errors, and a control time of *85 ms* covers all two-packet gaps. Further extensions allow recovery of multiple-packet errors. Since the control time for S-ARQ is directly linked with the packet size used by the voice protocol, larger packet sizes will decrease the effectiveness of the S-ARQ scheme for a fixed end-to-end delay bound. Conversely, shorter packets increase the effectiveness of S-ARQ.

## 4. Conclusions

The S-ARQ scheme is attractive for a number of reasons. First, the S-ARQ scheme does not preclude the use of preventive error control methods such as FEC, and may be used in conjunction with preventive error control. Secondly, it is not network architecture specific (as is channel coding) nor does it require hardware support (as does FEC in high-speed environments). Finally, mechanisms to implement the S-ARQ scheme can be easily incorporated into existing voice packet protocols with little protocol overhead. This is important since packet loss is typically rare, and the gain in quality due to occasional recovery of lost packets must necessarily be balanced against the additional system complexity and any service degradation in the common case of error-free transmissions.

Our continuing research is focused on using the insights gained from developing S-ARQ for packet voice protocols to develop delay-sensitive retransmission schemes for other continuous media applications. During the transfer of image sequences, for example, one common buffering strategy is to overlap the processing of an image with the network transfer of the following image [6]. This technique leads to a natural window of time in which retransmissions are useful, and the framework fits well with the notion of delay-sensitive retransmission schemes.

We are also assessing the feasibility of S-ARQ and similar techniques across different network environments. The results from [1] are encouraging for the application of our new error control scheme to continuous media communication across future wide-area Asynchronous Transfer Mode (ATM) networks. In the wide-area ATM environment, high-speed links and switches will keep network delays low while the statistical multiplexing of cells in switches will cause delay jitter and cell loss. Since the concept of delay-sensitive retransmission is not tied to a particular protocol layer, incorporation of these ideas at the ATM adaptation layer is an open research issue.

## References


[1] Dempsey, B., J. Liebeherr, and A. C. Weaver, "A New Error Control Scheme for Packetized Voice over High-Speed Local Area Networks", CS-93-23, Department of Computer Science, University of Virginia, May 1993.

[2] Krishna, H., and S.D. Morgera, "A New Error Control Scheme for Hybrid ARQ Systems", *IEEE Transactions on Communications*, Vol. COM-35, October 1987, pp. 981-89.

[3] Lin, S., and P. S. Yu, "A Hybrid ARQ Scheme with Parity Retransmission for Error Control of Satellite Channels", *IEEE Transactions on Communications*, Vol. COM-30, July 1982, pp. 1701-19.

[4] Sriram, K. and W. Whitt, "Traffic Smoothing Effects of Bit Dropping in a Packet Multiplexer", *IEEE Transactions on Communications*, Vol. 37, July 1989, pp. 703-712.

[5] Suzuki, J., and M. Taka, "Missing Packet Recovery Techniques for Low-Bit-Rate Coded Speech", *IEEE Journal on Selected Areas in Communications*, Vol. 7, June 1989, pp. 707-717.

[6] Turner, C., and L. Peterson, "Image Transfer: An End-to-End Design", *Proceedings of SIGCOMM '92*, Baltimore, Maryland, August 1992, pp. 258-269.

extant voice protocols, i.e., no retransmissions are attempted. If a packet is lost, playback of the lost packet is skipped. Note that skipping the playback of a lost packet results in a gap in the playback. In scenarios (b) and (c), the receiver requests a retransmission of the lost packet. In both scenarios the lost packet is detected upon arrival of the third packet to the receiver. Scenario (b) shows a retransmission scheme as used for reliable end-to-end protocols. If a packet is lost, playback of all packets is discontinued until the retransmission has been completed. Since the retransmission of the second packet is not completed before its playback time, a gap is observed. In scenario (c) we assume an S-ARQ scheme. Due to the extended control time, the retransmission of the second packet is completed before its playback time.

## 3. Evaluation of S-ARQ

Results from the simulation model developed in [1] establish that S-ARQ is feasible for a range of realistic scenarios running over an FDDI network. Feasibility is an issue since the control time can not be arbitrarily large due to the end-to-end delay requirements for interactive voice. The end-to-end delay requirements for voice data range from between 50 to 500 milliseconds, depending on the channel quality desired and the influence of off-network conversation.

