathbf{r}$ and the smallest nonzero entry c_q in $\mathbf{c}$;
6) $\quad\quad$ IF $r_p \le c_q$ THEN select a non-zero element in row p of $\mathbf{B}$ and denote its column number as i; $t_{pi}^{(m)} \leftarrow 1$; and set all elements in row p and column i of $\mathbf{B}$ to zero; ELSE select a nonzero element in column q of $\mathbf{B}$ and denote its row number as i; $t_{iq}^{(m)} \leftarrow 1$; and set all elements in row i and column q of $\mathbf{B}$ to zero;
7) $\quad$ ENDFOR;
8) $\quad \mathbf{C} \leftarrow \mathbf{C} - \mathbf{T}_m$;
9) ENDFOR.

The inner loop is executed M times, each with $O(M)$ steps. The outer loop is executed h times. Therefore, the time complexity of the PPSA Algorithm is $O(hM^2)$. Since time slots are assigned only when a new request for switched video is received, a few seconds of delay for call setup and time slot assignment computation should be acceptable to the customers. Therefore, the PPSA algorithm with time complexity $O(hM^2)$ is efficient enough. The correctness proof of the PPSA algorithm can be found in [19].

Example 1: Let $M = 3$, $h = 4$ and the group traffic matrix $\mathbf{C}$ be

$$\mathbf{C} = \begin{bmatrix}2 & 0 & 2 \\ 1 & 1 & 2 \\ 1 & 3 & 0\end{bmatrix}.$$

The details of determining $\mathbf{T}_1$ are given below (the numbers below matrix $\mathbf{B}$ are the c_i's and the numbers to the right of $\mathbf{B}$ are the r_i's).

Iteration 1: $\mathbf{B}$ is initially equal to $\mathbf{C}$:

$$\mathbf{B} = \mathbf{C} = \begin{bmatrix}2 & 0 & 2 \\ 1 & 1 & 2 \\ 1 & 3 & 0 \\ 3 & 2 & 2\end{bmatrix}\begin{matrix}2\\3\\2\end{matrix}.$$

We see that row 1, row 3, column 2 and column 3 of $\mathbf{B}$ have the smallest number of nonzero elements. We can select any one of them and suppose row 1 is chosen. Then, we select a nonzero element in row 1 to $\mathbf{B}$, say b_{13}, and set $t_{13}^{(1)}$ to 1. Therefore, $\mathbf{T}_1$ is given by:

$$\mathbf{T}_1 = \begin{bmatrix}0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0\end{bmatrix}.$$

Iteration 2: In slot 1, input group 1 cannot send additional packets and output group 3 cannot receive additional packets. Therefore, all the elements in row 1 and column 3 are set to zero:

$$\mathbf{B} = \begin{bmatrix}0 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 3 & 0 \\ 2 & 2 & 0\end{bmatrix}\begin{matrix}0\\2\\2\end{matrix}.$$

Now, rows 2, 3 and columns 1, 2 of $\mathbf{B}$ have the smallest number of nonzero elements and we can choose any one of them. Suppose row 2 is chosen. Then, we choose a nonzero element in row 2 of $\mathbf{B}$, say b_{21}, and set $t_{21}^{(1)}$ to 1. Therefore, $\mathbf{T}_1$ is given by:

$$\mathbf{T}_1 = \begin{bmatrix}0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 0\end{bmatrix}.$$

Iteration 3: Since input group 2 and output group 1 cannot handle additional packet transmissions, we set all the elements in row 2 and column 1 to zero:

$$\mathbf{B} = \begin{bmatrix}0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 1 & 0\end{bmatrix}\begin{matrix}0\\0\\1\end{matrix}.$$

Now, we select the element b_{32} and set $t_{32}^{(1)}$ to 1. Therefore, $\mathbf{T}_1$ is given by

$$\mathbf{T}_1 = \begin{bmatrix}0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0\end{bmatrix}.$$

For the computation of $\mathbf{T}_2$, we start with $\mathbf{B} = \mathbf{C} - \mathbf{T}_1$. Following the same procedure, we obtain

$$\mathbf{T}_2 = \begin{bmatrix}1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0\end{bmatrix} \quad \mathbf{T}_3 = \begin{bmatrix}1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0\end{bmatrix}$$

$$\mathbf{T}_4 = \begin{bmatrix}0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0\end{bmatrix}.$$

