TRANSPORTING MPEG-II VIDEO STREAMS ON ATM NETWORKS WITH A MODIFIED J-EDD SCHEME

Ting-Chao Hou, Chien-Chang Chen and Wen-Jer Wu
Department of Electrical Engineering
National Chung Cheng University
Chia-Yi, Taiwan 621, ROC
email: tch@ee.ccu.edu.tw

ABSTRACT

Real-time video demands a stringent delay and jitter requirement. A switch in the network needs to implement an effective scheduling algorithm to meet such a requirement. The Jitter-EDD (Earliest-Due-Date) scheme designed for packet switching networks is noted for its capability to control the end-to-end delay jitter to be within the delay variation in the last switching node of the path. It, however, cannot be used in an ATM network directly due to the lack of space in the ATM cell header to carry the required correction term. We propose a method to implement the J-EDD scheme in the ATM network and evaluate its performance in transporting real-time VBR video streams. A critical system design issue is also identified.

Keywords: ATM, MPEG, QoS, Jitter-EDD, VBR

1.0   INTRODUCTION

The communication of the next century is, in all its means, the exchange of multimedia information. Multimedia information ranges from best-effort data to real-time video with widely different traffic characteristics and Quality of Service (QoS) requirements. Among the many constituents of the multimedia information, the real-time video is the most challenging one for the network to deliver. It requires not only high bandwidth but also very stringent delay and jitter guarantees. To reduce the bandwidth requirements, video information is usually compressed before being sent to the network.

The ISO MPEG (Motion Picture Expert Group) suite of coding schemes are currently used in a large variety of applications for the compression of video data. It is expected that the MPEG-coded video will become dominant multimedia traffic in the future communication network. A typical MPEG-II video stream consists of a sequence of I-frames, P-frames, and B-frames, in a periodic pattern. These three types of frames are produced with different compression techniques, thus resulting in different frame data length. A MPEG-II video stream is thus a variable-bit-rate (VBR) traffic coming out of the video encoder.

In addition to the stringent delay requirement, real-time VBR traffic also requires some timing information to be transmitted across the network so that the receiver can be in synchronization with the transmitter. In MPEG-II video, the Program Clock Reference (PCR) is inserted into the transport stream packet periodically. The PCR value must arrive at the phase lock loop (PLL) at the receiver in time so that the receiver clock can lock into the transmitter's clock to guarantee a smooth play-out of the video picture.

Obviously, the transport network for multimedia information needs to provide not only higher bandwidth but also better QoS guarantees than what the existing networks can provide. New broadband networks are being deployed worldwide to fulfill this need. The Asynchronous Transfer Mode (ATM) technology is used in these broadband networks due to its high switching capability and its capability to differentiate QoS on a per connection basis. To facilitate the QoS differentiation, services provided by ATM networks are classified into five classes or categories. They are Constant-Bit-Rate (CBR), real-time Variable-Bit-Rate (rt-VBR), non-real-time Variable-Bit-Rate (nrt-VBR), Available-Bit-Rate (ABR), and Unspecified-Bit-Rate (UBR) services [4]. The MPEG-II video belongs to the rt-VBR service category.

To provide jitter guarantee to PCR cells, the network usually relies on a specific service scheduling algorithm at the switching nodes. Service scheduling is an essential mechanism in providing differential treatment to different classes of traffic. Service disciplines schedule packets (or cells in ATM networks) to be transmitted on an outgoing link in order to ensure that the QoS requirements for various traffic classes are satisfied. In this paper, we study the feasibility of a modified Jitter-EDD scheduling algorithms on controlling the jitter for MPEG-II coded video streams over ATM networks. We also investigate the system design issues that are critical in providing jitter control for MPEG-II video streams.

In Section 2, we give an overview of MPEG. Then, a brief overview of jitter-control service disciplines is presented in Section 3. Section 4 describes the implementation of the modified Jitter-EDD scheme. Section 5 describes the VBR source model. The performance evaluation is reported in Section 6. Section 7 states our conclusions.

2.0   MPEG-II OVERVIEW

In MPEG (MPEG-1 and MPEG-II) three types of pictures are defined [10]:

Intra-frames or I-frames: These are pictures that are coded autonomously without the need of a reference to another picture.

Predictive or P-frame: These frames use similar coding algorithm as that for I-frames, but with the addition of motion compensation with respect to the previous I- or P-frame.

Bidirectionally-predicted or B-frames: The B-frames use both previous and future I- or P-frames as a reference for motion estimation and compensation.

Typically, I-frames contain more bits than P-frames. B-frames have the lowest bandwidth requirements. The frames are arranged in a deterministic periodic sequence, e.g., "IBBPBB" or "IBBPBBPBBPBBPBB", which is called a Group of Picture (GOP).

