Adaptive Playout Control Schemes for Speech over the Internet

Marco Roccetti
University of Bologna, Italy

Stefano Ferretti
University of Bologna, Italy

INTRODUCTION

The transmission of speech over the Internet is often dismissed as an impractical application because of the poor quality experienced by many users of Internet audio tools. In fact, while Internet audio services are required to operate in a bandwidth-, delay-, and packet loss-constrained environment, the actual best-effort service offered by the Internet architecture does not provide any guarantee on the delivery of data packets. Thus, it often occurs that very high packet delay, delay jitter and packet loss are experienced over many congested Internet links.

To ameliorate the effect of network delay, delay jitter and packet loss rate, novel protocol suites and networking technologies (e.g., RSVP, ATM networks) have been devised that provide users with quality of service (QoS) guarantees (Zhang, Deering, Estrin, Shenker & Zappala, 1993). However, these approaches are not yet widely used and, as a consequence, in the absence of network support to provide guarantees of QoS to users of Internet voice tools, an interesting alternative amounts to the use of adaptive playout control mechanisms at the receiver side. Those approaches are adopted by the majority of the existing Internet audio tools, such as *NeVot* (Schulzrinne, 1992), *vat* (Jacobson & McCanne, n.d.), *rat* (Hardman, Sasse & Kouvelas, 1998), *FreePhone* (Bolot & Vega Garcia, 1996) and *BoAT* (Roccetti, Ghini, Pau, Salomoni & Bonfigli, 2001).

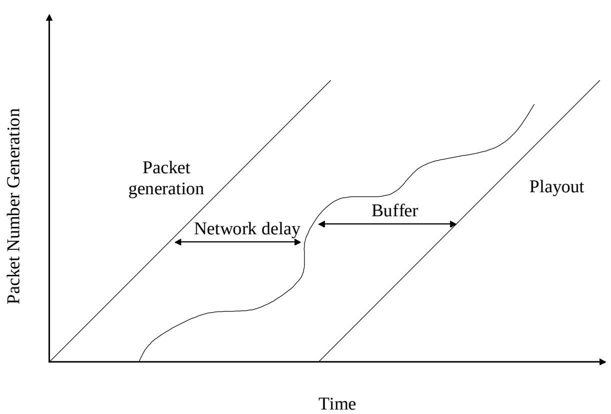
In essence, to compensate for variable network delays and remove jitter from audio packet streams, adaptive playout control mechanisms use a voice reconstruction buffer at the receiver and add artificial delay to the audio stream to smooth out the delay jitter. Simply put, received audio packets are first queued into the buffer and then the periodic playout of each packet is delayed for some quantity of time. This buffering policy must be adaptive, since jitter on the Internet may vary significantly with time. In this way, dynamic playout buffers can hide at the receiver packet delay variance at the cost of additional delay (see Figure 1).

It goes without saying that, even if this approach permits the removal of jitter from audio packet streams, and guarantees the speech intelligibility, a critical trade-off exists between the amount of delay that is introduced in the buffer and the percentage of late packets that are not received in time for playout (and are consequently lost). In fact, the longer the additional delay, the more likely it is that a packet will arrive before its scheduled playout time. Summing up, on one side, a too large percentage of audio packet loss may impair the intelligibility of an audio transmission, but, on the other side, too large playout delays (due to buffering) may disrupt the interactivity of an audio conversation (Kostas, Borella, Sidhu, Schuster, Grabiec & Mahler, 1998; Panzieri & Roccetti, 1997).

In conclusion, Internet playout control mechanisms adaptively adjust the playout delay of audio packets in order to keep this additional buffering delay as small as possible, while minimizing the number of packets delayed past the point at which they are scheduled to be played out (Boutremans & Le Boudec, 2003; Fujimoto, Ata & Murata, 2004; Liang, Farber & Girod, 2003; Sreenan, Chen, Agrawal & Narendran, 2000).