C. Slot Assignments

The slot assignment matrix $\mathbf{W} = [w_{ij}]_{M \times L}$ can be found by matching the destination output group number of every switched video with the nonzero elements of the transmission matrices:

$$\begin{bmatrix}\text{Inputs:} & M; h; \mathbf{T}_1, \mathbf{T}_2, \ldots, \mathbf{T}_h \\ \text{Outputs:} & [w_{ij}]_{M \times L}\end{bmatrix}$$

1) $\mathbf{W} \leftarrow \mathbf{0}$;
2) FOR $i = 1$ TO M, $m = 1$ TO h and $d = 1$ TO M DO

3) IF $t_{id}^{(m)} = 1$ and $\gamma(i,j;d) > 0$ for some $j \in \{1,2,\ldots,L\}$ THEN
4) set the m^{th} bit of w_{ij} to one;
5) $\gamma(i,j;d) \leftarrow \gamma(i,j;d) - 1$;
6) ENDIF;
7) ENDFOR.

IV. Slot Assignment for Point-to-Multipoint Transmissions

In this section, we consider the general case where a switched video packet can be multicast to a specific set of output groups.

A. Data Structure

Let group i video channels generate a total of v_i packets per frame. Label these v_i packets as packet 1, packet 2, ..., and packet v_i. The *group i traffic matrix* $\mathbf{C}_i = [c_{jd}^i]_{v_i \times M}$ is defined such that $c_{jd}^i = 1$ if packet j in group i is sent to output group d and $c_{jd}^i = 0$ otherwise. The jth row sum of $\mathbf{C}_i$ gives the number of output groups to which packet j will be destined, and the d^{th} column sum of $\mathbf{C}_i$ gives the total number of packets sent from input group i to output group d.

Since each group of switched video channels can send at most h packets in each frame, we have the following constraint:

$$0 \le v_i \le h \tag{1a}$$

Since each output group can receive at most h packets in each frame, we have the following constraint:

$$\sum_{i=1}^{M}\sum_{j=1}^{v_i} \mathbf{C}_{jd}^i \le h. \tag{1b}$$

The row sum of any transmission matrix (defined in Section III) is now larger than one for multicast packets. Let α_{ij} denote the input port number of packet j in group i and let $\mathbf{r}_i = [r_i^{(1)} \quad r_i^{(2)} \quad \cdots \quad r_i^{(m)}]$ $(i = 1, 2, \ldots, M)$ be the *slot assignment array* of group i, where $r_i^{(1)}, r_i^{(2)}, \ldots$ denote the input port number in group i that can transmit a packet in slot 1, slot 2, ..., respectively.

B. Point-to-Multipoint Slot Assignment Algorithm

The point-to-multipoint slot assignment (MPSA) problem is to determine the transmission matrices such that $\sum_{m=1}^{h} t_{id}^{(m)}$ is equal to the dth column sum of $\mathbf{C}_i$ for all $1 \le i, d \le M$. The following algorithm called MPSA algorithm can be used to find $\{\mathbf{T}_m\}$ one by one starting from $\mathbf{T}_1$. Consider the computation of $\mathbf{T}_m$ which is started as a zero matrix. Let $\mathbf{x} = [x_1 \quad x_2 \quad \cdots \quad x_M]$ be an array where $x_i = 1$ indicates that group i videos can send a packet in slot m and $x_i = 0$ otherwise, and let $\mathbf{y} = [y_1 \quad y_2 \quad \cdots \quad y_M]$ be an array where $y_i = 1$ indicates that output group i can receive a packet in slot m and $y_i = 0$ otherwise. If $x_1 \vee x_2 \vee \cdots \vee x_M = 1$ and $y_1 \vee y_2 \vee \cdots \vee y_M = 1$, then some packets can be sent from the input groups to the free output groups in slot m. Let g_i be the input group with $x_{g_i} = 1$ and that has a set of packets H_i that can complete the multicast in slot m. Let $G = \{g_i\}$.