The MPEG standard defines a way of multiplexing more than one stream (video or audio) in order to produce a program. A program consists of one or more elementary streams. Elementary streams are the basic entities of a program (Fig. 1). Two streams are used in MPEG-II standard for the multiplexing process: program stream (PS) and transport stream (TS). The program stream is used in the storage media environment (e.g., CD-ROM). The transport stream is used in environments where errors are likely and is the default choice for transport over a computer network. Transport Stream and Program Stream are each logically constructed from PES packets. A PES packet consists of a header and a payload. The payload is taken sequentially from the original elementary stream (Fig. 2).

The transport stream consists of short, fixed-length packets. A transport stream packet has a length of 188 bytes. It is comprised of a 4-byte header followed by an "adaptation field" or a payload or both. The PES packets from the various elementary streams are each divided among the payload parts of a number of transport stream packets.
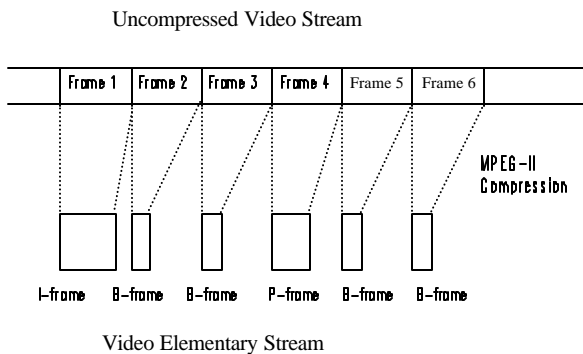
Uncompressed Video Stream

Video Elementary Stream

Fig. 1: From uncompressed data to elementary stream
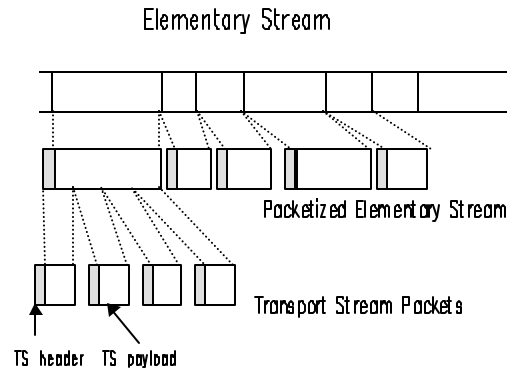

Fig. 2: Transport stream generation from PES-packets

2.1   Timing Model

The MPEG-II standard assumes a timing model in which the end-to-end delay from the signal input of the encoder to the signal output of the decoder is a constant. This delay is the sum of encoding, encoder buffering, multiplexing, communication or storage, demultiplexing, decoder buffering, decoding, and presentation delays. The MPEG system streams (include PES, TS, PS) coding contains timing information which can be used to implement systems which embody constant end-to-end delay. All timing is defined in terms of a common system clock, referred to as the System Time Clock (STC).

The STC is sampled regularly and the samples are inserted in the Transport Stream. Its value is stored in the Program Clock Reference (PCR) field found in the adaptation field of the transport stream packet. The standard defines the minimum frequency of sampling of the STC. A PCR value must appear in the Transport Stream at least every 0.1 second (10 Hz). And the PCR tolerance of delay variation is 1 *msec* [10].

2.2   MPEG-II over ATM

The ATM Forum Service Aspects and Application (SAA) Sub-group has discussed mechanisms on carrying MPEG-II transport stream over ATM networks. Compared with AAL1 (ATM Adaptation Layer 1), mapping MPEG-II transport stream into AAL5 Common Part (CPAAL5) with a null Convergence Sublayer (CS) is easier and more feasible [9, 18, 22]. These proposals consider header tradeoff, system buffer requirements, and packetization jitter. They suggest that, as a default, two Transport Stream (TS) packets (188 bytes each) are packed to form one AAL5 PDU (with an 8-byte AAL5 trailer) [17], which are then segmented into 8 ATM cells without any padding bytes (Fig. 3). The exception is when the first TS packet carries PCR, in which case this TS packet alone will form one AAL5 PDU. The resultant AAL5 PDU will need 44 padding bytes when it is segmented into 5 ATM cells. The proposed mechanism to encapsulate TS packets in AAL5-PDUs avoids introducing jitter during the generation of AAL5 PDUs (Fig. 4).
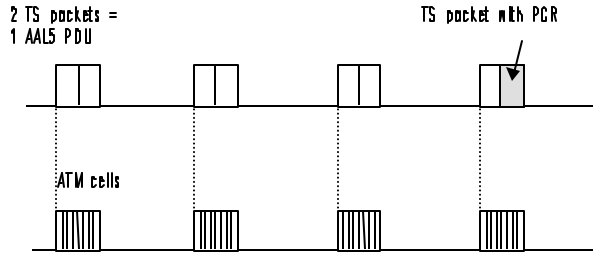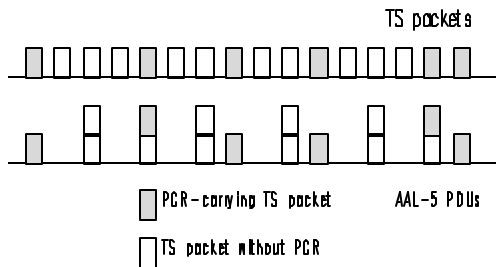
Fig. 3: Segment two TS packets into 8 ATM cells

Fig. 4: TS packetization scheme without jitter

3.0   SERVICE DISCIPLINES

A service discipline can be classified as either work-conserving or non-work-conserving. With a work-conserving service discipline, a server is never idle when there is a packet to be sent. (The "packet" here will means ATM cells when we discuss about ATM networks). With a non-work-conserving discipline, each packet is assigned, either explicitly or implicitly, an eligibility time. Even when the server is idle, if no packets are eligible, none will be transmitted.