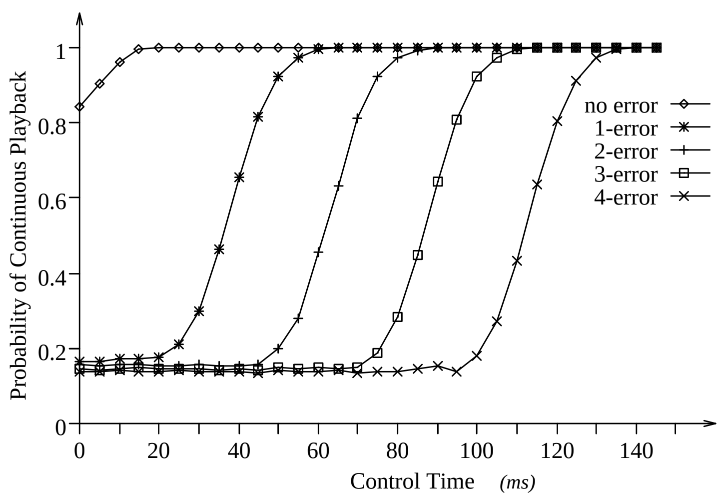


Figure 2: Probability of Continuous Playback.

The FDDI network in the simulation model is taken to be heavily loaded, the condition under which frame loss is non-negligible. We assume that all voice data is transmitted as synchronous traffic. A detailed description of the simulation model and the simulation parameters is given in [1].

Our results indicate that modest extensions to the control time required for delay jitter can provide significant error coverage while respecting end-to-end delay bounds. As an example, Figure 2 shows the effect of increasing the control time on jitter control and on the probability of successful retransmissions if the S-ARQ scheme is applied. Our performance measure is the probability of continuous playback of the voice packets, i.e., the probability that no gaps are observed during the playback of a talkspurt. The *n-error* curve denotes a scenario where in each talkspurt *n* consecutive voice packets are lost. In our example, each talkspurt has an exponentially distributed length with mean *350 ms*, and each voice packet contains data over a sample period of *25 ms*. As shown in Figure 2, the control time necessary to handle network delay jitter alone is *15 ms* while extending the control time to *60 ms* provides complete coverage for single-packet

We present simulation results that demonstrate the feasibility and effectiveness of retransmission in a realistic scenario. Finally, we report on our on-going efforts to extend the delay-sensitive retransmission scheme to other continuous media applications.

## 2. Delay-Sensitive Retransmission Approach for Voice Traffic

Voice packet protocols provide for mechanisms that account for delay jitter introduced by the network. A common mechanism for ensuring continuous voice playback in the presence of delay jitter is the use of a control time for the first packet in each talkspurt. The retransmission scheme presented in this paper, which is called *Slack Automatic Repeat Request* (S-ARQ), is based on extending the control time for the first packet in each talkspurt to allow for the timely retransmission of lost packets, i.e., before the lost packet is due for playback.

In S-ARQ, whenever a lost packet is detected, the receiver requests a retransmission of the missing packet. The packet voice receiver assumes that a packet is lost if it receives a packet out of sequence. If the retransmission is attempted but it is lost or late, the packet voice receiver does not hold back subsequent correctly received packets nor does it attempt any additional retransmissions of the lost data. Therefore, S-ARQ does not guarantee that lost packets are successfully recovered. The percentage of retransmissions that are completed successfully is largely dependent on the appropriate choice of the control time.