$\mathbf{T}_m$ is found in the following steps. Compute the sets G and H_i for all $i \in G$. If G is not empty (i.e., at least one packet can be sent to all of its destinations in slot m), select $\mathbf{C}_i$ (where $i \in G$) such that, among the matrices $\mathbf{C}_p$ for all $p \in G$, $\mathbf{C}_i$ has the largest number of nonzero rows. Then select row j from $\mathbf{C}_i$ (where $j \in H_i$) such that the jth row sum of $\mathbf{C}_i$ is the largest. The reasons for this choice are 1) Group i video channels have the largest number of outstanding packets, and 2) Packet j in group i has the largest multicast multiplicity and can complete the multicast in one slot. On the other hand, if G is empty, select $\mathbf{C}_i$ such that $x_i = 1$ and $\mathbf{C}_i$ has the largest number of nonzero rows. Then select packet j from group i such that packet j can be sent to the largest number of free output groups in slot m. Let D be a set of output groups that packet j in group i can be sent to in slot m. Packet j in group i is then assigned to transmit in slot m, in other words, set $t_{id}^{(m)} = 1$ for all $d \in D$. Next, set $x_i = 0$ to indicate that group i video channels cannot send other packets in slot m and set $y_d = 0$ for all $d \in D$ to indicate that output group d for all $d \in D$ cannot receive other packets in slot m. Repeat the above steps until $x_1 \vee x_2 \vee \cdots x_M = 0$ or $y_1 \vee y_2 \vee \cdots \vee y_M = 0$ to completely determine $\mathbf{T}_m$.

When $\mathbf{T}_1, \mathbf{T}_2, \ldots, \mathbf{T}_h$ are found with some $\mathbf{C}_i \neq 0$, then the call request cannot be accommodated by the current transmission schedule. We can either block the call or use backtracking to search for other possible transmission schedules. However, our computational experience reveals that backtracking is seldom needed except when both v_i and $\sum_{i=1}^{M}\sum_{j=1}^{V_i} \mathbf{C}_{jd}^i$ are close to h [see (1)]. In other words, the algorithm can obtain transmission schedules without backtracking unless the traffic is so heavy that very few time slots are available. The MPSA algorithm is given below:

MPSA Algorithm

Inputs: $h; M; v'_i s; \mathbf{C}_i$'s and α_{ij}'s
Outputs: $\mathbf{T}_1, \mathbf{T}_2, \ldots, \mathbf{T}_h; \quad \mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_M$

1) FOR $m = 1$ TO h DO
2) $\mathbf{T}_m \leftarrow \mathbf{0}$; $r_i^{(m)} \leftarrow 0$ for $i = 1, 2, \ldots, M$;
3) IF group i videos $(i = 1, 2, \ldots, M)$ has at least one outstanding packet THEN $x_i \leftarrow 1$
ELSE $x_i \leftarrow 0$;
4) IF at least one packet is sent to output group $d (d = 1, 2, \ldots, M)$ THEN $y_d \leftarrow 1$
ELSE $y_d \leftarrow 0$;
5) WHILE $x_1 \vee x_2 \vee \cdots \vee x_M = 1$ and $y_1 \vee y_2 \vee \cdots \vee y_M = 1$ DO
6) compute G; compute H_i for all $i \in G$;
7) IF G is not empty THEN
among the matrices $\mathbf{C}_p$ for all $p \in G$, select $\mathbf{C}_i (i \in G)$ such that $\mathbf{C}_i$ has the largest number of nonzero rows; and select row j from $\mathbf{C}_i$ such that the j^{th} row sum of $\mathbf{C}_i$ is the largest
ELSE select $\mathbf{C}_i$ with $x_i = 1$ such that $\mathbf{C}_i$ has the largest number of nonzero rows; and select row j from $\mathbf{C}_i$ such that packet j of group i can be sent to the largest number of free output groups;

8) $x_i \leftarrow 0$;
9) IF $y_d \wedge c_{jd}^i = 1$ for $d = 1, 2, \ldots, M$ THEN
$t_{id}^{(m)} \leftarrow 1$; $y_d \leftarrow 0$; $c_{jd}^i \leftarrow 0$; $r_i^{(m)} \leftarrow \alpha_{ij}$;
10) ENDWHILE;
11) ENDFOR;

The WHILE loop is executed at most M times, each with $O(LM)$ steps. The FOR loop is executed h times, each with $O(LM^2)$ steps. Therefore, the time complexity of the MPSA algorithm is $O(hLM^2)$. Since time slots are assigned only when a new request for switched video is received, a few seconds of delay for call setup and time slot assignment computation should be acceptable to the customers. Therefore, the MPSA algorithm with time complexity $O(hLM^2)$ is efficient enough.