First-Come-First-Serve (FCFS) is a traditional service discipline that was broadly used in switches in ATM networks. It is a work-conserving service discipline. Cells are served according to their order of arrival at the switch. FCFS cannot provide satisfactory delay or jitter guarantee due to the variable queue waiting time at different loading conditions. Many service disciplines have been proposed in recent years that aim to provide a better QoS. Some of them are work-conserving disciplines: earliest-due-day-first (EDF) scheduling [13], Delay-EDD (Delay-Earliest-Due-Date) [3], VC (Virtual Clock) [23,21], WFQ (Weighted Fair Queueing) [2], PGPS (Packet Generalized Processor Sharing) [1,15,16], WF2Q (Worst-case Fair Weighted Fair Queueing) [1], SCFQ (Self-Clocked Fair Queueing) [6,7]. The main focus of these schemes is to provide a fair allocation of bandwidth to all backlogged sessions (connections). Control of jitter, if there is any, is a by-product from the fair bandwidth allocation. Non-work-conserving disciplines that have been proposed include

HRR (Hierarchical Round Robin) [11], Stop-and-Go [5], and Jitter-EDD [20]. Both HRR and Stop-and-Go adopt the TDM-like framing technique and rely on peak bandwidth allocation to guarantee a jitter bound. Therefore, they are not capable of providing any statistical multiplexing gain that is available in ATM networks. The Jitter-EDD discipline may increase the average delay of the packets and decrease the average throughput of the server. But in general, it provides better jitter control than work-conserving disciplines. Such a philosophy fits best with the case that we are interested in, i.e., devising a service discipline that allows more MPEG-II connections to be admitted in the network while still meets the MPEG-II traffic's stringent jitter requirement.

3.1   Jitter-EDD

The jitter-EDD discipline provides a bound on the maximum delay difference between two packets. After a packet has been served at a server, a field in its header is stamped with *PreAhead*, which is the difference between its *deadline* and the actual finish time. A regulator at the entrance of the next server holds the packet for the *PreAhead* period before it becomes eligible to be scheduled. Fig. 5 shows the progress of a packet through two adjacent servers. In the first server, the packet gets served *PreAhead* second before its deadline. So, in the next server, it is made eligible to be sent on after *PreAhead* seconds. Since there is a constant delay between the eligibility times of the packet at two adjacent servers, the packet stream can be provided a delay jitter bound. Assuming there is no regulator at the destination host, the end-to-end delay bound is the same as the local delay bound at the last server.

Jitter-EDD was originally designed for packet switching networks. Data is packed in variable length packets for transmission. Also the field for stamping Jitter-EDD timing information is generally a small part of the whole packet. Its applicability to other network is not as straightforward as one might think. We discuss the implementation of the jitter-EDD algorithm in an ATM network in the next section. How to carry the timing information in ATM networks is the main problem we need to solve.
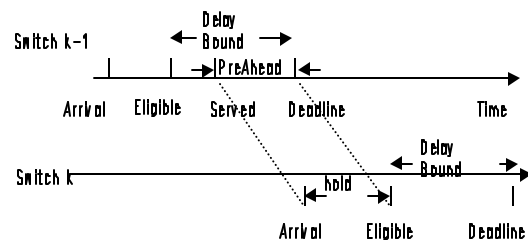

Fig. 5: Jitter-EDD control algorithm

4.0 JITTER-EDD IMPLEMENTATION

In packet switching networks, data is encapsulated in variable size packets. Additional fields can easily be included in the packet's header or the packet's trailer to carry the required control information. In ATM networks, data is encapsulated in fixed-length 53-byte ATM cells. It is impossible to increase the length of the ATM cell or extend the header field to carry Jitter-EDD timing information. Consequently, we have to use a whole ATM cell to pass Jitter-EDD timing information to downstream nodes.

In ATM networks, all switches process the cell header and do not interpret the cell payload of data cells (Payload type=0xx). Putting timing information in a normal ATM data cell is of no use for communication between ATM switches. To use cells other than the data cells to carry the timing information, one possibility is to define a new Payload Type (PT) to identify this special cell. Currently, there is only "half bit" for extending the PT definition. Only "111" is reserved for future usage. Another possibility is to reuse the OAM cell.

From [8], we know that the payload of the OAM cell will be read at switches. Two flows of OAM cells are defined at the ATM layer. F4 AM flow is used for the operation and maintenance at the virtual path layer. With Jitter-EDD, we need to sent timing information per VPI/VCI. Therefore, we have to use F5 OAM flow to carry the timing information we need. There are still un-used bits for us to define new OAM type in OAM cells. There is a 45-bytes function specific field to carry the J-EDD time stamp. In this study, we use the OAM cell to carry the J-EDD time stamp.

