Transmission of MPEG-2 Streams over Non-Guaranteed Quality of Service Networks

A. Basso, G. L. Cash, M. R. Civanlar

AT&T Labs - Research

ABSTRACT*: The explosive growth of the Internet and the intranets attracted a great deal of attention to the implementations and performances of networked multimedia services, which involve the transport of real-time multimedia streams over non-guaranteed quality of service (QoS) networks. In this paper, we discuss some issues related to the transport of MPEG-2 streams over such networks by means of the most recent transport protocols compliant with the Real-Time Transport Protocol (RTP) [1] defined by the Internet Engineering Task Force (IETF). MPEG-2 encoded audio and video transmission is important for several applications including high quality video-on-demand as a part of information-on-demand and high quality video conferencing using the existing network infrastructures.*

1. INTRODUCTION

Multimedia transport protocols play an important role in the effective delivery and networking of multimedia data. In recent years, several application and transport protocols have been developed. In this context, the MPEG committee has developed the program and the transport streams for the delivery of several MPEG-2 encoded video and audio programs together with general purpose data. The program stream (PS) is intended for mostly error free environments. The transport stream (TS), on the other hand, addresses the needs of relatively error prone systems such as point-to-point, dedicated transmission systems for television program distribution, cable systems for set-top boxes, etc. [2].

The ubiquity of the Internet and the continuous increase in the computing power of the desktop computers together with the availability of relatively inexpensive MPEG-2 codec hardware in the form of computer plug-in boards have made MPEG-2 based, high quality video communications an interesting possibility for the desktop computers. The biggest challenge in implementing such systems is to be able to sustain continuous playback under high packet losses without losing the multicasting functionality provided by these networks.

The RTP is designed to deliver various kinds of real time data over packet networks that include non-guaranteed quality-of-service (QoS) networks. RTP addresses the needs of real-time data transmission only and relies on other well established network protocols for other communications services such as routing, multiplexing and timing. This way, we don't need to re-define these services, which are proven to be satisfactory in general, for each payload type. Additionally, RTP is based on the Application Level Framing (ALF) and Integrated Layer Processing (ILP) [3][4] principles, which dictate using the properties of the payload in designing a data transmission system as much as possible. For example, if we know that the payload is MPEG video, we should design our packetization scheme based on "slices" because they are the smallest independently decodable data units for MPEG video. This approach provides a much more suitable framework for MPEG-2 transmission over networks with high packet loss rates.

The next section describes the services provided by RTP for MPEG transmission. Section 3 discusses specific MPEG-2 payload types and their advantages and disadvantages. Section 4 describes a system model that we used in our implementations. Section 5 discusses the results from our early experiments.

2. RTP

The services provided by the RTP include payload type identification, sequence numbering, time stamping and delivery monitoring. RTP typically runs on top of User Datagram Protocol (UDP) [5] to make use of its multiplexing and checksum services. This is in addition to the basic networking services provided by the underlying IP layer. However, RTP may be used with other suitable underlying network or transport protocols. Also, RTP supports data transfer to multiple destinations using multicast distribution if this functionality is supported by the underlying network as in TCP/IP networks.

The payload type identification service of the RTP together with the multiplexing services supported by the underlying transport protocol, such as UDP, provides the necessary infrastructure to multiplex a large variety of information effectively. Multicast transmission of several MPEG streams multiplexed together with any other auxiliary information can easily be handled using these services. RTP allows additional information to be added to its generic headers for each payload type. The specific information related to MPEG data, carried in these payload specific header extensions, will be discussed in the next section.

The checksum service that may be provided by the underlying transport or network protocol protects the higher layers from receiving corrupt packets. In most implementations, the UDP layer does not pass corrupt packets to the higher layers. So, even a single bit error may result in a lost packet. On the other hand, without this service, it is the application's (in this case, MPEG decoder's) responsibility to deal with bit errors. The packet sequence numbering function of the RTP makes packet loss detection possible because the lower layers need not provide this information.

The time stamping service provides for encoder/ decoder clock matching as well as synchronization of several sources and for measuring packet arrival jitter. Each RTP packet carries a 32 bit timestamp which reflects the sampling instant of the first byte in the data packet. For MPEG payloads, the sampling instant is derived from a 90 KHz clock.
RTP packets that belong to the same video frame have equal timestamps. The interpretation and use of the timestamp is payload dependent as described in the next section. The initial value of the timestamp is random to make known-plaintext attacks on encryption difficult.

2.1 RTP Control Protocol - RTCP

RTCP is an associated protocol with RTP and it is designed to handle the delivery monitoring service of the RTP. It is based on periodic transmission of control packets from all participants of a session to all other participants using the same distribution mechanism as the RTP data packets. It has four main functions:

1. provide feedback on the quality of the data distribution including timing information
2. establish an identification, i.e. a canonical name (CNAME) for each participant
3. scale the control packet transmission with the number of participants
4. provide minimal session control information.