With this in view, in the remainder of this article we survey three different adaptive playout delay control mechanisms that have been designed to support speech

Figure 1. Smoothing out jitter delay at the receiver



transmission over the Internet. In particular, all three different approaches adopt adaptive control mechanisms that keep the same playout delay constant throughout a given talkspurt, but permit different playout delays in different talkspurts.

BACKGROUND

In this section we survey three different adaptive playout delay control schemes designed to transport speech over the Internet.

Playout Control Mechanism - #1

The adaptive playout delay adjustment algorithm proposed in Ramjee, Kurose, Towsley, and Schulzrinne (1994) is used in several Internet audio tools, such as *NeVoT* (Schulzrinne, 1992), *rat* (Hardman et al., 1998) and *FreePhone* (Bolot & Vega Garcia, 1996). This algorithm assumes that an external mechanism exists that keeps synchronized the two system clocks at both the sending and the receiving sites and that the delays experienced by audio packets on the network follow a Gaussian distribution. The mechanism works as follows.

The receiver buffers packets and delays their playout for a time quantity that is usually adaptively adjusted from one talkspurt to the next one. In essence, this mechanism works by calculating the playout time p_i for the first packet i of a given talkspurt as:

$$p_i = t_i + \underline{d}_i + k * \underline{v}_i,$$

where t_i is the time at which the audio packet i is generated at the sending site, d_i is the average value of the playout delay, $k \in [1, 2, 4]$ is a variation coefficient (whose effect can be enforced through shift operations) and v_i is the average variation of the playout delay. The playout point p_j for any subsequent packet j of that talkspurt is computed as an offset from the point in time when the first packet i in the talkspurt was played out:

$$p_j = p_i + t_j - t_i.$$

The estimation of both the average delay and the average delay variation are carried out using a stochastic gradient algorithm.

This strategy is also equipped with a delay spike detection and management mechanism. In essence, the algorithm works as described here, but when a spike is detected the mechanism uses the delay of the first packet of the talkspurt as the estimated playout delay for each packet in the talkspurt, in order to effectively react to very large change in transmission delays.

Playout Control Mechanism - #2

Another adaptive delay adjustment algorithm for speech transmission has been presented in Moon, Kurose, and Towsley (1998). The main idea behind this algorithm is to collect statistics on packets already arrived and then use them to calculate the playout delay.

In essence, the value of each packet delay is recorded and the distribution of packet delays is updated with each new arrival. When a new talkspurt starts, this mechanism calculates a given percentile point for an established amount of last arrived packets, and uses it as the playout delay for the new talkspurt.

In the presence of delay spikes the algorithm stops collecting packet delays as soon as a delay spike is detected, and starts following the spike by using as playout delay the delay value experienced by the packet that commenced that spike.

Playout Control Mechanism - #3

Another mechanism designed to dynamically adapt the talkspurt playout delays to the network traffic conditions has been proposed in Roccetti et al. (2001). This mechanism is at the basis of an Internet audio tool, called *BoAT*.

The mechanism is able to dynamically adjust the talkspurt playout delays to the network traffic conditions without assuming either the existence of an external mechanism for maintaining an accurate clock synchronization between the sender and the receiver, or a specific distribution of the end-to-end transmission delays experienced by the audio packets.

Succinctly, the technique for dynamically adjusting the talkspurt playout delay is based on obtaining, in periodic intervals, an estimation of the upper bound for the packet transmission delays experienced during an audio communication. Such an upper bound is periodically computed using round-trip time values obtained from packet exchanges of a three-way handshake protocol performed between the sender and the receiver of the audio communication. Then, the upper bound is used to dynamically adjust the playout delay from one talkspurt to the next, with the introduction of artificially elongated or reduced silence periods.

CRITICAL ISSUES

The need of silent intervals for allowing a playout delay control mechanism to adjust to the fluctuating network conditions is common to all three Internet audio mechanisms described in the previous section. This renders all the described control schemes particularly suitable for