How many OAM cells are needed to control the jitter and delay? If we insert one OAM cell per data cell to absorb the delay jitter, though the switch can maintain the original traffic very well but the utilization of the network will be cut by half. If we examine the MPEG-II decoder's delay bound and jitter bound at the cell level, we can see that all cells of the same frame should reach the destination at the same 1/30 second frame period. However, cells other than the PCR-carrying cell can arrive anytime within the 1/30 second period. But cells which carry PCR information need to appear at almost the same place in its 1/30 second period. For the reason above, we think that one OAM cell per frame will do the jitter control well. Each MPEG-II source begins with COM (continuation of message) cells being generated until approaching the end of a frame when the last cell, named "EOM" (End of Message) cell, is generated. After the source sends the EOM cell, that means the last cell of PCR-carrying TS pair, we insert an OAM cell that carries the *PreAhead* information. If the PCR-carrying cell can reach the destination and satisfy the delay bound, the corresponding frame will do, too. It does not help much to control other cells to minimize the jitter, because they do not carry any timing information and only have delay bound to match.

4.1 Jitter-EDD compliant ATM switch

To use the Jitter-EDD scheme, we need the switch to provide the following functionality. First, it is necessary to buffer one cell to see if the next cell is an OAM cell or not. If the next cell is not an OAM cell, then the cell will go to the output queue directly; otherwise it means that the cell is the last cell of the last TS packet with PCR, and the cell will be sent to the Jitter-EDD queue (Fig. 6). The switch will then calculate the eligibility time to release the cell, at which time the cell will be put in the output FIFO queue. When the Jitter-EDD buffer is not empty, subsequent cells of the same VPI/VCI must be put into the buffer in order not to distort the cell sequence. When the PCR-carrying cell is ready to leave the output queue, the switch needs to put the *PreAhead* time in the OAM cell which will be sent out next to notify the downstream node. Similar calculation of the scheduling will be done at the downstream node again.
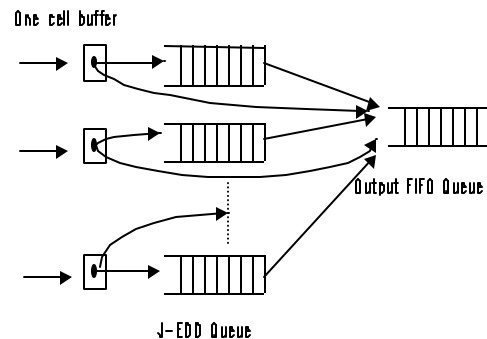

Fig. 6: Cell queueing in a Jitter-EDD switch

5.0 VBR SOURCE MODEL

MPEG-II video stream is a unique VBR traffic. The frame sequence is periodical but the bandwidth requirement for each frame type is time varying. There are many mathematical approaches for modeling MPEG-II traffic, such as histogram models [19], Markovian models [14], etc. But these models are only statistical models. They are mainly used for analytic purpose. Since we use the simulation approach to study the MPEG-II over ATM problem, we are better off using the trace files captured from real MPEG-II traffic. The trace files of MPEG-II video we used were obtained from [12]. They contain the frame sizes of a whole video sequence (*lecture*). The bit-rate statistics are shown in Table 1.

Table 1: Bit rate of lecture (Mbps)

Frame type	Average rate	Min rate	Peak rate
All	3.3	0.6	14.1
I	10.0	4.3	14.1
B	4.6	2.4	9.8
P	1.2	0.6	3.5

According to the MPEG-II specification, after PES packets are generated, the system passes them to the lower layer to generate the TS packets and adds the PCR field if there are enough stuff bytes at the last TS packet, as illustrated in Fig. 7. Every two TS packets then form an AAL5 CS-PDU (8 ATM cells).

The way that TS packets are generated and sent out has a big influence on whether the PCR jitter can be controlled to be within 1 *msec*. Suppose the source bursts out a frame at a peak rate of *Bp* for a duration *Tp*, and then stays idle for the rest of frame interval (1/30 *sec*). If there are *N* frames from different sources competing for the same outgoing link in a switch, each source will be allocated an average of *Balloc=R/N* bandwidth, where *R* is the link bandwidth. Then the last bit of one of the competing frames would be delayed by at least

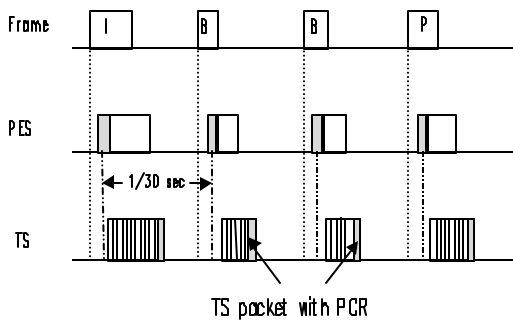$$D = \frac{B_p - B_{alloc}}{B_{alloc}} \times Tp \quad sec$$

Fig. 7: Generation of TS packets pack from video frames

The delay variation between two consecutive frames is maximized when the difference in their *Tp* is a maximum. It corresponds to the case when a small B frame is followed by an I frame. When the source bursts out the frames at the ATM link speed (150 *Mbps*), the required bandwidth *Balloc* needs to be higher than 91 *Mbps* so that the 1 *msec* jitter can be guaranteed. In other words, no more than one MPEG-II connection can be carried in a 150 *Mbps* ATM link, not a very efficient use of the link bandwidth. Based on the above analysis, it is apparent that the source needs to shape its traffic before sending it to the network.