A complete discussion of the RTCP is out of the scope of this paper; however, an understanding of its timing services is needed for explaining MPEG-2 transmission over RTP. In an RTP session, each sender transmits periodic RTCP packets that contain 64 bit Network Time Protocol (NTP) [6] timestamps, indicating the wallclock (absolute) time when the packet was sent. This information can be used in combination with timestamps returned in reception reports from other receivers to measure round-trip propagation to those receivers. Additionally, the same packet contains RTP timestamps that corresponds to the same time as the NTP timestamp (above), but in the same units and with the same random offset as the RTP timestamps in the RTP data packets. This correspondence may be used for intra- and inter-media synchronization for sources whose NTP timestamps are synchronized.

In the next section, we will focus on the functions of the payload specific headers and different packetization approaches for the transport of MPEG-2 data over RTP.

3. RTP PAYLOADS FOR MPEG

Currently, three payload types are defined for the transmission of MPEG - 2 streams over RTP:

3.1 Payload Based on Encapsulated MPEG-2 PS and TS [7]*:*

This payload type is established to accommodate for the hardware MPEG-2 codec implementations that operate directly on PS or TS. Systems using large existing data bases that are based on MPEG -2 PS or TS material may benefit from this payload type also.

This packetization approach uses MPEG - 2 timing model which is based on MPEG-2 Program Clock Reference (PCR), Decoding Time stamps (DTS) and Presentation Time Stamps (PTS) [2]. The RTP timestamps are not used in decoding; however, they can be functional for estimating and reducing network-induced jitter and time drift between the transmitter and the receiver. Considering the sensitivity of the MPEG - 2 timing system to network jitter, this may prove to be a significant functionality.

In this approach, the TS's are packetized so that each packet contains an integral multiple of the MPEG-2 transport packets, which are 188 bytes, to increase the transmission efficiency. The PS's are packetized as stream of bytes without any structure.

3.2 Payload Based on Separate Packetization of MPEG- 2 Audio and Video Elementary Streams [7]:

This payload format is radically different from the previous case and takes advantage of the MPEG-2 video and audio stream structure to reduce dependencies among packets and to maximize the amount of decodable data at the receiver end. For example, the MPEG "video sequence header", when present, is required to be at the beginning of an RTP payload. Also, GOP and picture headers must be put at the beginning of a packet or follow a prior header. Each header must be completely contained within the packet. The beginning of a slice must either be the first data in a packet after any MPEG headers or must follow after some integral number of slices in a packet. This requirement insures that the beginning of the next slice after one with a missing packet can be found without requiring that the receiver scan the packet contents.

For this payload type, it is recommended that the "sequence" and the "GOP" headers are repeated periodically. If the original stream does not have this property, these headers can be saved and inserted into the transmitted stream at the required frequency. This is needed to increase the error resilience, as well as to allow channel changing and starting to decode an MPEG bit-stream at almost arbitrary points in the stream.

The information in the "picture header" can be replicated in each RTP packet to increase the decodability after packet losses. Also, there is a mechanism to signal the usability of the previously transmitted header information to reconstruct lost headers.

This packetization structure allows detection of header losses and makes almost all received packets decodable.

This payload type uses the RTP timing mechanisms. Transmitter/receiver synchronization is provided through NTP. The RTP timestamps serve a purpose similar to the PTS of MPEG. All packets that make up a picture or audio frame have the same timestamp. The timestamps may not be monotonically increasing in a video stream if B pictures are present. For packets that contain only headers, the timestamp is that of the subsequent picture. There is no equivalent to MPEG's DTS and no explicit decoder buffer control mechanisms are defined.

3.3 Payload Based on Joint Packetization of MPEG-2 Audio and Video Elementary Streams (bundled MPEG format) according to [8]:

This bundled packetization scheme is needed because it has several advantages over other schemes for some important applications including video-on-demand (VOD) where, audio and video are always used together. Its advantages over independent packetization of audio and video (Section 3.2) are:

1. Uses one transmission and one reception port per each bundled A/V. This may increase the number of streams that can be served e.g., from a VOD server. Also, it may improve the client side performance by removing the system overhead caused by handling two ports.

2. Provides implicit synchronization of audio and video. This is particularly convenient when the A/V data is stored in an interleaved format at the server and no stream other than the bundled A/V is to be transmitted during the session.

3. Reduces the header overhead. Since using large packets increases the effects of losses and delay, audio only packets need to be smaller increasing the overhead. An A/V bundled format can provide about 1% overall overhead reduction.