Example 2: Let $M = 4$, $h = 4$ and $\{\mathbf{C}_i\}$ be:

$$\mathbf{C}_1 = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad \mathbf{C}_2 = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{C}_3 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad \mathbf{C}_4 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The details of determining $\mathbf{T}_1$ is given below:

Iteration 1: Initially, $\mathbf{x} = \mathbf{y} = [1\ 1\ 1\ 1]$. Since all packets can complete the multicast in one slot, we can select either input group 1 or 3 because they have the largest number of outstanding packets. Then, we select a packet from input group 1 or 3 such that this packet has the largest multicast multiplicity. Suppose the first packet from input group 1 is chosen. Therefore, $\mathbf{T}_1$ is given by:

$$\mathbf{T}_1 = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Iteration 2: In slot 1, input group 1 cannot send additional packets and hence $\mathbf{x} = [0\ 1\ 1\ 1]$. Similarly, output groups 2 and 3 cannot receive additional packets and hence $\mathbf{y} = [0\ 1\ 1\ 1]$. Now, the second packet from input groups 2, 3, and 4 can complete the multicast in one slot. We choose the second packet from input group 3 because input group 3 has the largest number of outstanding packets. Therefore, $\mathbf{T}_1$ now becomes:

$$\mathbf{T}_1 = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Iteration 3: Now, input groups 1 and 3 cannot send additional packets and hence $\mathbf{x} = [0\ 1\ 0\ 1]$. Similarly, output groups 2, 3, and 4 cannot receive additional packets and hence $\mathbf{y} = [1\ 0\ 0\ 0]$. Now, the second packet from input group 2 is chosen because only this packet can complete the multicast in this slot. Therefore, $\mathbf{T}_1$ is given by

$$\mathbf{T}_1 = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

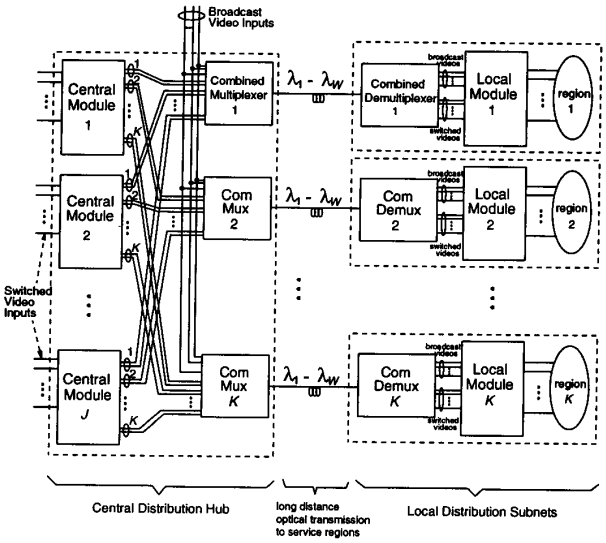


Fig. 9. A hierarchical video distribution network.

Following the same procedure, $\mathbf{T}_2, \mathbf{T}_3$ and $\mathbf{T}_4$ can be found to be:

$$\mathbf{T}_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{T}_3 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{T}_4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix}.$$

C. Slot Assignments

The slot assignments w_{ij}'s sent to the switched video adaptors can be determined as follow:

Inputs: $\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_M$
Outputs: $[w_{ij}]_{M \times L}$

1. FOR $m = 1$ TO h and $i = 1$ TO M DO
2) the mth bit of w_{ij} (where $j = r_i^{(m)}$) is set to one.

V. Network Design Example

A. Hierarchical Distribution of Video

To provide video services to a large number of customers, it is possible to use multiple copies of the above designed distribution network and connect them in a hierarchical manner. Such an arrangement is shown in Fig. 9. This hierarchical design consists of a central distribution hub and K local distribution subnets. The central distribution hub is located in a convenient place where there is access to the public broadband network and a video warehouse. It is responsible for distributing broadcast and switched video channels to the K local distribution subnets through optical carriers. Each local distribution subnet distributes the received video channels to the customers in its service region. This hierarchical design can replace a large switch (or bus matrix) by a network of smaller switches and reduce the overall circuit mileage.