5.1 Smoothing at the TS layer

At the MPEG-II decoder, a frame is expected to arrive within the jitter requirement of 1 *msec*. If we focus on the ATM layer, all cells of a frame must reach the destination every 1/30 seconds. If a cell is not the last one, it does not matter when within the 1/30 period the cell arrives. It always needs to wait for the last cell so that the system can reassemble the frame together for further decoding. It is therefore not necessary to send cells in a burst since the burstiness is one of the main reasons that causing cell delay variation. So, a MPEG-II source can be modified as follows. First we determine the size of the PES packet. Second, we calculate the number of TS packet pairs that are formed from the PES packet. Third, as shown in Fig. 8, we smoothly pass the TS packet pairs into the AAL-5 during the whole frame time (1/30 second). Following this procedure, the burst length of the MPEG source is only 8-ATM-cell. Smoothing traffic source makes it easier to solve the problem of cell delay variation (CDV) due to network congestion.
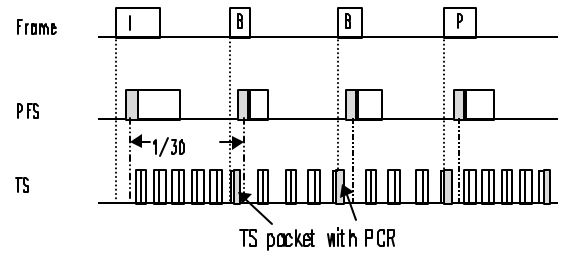

Fig. 8: Smoothed TS packets to reduce the burst length

6.0 PERFORMANCE EVALUATION

The Jitter-EDD's algorithm has a unique property in that it works mainly between two adjacent switching nodes. The traffic pattern incoming to the first switch is recreated at the middle of each subsequent switches. An end-to-end path can be considered as a concatenation of independent node pairs. By inserting delay for absorbing the jitter, the buffer requirement for each node pair is only proportional to the loading of the upstream node and not to the nodes further up. We therefore conduct our experiments based on the two-node topology.

6.1 CDV and Buffer Length with FCFS Scheduling

First, we characterize the behavior of MPEG-II traffic when the switches in the ATM network use the FCFS algorithm to serve ATM cells. Different numbers of MPEG-II connections are input to the network to provide different loading to the switch. The maximum switch buffer occupation and the cell delay variation of the last cell of the PCR-carrying TS packet are the main performance measures of this measurement. We initialize MPEG-II sources at different starting times, randomly chosen from an interval *T*. If *T* is small, I frames from different sources

21

will arrive at the switch at nearly the same time, causing short term congestion. If *T* is large, I frames from different sources will be spaced apart to avoid I frame contention. The cell delay variations under FCFS with different starting intervals *T* are plotted in Fig. 9. The legend "*T*=1000" means that the connections start in the interval of 1000 ticks (1 tick is equivalent to 5.3 x 10^{-8} *msec*). CDV data are collected from three reference connections.
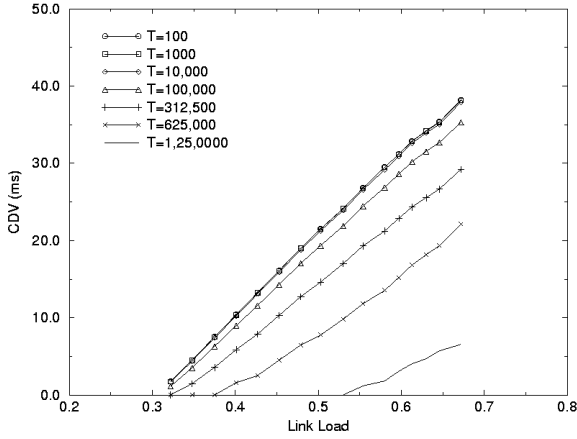

