ADAPTIVE, BEST-EFFORT DELIVERY OF LIVE AUDIO AND VIDEO ACROSS PACKET-SWITCHED NETWORKS

Video Abstract

Kevin Jeffay *Donald L. Stone*
University of North Carolina at Chapel Hill
Department of Computer Science
Chapel Hill, NC 27599-3175
{jeffay,stone}@cs.unc.edu

INTRODUCTION

This videotape is a demonstration of a transport protocol developed by the authors for the transmission of live audio and video streams. The goal of this work has been to understand the complexity of supporting applications such as desktop video conferencing when the network does not support real-time communication. We believe this problem is important because such networks will likely exist for the foreseeable future, hence the problems addressed by this work are fundamental in delivering continuous media in real-time across the "last mile" to the desktop.

Our protocol is a "best effort" protocol that attempts to ameliorate the effect of three basic phenomena: jitter, congestion, and packet loss, to provide low latency, synchronized audio and video communications [3]. This goal is realized through four transport and display mechanisms, and a real-time implementation of these mechanisms that integrates operating system services (*e.g.*, scheduling and resource allocation, and device management) with network communication services (*e.g.*, transport protocols), and with application code (*e.g.*, display routines). The four mechanisms are: a facility for varying synchronization between audio and video to achieve continuous audio in the face of jitter, a network congestion monitoring mechanism that is used to control media latency, a queueing mechanism at the sender that is used to maximize throughput with out unnecessarily increasing latency, and a forward error correction mechanism for transmitting audio frames multiple times to ameliorate the effects of packet loss in the network.

A key difficulty in evaluating our work has been the lack of metrics for comparing two given media transmission and playout scenarios. For example, performance measures such as end-to-end latency, frame transmission and playout rates, gap-rates, intermedia synchronization differential, *etc.*, are relatively easy to compute, but difficult to relate. For example, if scheme *A* results in lower end-to-end latency than scheme *B*, but *B* provides a lower gap-rate than *A*, which has performed better?

We do not provide any answers to this dilemma. Instead, we simply demonstrate, through the use of our protocol, the qualitative effects of varying and trading off performance parameters such as lip synchronization and gap-rate.

This videotape attempts to (1) demonstrate the quality of the audio/video streams delivered via our protocol on congested networks, and (2) give viewers a qualitative feel for the effects of varying various so-called quality-of-service parameters such as number of discontinuities (*e.g.*, gap rate), end-to-end latency, lip sync, and throughput.

DESCRIPTION OF VIDEOTAPE

Three demonstrations of transmitting digital audio and video across interconnected local-area networks are presented. The first illustrates the latency inherent in our video conferencing system. End-to-end latency is one of the most important performance parameters for a videoconferencing system as latency can severely impair and impede interaction between conference participants [2, 8]. At present there is some agreement that an end-to-end latency of no more than 250 ms. is acceptable [1]. In the best case, our system is capable of delivering synchronized audio and video streams with an end-to-end latency of approximately 170 ms. In the first demonstration we illustrate the effect of this latency by comparing our system with an analog conferencing system (with no latency). We show a split screen with analog video in one half and digital video in the other half (Figure 1). The digital video is shown after having been acquired by a workstation, compressed, transmitted over an idle network, received by a second workstation, decompressed, and dis-

Figure 1: Demonstration of latency differential between analog (upper left) and digital (lower right) systems.

played. It takes approximately 170 ms. for a video frame to propagate from the camera to the display [3].

The second demonstration illustrates the effect of varying the synchronization between the audio and video streams. As described in [3], a useful technique for ameliorating the effect of network congestion is to purposely play audio and video out of exact synchronization; specifically, to play audio frames ahead (in time) of their corresponding video frames. Although this technique has proved effective in improving quantitative measures of video conference performance, playing audio "ahead" of video is unnatural. In nature the speed of sound is several orders of magnitude slower than the speed of light and hence whenever we view noise-emitting scenes from a distance, we perceive the visual information before the corresponding sonic information. Humans are therefore more tolerant of audio "behind" video. Our system assumes (somewhat arbitrarily although motivated by [7]) that users will tolerate a synchronization differential of at least 100 ms.

The second demonstrations varies the degree to which audio is played ahead of video while a person is speaking and while a person claps (Figure 2). This illustrates that while humans are, in general, relatively intolerant of audio ahead of video, our ability to perceive this to be the case depends on such (arbitrary) factors as the resolution of the image and composition of the scene. For example, it is much easier to discern the difference in synchronization in the clapping experiment than in the speaking experiment.