4. May reduce overall receiver buffer size. Audio and video streams may experience different delays when transmitted separately. For lip-synch, the receiver buffers need to be designed for the longest of these delays. Reduced buffer sizes are important for low delay applications.

5. Provides better control of the total bandwidth.

This payload type is to be used when these advantages are important enough to sacrifice the modularity of having separate audio and video streams.

The advantages of this approach over packetization of the MPEG - 2 TS (Section 3.1) are:

1. Reduced overhead. It does not contain systems layer information which is redundant for the RTP (essentially they address similar issues). This overhead is 2-3% of the overall bandwidth.

2. Easier error recovery. As discussed before, the TS is not designed for handling large losses. Because of the structured packetization of this payload type which is consistent with the ALF principle, loss concealment and error recovery can be made simpler and more effective.

3.4 Layered Payloads:

The "data partitioning" method of MPEG-2 [2],[9] defines the syntax and semantics for partitioning an

MPEG-2 encoded video bitstream into "high priority" and "low priority" parts. If the "high priority" (HP) part is selected to contain only the header information, it is less than two percent of the video data and can be transmitted before the start of the real-time transmission using a reliable protocol. In order to synchronize the HP data with the corresponding real-time stream, the initial value of the timestamp for the real-time stream may be inserted at the beginning of the HP data.

Alternatively, the HP data may be transmitted along with the A/V data using layered multimedia transmission techniques for RTP [10].

This approach can be used with all of the payloads defined above. However, in practice, using it with the TS encapsulation system requires significant processing at both the transmitter (to extract the HP stream) and the receiver (to construct the transport stream from the HP data) ends. Instead of performing these operations on the TS, it can be converted to the elementary streams which can be packetized for improved performance using one of the payloads described above.

4. SYSTEM MODEL

The encoder/decoder model used in our test-bed is shown in Fig.1.

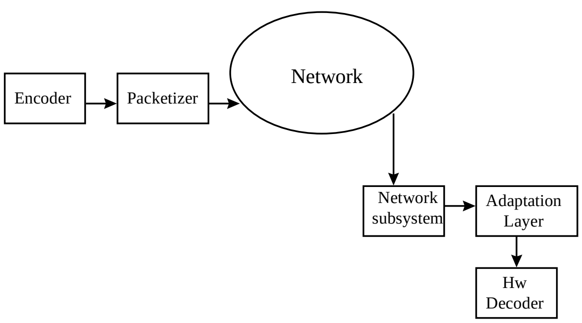

Fig. 1

We used two different encoders in our experiments. The first one, that is used to test the packetization approach described in Section 3.2, generates MPEG-2 TS data which gets delivered to a workstation over a proprietary interface. The workstation performs the packetization and delivers the RTP packets to the PC running the “network subsystem and adaptation layer” application over a network connection.

The second encoder, that is used with the other packetization techniques, is a PC plug-in board that delivers MPEG-2 elementary streams to the system over the PCI bus. The packetization software runs on the same PC and is responsible for generating and handling the necessary information for constructing the packets, packetizing the data and transmitting them to the network. We used regular ethernet as our network connection.

The receiver’s “network subsystem and adaptation layer” is responsible for converting the RTP packets received from the network into MPEG-2 elementary streams and delivering them to the decoder, which is implemented as a PC plug-in board. The entire error recovery is implemented in the network adaptation unit of the receiver.

5. DISCUSSION

5.1 Implementation issues

Multi-thread vs Single-thread

Multithreading is a powerful and effective programming technique than can sensibly improve the performance and the responsiveness of an application.

On the other side, because a preemptive multitasking system can interrupt a thread at any point to switch control to another thread, any undesirable interaction between two threads may show up on a random basis. Thus designing, coding and debugging a multithreaded application implies several complexities such as cross-synchronization among different threads, debugging, and timing.

The three main tasks performed in our encoder are

1) Capturing of the coded audio and video bitstreams from an MPEG-2 encoder board
2) Packetize the captured streams in RTP packets
3) Schedule the delivery of the RTP packets.
4) Perform a timed delivery of the packets

In a single-threaded model these four tasks are executed sequentially by means of non-blocking functions for maximum flexibility. The tasks are connected by properly designed buffers.

A nice property of the single-threaded model is the fact that any change in the MPEG-2 audio video rate will automatically be reflected on the packet delivery rate.

In the multi-threaded model the tasks are performed in separate threads using blocking functions. The threads are connected by means of buffers. For each thread the timing has to be computed very carefully and some threads (i.e. capture and timed delivery) have to be made “safe” by defining critical sections which disable possible interruption during the thread execution.