Fig. 9: CDV at different link loads and starting intervals

We observe that the CDV increases linearly as the switch loading increases beyond a threshold. This threshold is higher as the starting interval *T* is larger. With larger starting interval *T*, the network is able to carry more connections and guarantee the jitter requirement. Fig. 10 shows the maximum buffer length vs. link load at different starting intervals. The buffer occupancy exhibits a similar behavior as the CDV when the link load and the starting interval are varied.
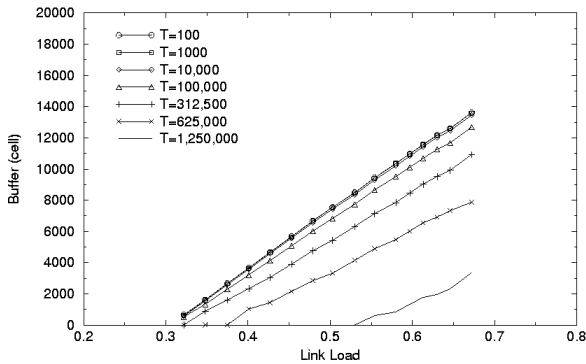

Fig. 10: Max. switch buffer length with different link loads and starting intervals

6.2   CDV and Buffer Length with Jitter-EDD

In the J-EDD algorithm, if the delay bound is chosen properly, the CDV can be controlled within the jitter bound. The CDV results in Fig. 9, which we derive from the FCFS

scheduling, are used to allocate the delay bound in the J-EDD algorithm. The delay bound is determined by

$$Delay\ Bound = CDV_{FCFS} - Jitter\ Bound. \qquad (1)$$

Fig. 11 compares the CDV of FCFS with the CDV of J-EDD. From the figure, we can see that Jitter-EDD can easily control the jitter within 1 ms to meet the MPEG-II's requirement. Since jitter-EDD is a non-work-conserving service discipline, extra buffer space will be needed to queue the cells. Therefore, Jitter-EDD requires not only extra calculation capability, but also extra buffer space in the switch. In the J-switch model in Fig. 6 of Section 4, there are two places where cells are buffered.
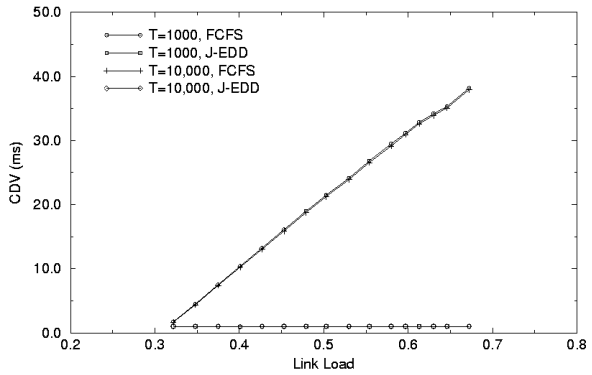

Fig. 11: CDV of J-EDD and FCFS scheduling

A.   FIFO-queue, same as a traditional FIFO switch, which is assigned per output port. The maximum FIFO buffer lengths for the jitter control experiment of smoothed sources are shown in Fig. 12.

B.   B.J-Queue, which is assigned per connection, is used to buffer the incoming cells of the next frame before current frames's PCR-carrying cell become eligible. We draw the maximum per-connection J-Queue length in Fig. 13.
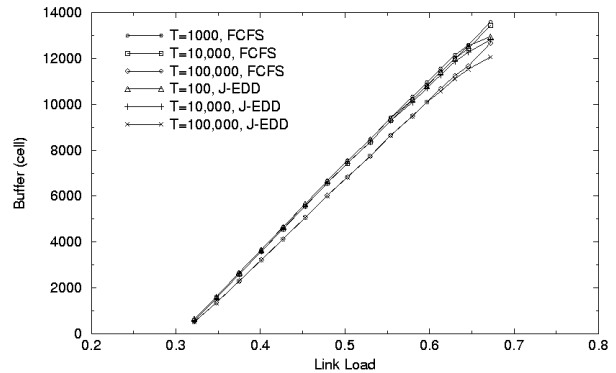

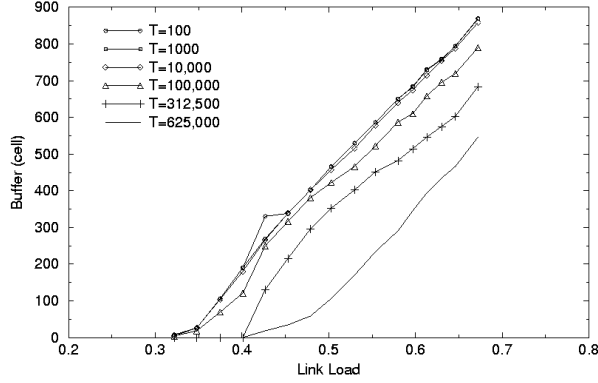Fig. 12: Max. switch FIFO buffer length of FCFS

and J-EDD

Fig. 13: Max. J-Queue buffer length of J-EDD

We first take a look at Fig. 12. The Jitter-EDD scheme does not increase the length of the output port FIFO buffer. It even makes it a little smaller. The main buffer increase is seen in the J-Queue. In Fig. 13, we see that the per-connection J-Queue length increases in proportion to the increase in the link load, which implicitly represents an increase in the holding time. The total additional buffer requirement for J-EDD is the sum of the J-Queue lengths for all connections.