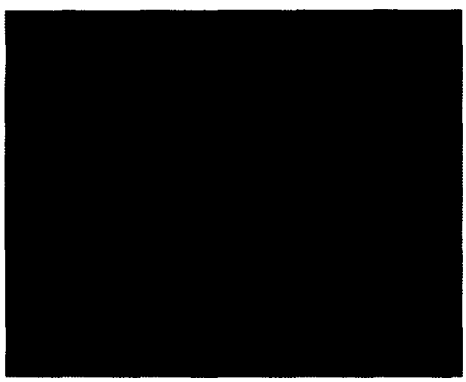

Figure 2: Demonstration of the effect of varying audio/video synchronization by clapping.

Finally, we demonstrate the effect of transmitting audio and video via our protocol and compare the protocol's performance to UDP. We present a set of controlled experiments wherein media is transmitted over a small internetwork while varying degrees of traffic are introduced into the network. In the first case UDP is used for transport. The video steam is jerky (because of loss) and audio has numerous gaps (because of jitter and loss). Next, our protocol is used for transport. In this case video is marginally better but audio is perfect (although in the case of video, the jerkiness is now due to fact that frames were never transmitted because the protocol is trying to avoid wasting network resources). A quantitative comparison of a similar set of experiments can be found in [3].

TECHNICAL DETAILS

The workstations used in this videotape are IBM PS/2 (20 Mhz x386 processor) personal computers using IBM/Intel ActionMedia I audio/video adapters. We use an experimental real-time operating system kernel and video conferencing application we have developed. The kernel is described in [4, 6]; the application in [4]. The adaptations used in the protocol for managing media streams are described in [3]. A more detailed description and analysis of the delay jitter management scheme used in this work is presented in [5].

The conferencing system generates 60 audio frames and 30 video frames per second. An average video frame is approximately 8000 bytes; an audio frame is approximately 250 bytes. This yields an aggregate data stream of approximately 2 Mb/s.

The network used in these experiments is a building-sized internetwork consisting of several 10 Mb Ethernets and 16 Mb token rings interconnect by bridges and routers. It supports approximately 400 UNIX workstations and Macintosh personal computers. The workstations share a common file system using a mix of NFS and AFS. The application mix running on these workstations should be typical of most academic computer science departments.

ACKNOWLEDGEMENTS

Terry Talley and Ta-Ming Chen helped construct the network and traffic generators used in the demonstrations in this video. David Harrison and Elliot Poger recorded and mastered the first versions of this video. Peggy Wetzel edited and produced the final version.

This work supported in parts by the National Science Foundation (grant numbers CCR-9110938 and ICI-9015443), and the IBM and Intel Corporations.

REFERENCES

[1] Ferrari, D., 1990. *Client Requirements for Real-Time Communication Services*, IEEE Communications, (November), pp. 65-72.

[2] Isaacs, E., Tang, J.C., *What Video Can and Can't Do for Collaboration: A Case Study*, Proc. ACM Multimedia 1993, pp. 199-205.

[3] Jeffay, K., Stone, D.L., and Smith, F.D., *Transport and Display Mechanisms for Multimedia Conferencing Across Packet-Switched Networks*, Computer Networks and ISDN Systems, Vol. 26, No. 10 (July 1994), pp. 1281-1304.

[4] Jeffay, K., Stone, D.L., and Smith, F.D., *Kernel Support for Live Digital Audio and Video*, Computer Communications, Vol. 16, No. 6 (July 1992), pp. 388-395.

[5] Stone, D.L., Jeffay, K., *An Empirical Study of Delay Jitter Management Policies*, ACM Multimedia Systems, *to appear*.

[6] Jeffay, K., Stone, D.L., Poirier, D., *YARTOS: Kernel support for efficient, predictable real-time systems*, Proc. Joint Eighth IEEE Workshop on Real-Time Operating Systems and Software and IFAC/IFIP Workshop on Real-Time Programming, Atlanta, GA, Real-Time Systems Newsletter, Vol. 7, No. 4, Fall 1991, pp. 8-13.

[7] Steinmetz, R., Meyer, T., 1992. *Multimedia Synchronization Techniques: Experiences Based on Different System Structures*, IEEE Multimedia Workshop, Monterey, CA, April, 1992.

[8] Wolf, C., *Video Conferencing: Delay and Transmission Considerations*, in Teleconferencing and Electronic Communications: Applications, Technologies, and Human Factors, L. Parker and C. Olgren (Eds.), 1982.