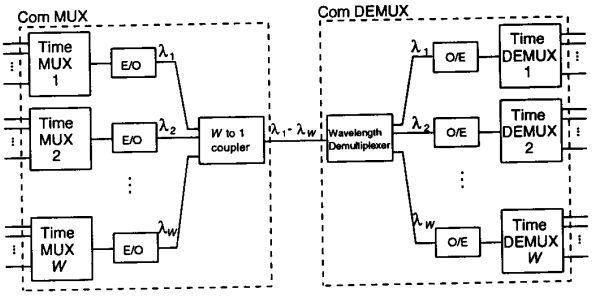


Fig. 10. The combined multiplexer and demultiplexer.

A central distribution hub consists of J *central modules* and K *combined multiplexers*. A central module is the architecture shown in Fig. 1 but without a video multiplexer as only switched video channels are connected. It can therefore devote all the slots in a frame to switched video channels. The output lines from each central module are divided into K groups (one for each region) connected to K combined multiplexers. The broadcast video inputs are fed into all K combined multiplexers. The left half of Fig. 10 shows a block diagram of a combined multiplexer. The broadcast and switched videos are further divided into W groups and each group is connected to a time multiplexer. The time multiplexed signals are then converted to optical signals at specific wavelengths. The W optical channels at wavelengths $\lambda_1, \lambda_2, \ldots, \lambda_W$ are coupled into an outgoing optical fiber for long distance transmission.

A local distribution subnet consists of a *combined demultiplexer* and a *local module*. A combined demultiplexer is shown in the right half of Fig. 10. It demultiplexes the received optical signals, converts them into electrical signals and time-demultiplexes them into individual video sources. These video sources are then connected to a local module, which is the architecture shown in Fig. 1, for further distribution to the customers in the local service region.

B. Blocking Probability

In the above design, requests for broadcast videos will never be blocked because broadcast videos are sent to all the local distribution subnets and then to all the output buses in each local module. However, a request for a switched video channel may be blocked because no time slot is available in the local distribution subnet (called local blocking) or in the central distribution hub (called central blocking). In this subsection, we derive the blocking probability for point-to-point transmission of switched video channels based on the traffic model proposed by Yum [20].

At the busiest hour of a day, let p be the probability that a customer requests a switched video session. We assume that a customer is equally likely to request a switched video from any one of the J central modules. Remind that the central modules devote all the time slots for switched videos. If the central module uses a $R_C \times R_C$ bus matrix, it can handle a maximum of $R_C(g+h)$ switched video channels and hence it can send $Y = \lfloor R_C(g+h)/K \rfloor$ switched video channels to each service region through a combined multiplexer and a local distribution subnet. Each local distribution subnet has $\lfloor N/K \rfloor$ customers where N is the total number of customers. If the local module use a $R_L \times R_L$ bus matrix, there are R_L output groups of $X = \lfloor N/KR_L \rfloor$ customers each.

Consider local blocking. Since each local module provides h time slots to each output group, blocking occurs when more than h customers in the same output group request switched videos. The local blocking probability B_L is given by

$$B_L = \sum_{n=h+1}^{X} \frac{n-h}{n} \binom{X}{n} p^n (1-p)^{X-n}. \tag{2}$$

Consider central blocking. Without loss of generality, we consider central module 1 and service region 1. Let $\tilde{Z}_i$ be the number of requests from group i customers in service region 1 to central module 1. Since each group consists of $X = \lfloor N/KR_L \rfloor$ customers and each customer requests a switched video channel from central module 1 with probability p/J, the probability $P[\tilde{Z}_i = z]$ is given by

$$P[\tilde{Z}_i = z] = \begin{cases} \binom{X}{z} \left(\frac{p}{J}\right)^z \left(1 - \frac{p}{J}\right)^{X-z} & z < h \\ \sum_{j=h}^{X} \binom{X}{j} \left(\frac{p}{J}\right)^j \left(1 - \frac{p}{J}\right)^{X-j} & z = h \end{cases} \tag{3}$$

Since there are R_L groups of customers in service region 1, the total number of requests from service region 1 to central module 1 is $\tilde{Z}_{\text{SUM}} = \tilde{Z}_1 + \tilde{Z}_2 + \cdots + \tilde{Z}_{R_L}$. Since all $\tilde{Z}_i$'s are identical and independently distributed random variables, the distribution of $\tilde{Z}_{\text{SUM}}$ is just the R_L-fold convolution of the distribution of $\tilde{Z}_1$ with itself. Since central module 1 can send at most $Y = \lfloor R_C(g+h)/K \rfloor$ switched videos to service region 1, blocking occurs when more than Y customers in service region 1 request switched videos from central module 1. In other words, central blocking occurs when $\tilde{Z}_{\text{SUM}} > Y$ and its probability B_C is given by

$$B_C = \sum_{n=Y+1}^{R_L h} \frac{n-Y}{n} P[\tilde{Z}_{\text{SUM}} = n]. \tag{4}$$

The overall blocking probability B_o is $B_L + B_C$.