6.3 Delay Bound Assignment

The accuracy of the assigned delay bound for a Jitter-EDD controlled connection is critical in controlling the CDV. It also determines how many buffers are required in the switch for this connection. How to allocate an accurate delay bound for a Jitter-EDD controlled connection in a switch? At the present time, we need to run the FCFS counterpart once and use the derived CDV_{FCFS} in Equation (1) to calculate the actual delay bound. In practice, the delay bound is assigned during the connection admission Control (CAC) phase in setting up an ATM connection. The complete CAC will be the future work of the jitter-EDD implementation. Now, we make an assumption that we have a scheme to calculate the delay bound. What will happen if the calculation does not provide a "good" delay bound? Fig. 14 shows how the CDVs change when two different delay bounds, 2.5 *msec* and 13.3 *msec*, are assigned.

First consider the case when FCFS is used. The CDV is higher than 1 *msec* when the link load increases beyond 0.32. When a 2.5 *msec* delay bound is assigned to the J-EDD controlled connection, the CDV is controlled to be within 1 ms when the load is less than 0.35. As the link load increases beyond 0.35, the delay bound is not sufficient to guarantee the 1 ms CDV. This corresponds to the situation when insufficient delay bound is assigned. We can see that the CDV is reduced by roughly 2.5 *msec* in comparison with the FCFS case. Similarly, when a 13.3 *msec* delay bound is assigned to the J-EDD controlled connection, the CDV is controlled to be within 1 ms when

the load is less than 0.45. As the link load increases beyond 0.45, the delay bound is not sufficient to guarantee the 1 ms CDV. We can also see that the CDV is reduced by roughly 13.3 *msec* in comparison with the FCFS case. The maximum J-Queue length of the two conditions are plotted in Fig. 15.
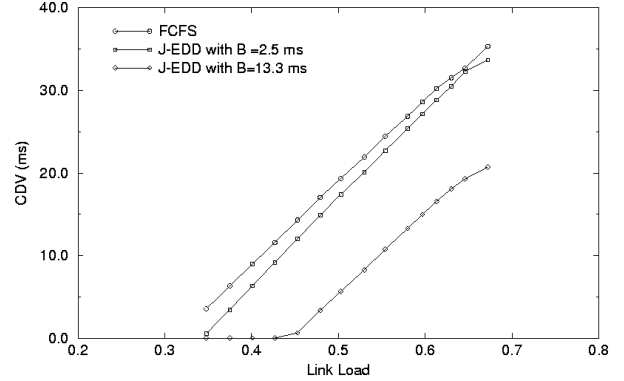

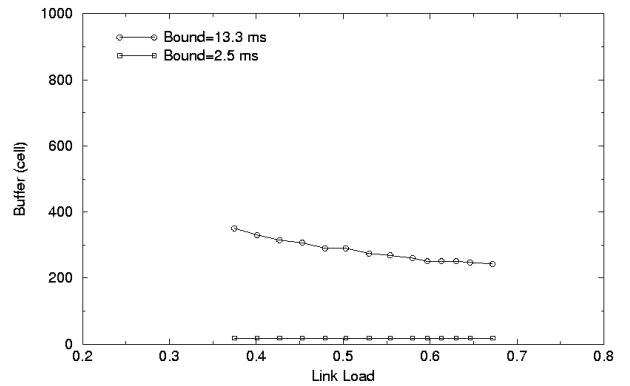Fig. 14: CDV when insufficient bound is assigned

Fig. 15: Maximum J-Queue length vs. link load with a fixed delay bound

When over allocating the delay bound, we found that jitter-EDD can make all PCR-carrying cells' CDV down to 0. If jitter-EDD can control CDV so well, why not allocate a huge delay bound to absorb the CDV for all loading conditions? The problem is that we then need a large buffer in the J-EDD Queue and the delay requirement for real-time video may mot be met.

7.0 CONCLUSION

Transporting multimedia traffic over a broadband network poses a great challenge to the system designer as to how to guarantee the stringent jitter requirement of real-time video traffic. For any jitter control algorithm to be effective, the video stream needs to be shaped at the Transport Stream packet level before it is passed to the ATM AAL layer. We have implemented a modified Jitter-EDD algorithm that is suitable for use in ATM networks. The basic principle of the Jitter-EDD algorithm is that it increases cells' end-to-

end delay to reduce the cell delay variation. The algorithm can provide jitter guarantee to MPEG-II PCR by controlling the jitter of the PCR-carrying cells. Increasing cells' delay results in larger buffer requirement in the switch. The extra buffer requirement is proportional to the holding time if the delay bound is assigned correctly. If insufficient delay bound is assigned to a connection, the CDV will not be controlled.

REFERENCES

[1]   J. C. R. Bennett and H. Zhang., "WF2Q: Worst-case fair weighted fair queueing". In *Proceedings IEEE INFOCOM,* 1996.

[2]   A. Demers, S. Keshav, and S. Shenkar. "Analysis and simulation of a fair queueing algorithm". *Internet Res. and Exper.*, 1, 1990.