Figure 2 shows the jitter in the timed delivery of the RTP packets relative to 1200 frames at a bitrate of 3Mb/s that have been measured in the multithread model. For a timed delivery that required an interval between delivered packets of 4 ms we measured a jitter that had up to 12 ms peaks i.e. three times larger than the interval itself.

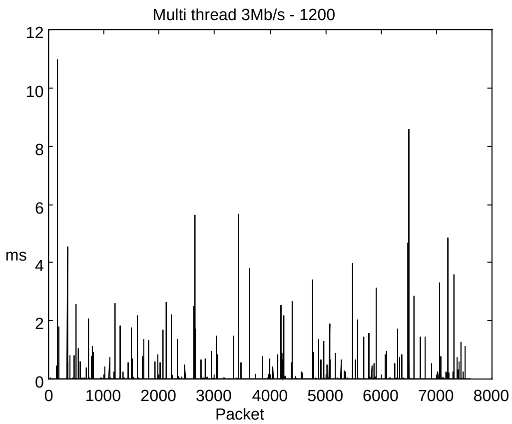

Fig. 2

The jitter problem has been finally solved at the expense of a complex fine tuning in the timing of the threads.

Our preliminary experiments show that the overall performance of the single-threaded encoder is superior to the multi-threaded one and that the multithreaded approach is less flexible than the single-thread one.

5.2 Protocol Performance

Preliminary experiments have been carried out to evaluate the performances of the different MPEG-2 packetization schemes.
The server and the client have been tested in different network configurations .
Tests have been performed at 4Mb/s for the MPEG-2 video and 192 Kb/s MPEG-1 audio. The slice size chosen was 22 MacroBlocks/slice.
Regarding MPEG-2 Transport Stream over RTP experiments the chosen fixed RTP payload size was of 1316 bytes equivalent to 7 transport packets of 188 bytes each.
The MPEG-2 elementary streams over RTP experiments used variable payload size .
Experiments have been carried out under typical intranet network load which includes background broadcast traffic, interactive session and ftp traffic and an additional low bitrate video session (~800Kb/s). In the experiments the single-thread server model has been used.

The simple evaluation criterion employed in the experiments consisted in counting the video and audio glitches which occurred at the receiver end for the different packetization schemes. The same average network conditions and video/audio material was used during the tests.

Our preliminary experiments show that the packetizations presented in sections 3.2 and 3.3 are more robust than the one presented in section 3.1 as expected.
This result is partly explained by the fact that in the packetizations presented in sections 3.2 and 3.3 the MPEG-2 video/audio stream is packetized only once while in the case of Transport Stream over RTP (section 3.1) it is packetized three times: it is firstly packetized in Packetized Elementary Stream packets (PES) then into Transport Stream packets (TS) and finally in RTP packets. These packets can have different lengths. Thus the impact of a packet loss or a bit error will be on average more significant in this packetization scheme that in the other two.

5.1 Future directions

We plan to perform more precise evaluations of the packetization schemes discussed here under several simulated network loads and to evaluate the overall quality as perceived at the receiver end by an user. Furthermore we plan to analyze in detail the performances of the different synchronization schemes used by the packetization methods discussed.

6. REFERENCES

[1] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", RFC 1889, January 1996.

[2] ISO/IEC International Standard 13818, "Generic coding of moving pictures and associated audio information," November 1994.

[3] Clark, D., and Tennenhouse, D. " Architecture Considerations for a New Generation of Protocols," Proc. of ACM SIGCOM '90, Sept. 90,pp.201-208.

[4] H. Schulzrinne, "RTP Profile for Audio and Video Conferences with Minimal Control," RFC 1890, January 1996.

[5] D. E. Comer, "Internetworking with TCP/IP," ISBN 0-13-468505-9, Prentice-Hall, NJ, 1991.

[6] Mills, D., "Network Time Protocol Version 3," RFC 1305, UDEL, March 1992.

[7] D. Hoffman, G. Fernando, V. Goyal and M. R. Civanlar, "RTP Payload Format for MPEG1 / MPEG2 Video," draft-ietf-avt-mpeg-new-01, Internet draft, June 1997.

[8] M. Reha Civanlar, Glenn L. Cash, Barry G. Haskell, "RTP Payload Format for Bundled MPEG," draft-civanlar-bmpeg-01, Internet draft, February 1997.

[9] R. Aravind, M. R. Civanlar, Amy R. Reibman, "Packet Loss Resilience of MPEG-2 Scalable Video Coding Algorithms," IEEE Transactions on Circuits and Systems for Video Technology, October 1996.

[10] M. F. Speer, S. McCanne, "RTP Usage with Layered Multimedia Streams," Internet Draft, draft-speer-avt-layered-video-02, December 1996.