C. Optimal Network Dimensioning

The complexity of a distribution network can be grossly measured by the number of switching elements used. In the above hierarchical design, this is equal to $JR_C^2 + KR_L^2$. The size of a network depends on the service requirement. Here, we stipulate that the overall blocking probability B_o must be less than a specified value B^*, say equal to 10^{-3}. The minimum complexity network configuration can be determined by minimizing $JR_C^2 + KR_L^2$ with respect to J, K and X subject to $B_o \leq B^*$. As an example of optimal design, we take $g = 10$, $h = 30$ and $p = 0.25$ and show in Fig. 11 the complexity of the optimal hierarchical networks as a function of N. Thus for $N = 5000$ and $B^* = 10^{-3}$, the complexity of the optimal hierarchical network is 508. The corresponding values of J, K, X, R_C and R_L are $J^* = 63, K^* = 16, X^* = 79, R_C^* = 2$ and $R_L^* = 4$. In other words, the optimal hierarchical network consists of 63 2 × 2 bus matrix based central modules in the central distribution hub and 16 4 × 4 bus matrix based local modules for the 16 local distribution subnets. This reveals that bus matrices with a smaller number

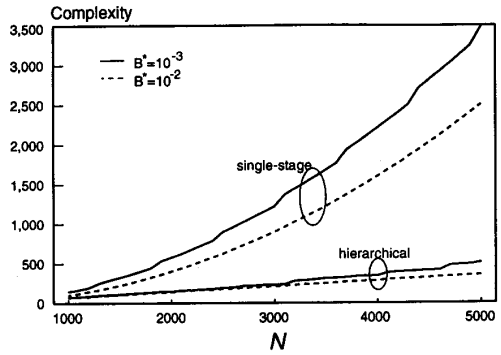


Fig. 11. Complexity versus N.

of input buses are preferred because the complexity of a bus matrix increases as the square of the number of input buses. When the blocking requirement B^* is relaxed to 10^{-2}, the complexity of the optimal hierarchical network is decreased to 356, a 30% reduction from 508 for $B^* = 10^{-3}$. The actual effect of B^* on the customers is subjective and is not the aim of this study. Also shown in Fig. 11 is the complexity of the optimal single-stage network. It is seen that the complexity of the hierarchical network is only a small fraction of that of the single-stage network. For $B^* = 10^{-3}$, this fraction decreases from 0.19 for $N = 3000$ to 0.146 for $N = 5000$.

VI. Conclusion

Video channels require relatively large bandwidth and therefore it is essential that they are switched and transmitted to the customers in a cost effective manner. In this paper, we proposed a modular architecture which uses high speed shared buses to distribute broadcast and switched videos. Two slot assignment algorithms were designed, one for point-to-point transmissions and another for multipoint transmissions. Their respective time complexities are $O(hM^2)$ and $O(hLM^2)$. A network design example was given to illustrate how a large video distribution network can be constructed and dimensioned using this architecture as building blocks. Extensions of the present work include traffic engineering for point-to-multipoint transmissions of switched videos in the hierarchical video distribution network and the resolution of bus access contention for variable bit rate video channels.

Acknowledgment

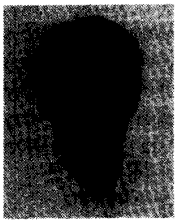
The authors would like to express their gratitude to the anonymous referees for their constructive comments and suggestions.