[3]   D. Ferrari and D. Verma. "A scheme for real-time channel establishment in wide-area network". *IEEE Journal on Selected Areas in Communication*, Vol. 8, April 1990, pp. 368-379.

[4]   ATM Forum 95-0013R10. *Traffic management specification version 4.0*, ATM-Forum, 1996.

[5]   S. J. Golestani, "Congestion-free communication in high-speed packet network". *IEEE Transactions on communication*, Dec. 1991, pp. 1802-1812.

[6]   S. J. Golestani. "A self-clocked fair queueing scheme for broadband application". In *Proc. IEEE INFOCOM,* 1994, pp. 636-646.

[7]   S. J. Golestani. "Network delay analysis of a class of fair queueing algorithms". *IEEE J. Select. Areas Commun.*, August 1995, pp. 1057-1070.

[8]   R. Handel, M. N. Hunber, and S. Schroder. *ATM Networks Concepts, Protocols, Applications.* Addison-Wesley, 1994.

[9]   P. Hodgins and E. Itakura. *AAL5 encapsulation MPEG-2*. Technical report, ATM-Forum/94-1129, Dec. 1994.

[10] ISO/IEC JTC1/SC29/WG11. *Generic coding of moving pictures and associated audio: Systems.* Technical report, International Standard ISO/IEC 13818-1, 1994.

[11] C. R. Kalmanek, H. Kanakia, and S. Keshav. "Rate controlled servers for very-high speed networks". In *Proceedings IEEE GLOBECOM'90*, 1990, pp. 300.3.1-300.3.9.

[12] E. W. Knightly and H. Zhang. "Traffic characterization and switch utilization using a deterministic bounding interval dependent traffic model". In *Proceedings of IEEE INFOCOM'95*, 1995.

[13] C. Liu and J. Iayland. "Scheduling algorithms for multiprogramming in a hard-real-time environment". *J. ACM*, Vol. 20, Jan. 1973, pp. 46-61.

[14] D. M. Luncantoni, M.F. Neuts, and A.R. Reibman. "Methods for performance evaluation of VBR video traffic models". *IEEE/ACM Transactions on Networking*, Vol. 2, No. 2, Apr. 1994, pp. 176-180.

[15] A. Parekh and R. Gallager. "A generalized processor sharing approach to flow control in integrated services networks: The single node case". *IEEE/ACM Trans. Networking*, Vol. 1, No. 3, June 1993, pp. 344-357.

[16] A. Parekh and R. Gallager. "A generalized processor sharing approach to flow control in integrated services networks: The multiple node case". *IEEE/ACM Trans. Networking*, Vol. 2, No. 2, 1994, pp. 137-150.

[17] D. Singer and M. Lin. *MPEG-2/AAL5 proposed pdu size*. ATM Forum/94-1047, November 1994.

[18] P. Skelly. *Proposal for MPEG-2 transport over ATM.*, ATM Forum 94-0842, September 1994.

[19] P. Skelly, M. Schwarz, and S. Dixit. "A histogram-based model for video traffic behavior in an ATM multiplexer". *IEEE/ACM Transactions on Networking*, Vol. 1, No. 4, Aug. 1993, pp. 446-459.

[20] D. C. Verma, H. Zhang, and D. Ferrari. "Delay jitter control for real-time communication in a packet switching network". In *IEEE Conference on Communication Software*, 1991, pp. 35-43.

[21] A. Weinrib and L. T. Wu. "Virtual clock and leaky buckets: flow control protocols for high-speed networks". In *Protocols for High-Speed Network, II IFIP*, 1991.

[22] J. Zhang, M. Perkins, et al. "*Carrying MPEG-2 transport stream using AAL-5*". ATM Forum 94-0879, Sep. 1994.

[23] L. Zhang. "Virtual clock: A new traffic control algorithm for packet-switched networks". *ACM Trans. Comp. Systems*, Vol. 9, No. 2, May 1991, pp. 101-124.

BIOGRAPHY

Ting-Chao Hou was born in Taiwan, R. O. C., in 1957. He received the B.S. degree from the National Taiwan University, Taipei, in 1979, and the M.S. and Ph.D. degrees from the University of Southern California, Los Angeles, in 1982 and 1985, respectively, all in electrical engineering. He was in military service in 1979-1981. In October 1985 he joined the Teletraffic Theory and System Performance Department at the AT&T Bell Laboratories, where he was a Member of Technical Staff. From 1993 to 1995, he was on leave with the National Chung Cheng University, Chia-yi, Taiwan, as a visiting associate professor. In 1995, he joined the faculty of the National Chung Cheng University. His research interests are in the areas of packet radio networks, ATM/STM switching, congestion controls, Advanced Intelligent Network, and the UNIX operating system.

Chien-Chang Chen received his M.S. degree from the department of Electrical Engineering, National Chung Cheng University, in 1997. He is currently in military service.

Wen-Jer Wu received his B.S. degree from the department of Electrical Engineering, National Taiwan University, in 1996. He is currently a graduate student in the department of Electrical Engineering, National Chung Cheng University.