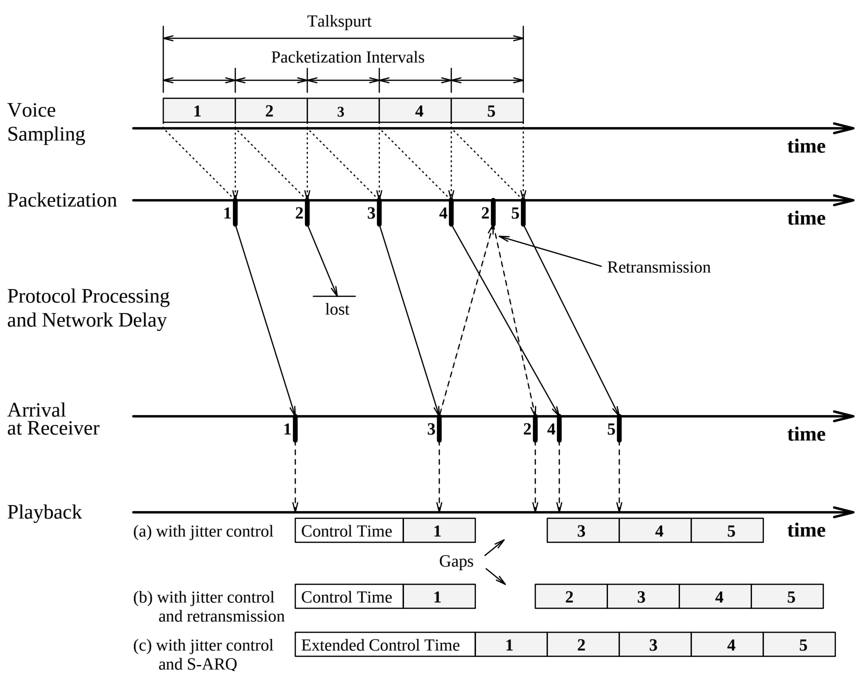


Figure 1: S-ARQ Scheme.

We illustrate the advantages of S-ARQ in Figure 1where we show the transmission of a talkspurt consisting of five packets. We assume that the second packet of the talkspurt is lost. At the bottom of Figure 1 we show three scenarios. For all scenarios we assume the existence of jitter control with an appropriately selected control time at the beginning of the talkspurt. Scenario (a) shows error handling typically found in

# A Delay-Sensitive Error Control Scheme for Continuous Media Communications

*Bert J. Dempsey    Jörg Liebeherr    Alfred C. Weaver*

Computer Science Department
University of Virginia
Charlottesville, VA 22903

**Abstract**

In this paper we present a new error control approach for continuous media communication protocols. Since continuous media data are inherently bursty, these protocols commonly use a control time for the first packet in a burst to ensure continuous playback in the presence of delay jitter. The retransmission scheme presented in this paper, which is called Slack Automatic Repeat Request (S-ARQ), is based on appropriately extending the control time for the first packet in a burst to allow for the timely retransmission of lost packets, i.e., before the lost packet is due for playback. We present simulation results that demonstrate the feasibility and effectiveness of this retransmission scheme under a realistic scenario for an FDDI network.

## 1. Introduction

Digital multimedia applications are demanding network services that respect real-time constraints. In particular, distributing continuous media streams (e.g., digital audio or video) across an asynchronous network for playback at a remote site requires careful attention to the effects of end-to-end packet delay and packet delay variation (*jitter*). Once playback begins, each packet in the stream has an implicit deadline, at which time it must be available at the remote site else the fidelity of signal reconstruction will suffer.

Error control in this context is problematic since an inherent conflict arises between simultaneously meeting delay and error requirements. In reliable end-to-end data protocols (e.g., TCP), all packets lost or corrupted by the network are recovered using some acknowledgment-retransmission scheme, typically Automatic Repeat Request (ARQ). For continuous media communications such error control is inappropriate since a receiver waiting for a retransmission will interrupt the continuous playback of subsequent data. Thus, in contrast to error control in reliable end-to-end protocols, the purpose of error control mechanisms for continuous media streams is to improve the quality of service provided by delivering more data in a timely fashion than would be delivered without these mechanisms.

The focus in error control for continuous media applications has been to reduce the impact of losses on channel quality through *preventive error control* techniques such as forward-error correction (FEC) [2][3], priority channels [4], and error masking during signal reconstruction [5][6]. However, all known preventive error control schemes either substantially increase bandwidth requirements, or increase the complexity of the end-to-end processing path.

In this paper, we show that retransmission-based error control schemes can be successfully employed for continuous media data transfer. Different from error control approaches for reliable protocols, lost or corrupted data is only recovered when the latency penalty for retransmission does not violate the delay constraints of the traffic flow. In the following, we describe a retransmission scheme for voice traffic.