References

[1] C. N. Judice, E. J. Addeo, M. I. Eiger and H. L. Lemberg, "Video on demand: a wideband service or myth?" in *Proc. IEEE ICC*, June 1986, pp. 1735–1739.
[2] F. A. Tobagi, "Fast packet switch architectures for broadband integrated services digital networks," *Proc. IEEE*, vol. 78, no. 1, pp. 133–167, Jan. 1990.
[3] H. Ahmadi and W. E. Denzel, "A survey of modern high-performance switching techniques," *IEEE J. Select. Areas Commun.*, vol. 7, pp. 1091–1103, Sept. 1989.
[4] I. Cidon and I. Gopal, "PARIS: An approach to integrated high speed private networks," in *Int. J. Digit. Analog Cabled Syst.*, vol. 1, pp. 77–85, 1988.
[5] I. Cidon, I. Gopal, G. Grover, and M. Sidi, "Real time packet switching: A performance analysis," *IEEE J. Select. Areas Commun.*, vol. 6, pp. 1576–1586, Dec. 1988.
[6] H. Suzuki, H. Nagano, T. Suzuki, T. Takeuchi, and S. Iwasaki, "Output buffer switch architecture for asynchronous transfer mode," in *Proc. IEEE ICC*, pp. 99–103, 1989.
[7] S. Nojima, E. Tsutsui, H. Fukuda, and M. Hashimoto, "Integrated services packet network using bus matrix switch," *IEEE J. Select. Areas Commun.*, vol. 5, pp. 1284–1292, Oct. 1987.
[8] T. S. Yum and Y. W. Leung, "A TDM-based multibus packet switch," in *Proc. IEEE INFOCOM*, Italy, 1992.
[9] F. Tobagi, "Tutorial notes on high-speed networking," in *IEEE INFOCOM*, 1991.
[10] I. Gopal, R. Guerin, J. Janniello, and V. Theoharakis, "ATM support in a transparent network," *IEEE Networks*, pp. 62–68, Nov. 1992.
[11] C. E. Strangio, *Digital Electronics*. Englewood Cliffs, NJ: Prentice-Hall, 1984.
[12] T. Inukai, "An efficient SS/TDMA time slot assignment algorithm," *IEEE Trans. Commun.*, vol. COM-27, pp. 1449–1455, Oct. 1979.
[13] M. Chen and T. S. Yum, "A conflict-free protocol for optical WDMA networks," in *Proc. IEEE GLOBECOM*, 1991, pp. 1276–1281.
[14] S. Chalasani and A. Varma, "An improved time-slot assignment algorithm for TDM hierarchical switching systems," *IEEE Trans. Commun.*, vol. 41, pp. 312–317, Feb. 1993.
[15] ——, "Parallel algorithms for time-slot assignment in TDM switching systems," *IEEE Trans. Commun.*, vol. 41, pp. 1736–1741, Nov. 1993.
[16] M. A. Bonuccelli, "A fast time-slot assignment algorithm for TDM hierarchical switching systems," *IEEE Trans. Commun.*, vol. 37, pp. 870–874, Aug. 1989.
[17] M. A. Bonuccelli, I. Gopal, and C. K. Wong, "Incremental time-slot assignment in SS/TDMA satellite systems," *IEEE Trans. Commun.*, vol. 39, pp. 1147–1156, July 1991.
[18] M. A. Bonuccelli, "A fast time-slot assignment algorithm for TDM hierarchical switching systems," *IEEE Trans. Commun.*, vol. 37, pp. 870–874, Aug. 1989.
[19] Y. W. Leung and T. S. Yum, "A modular multirate video switch: Design and dimensioning," Dep. Inform. Eng., The Chinese University of Hong Kong, Res. Rep. '92.
[20] T. S. Yum, "Hierarchical distribution of video with dynamic port allocation," *IEEE Trans. Commun.*, vol. 39, pp. 1268–1274, Aug. 1991.

Yui-Wing Leung (S'88-M'92) received the B.Sc. and Ph.D. degrees from the Chinese University of Hong Kong in 1989 and 1992 respectively.

He is currently a lecturer in the Department of Computing, Hong Kong Polytechnic, Hong Kong. His current research interests include information networks and software systems.



Tak-Shing Yum (S'76-M'78-SM'86) received the B.S., M.S., M.Ph., and Ph.D. degrees from Columbia University in 1974, 1975, 1977, and 1978, respectively.

He worked for Bell Telephone Laboratory for two and a half years and taught at the National Chiao Tung University, Taiwan, for two years before joining the Chinese University of Hong Kong in 1982. He has published original research on packet-switched networks with contributions in routing algorithms, buffer management, deadlock detection algorithms, message resequencing analysis, and multiaccess protocols. In recent years, he branched out to work on design and analysis of cellular networks, lightwave networks, and video distribution networks. He believes that the next challenge is designing an intelligent network that can accomodate the needs of individual customers.