Adaptive FEC-Based Error Control for Internet Telephony

Jean-Chrysostome Bolot Sacha Fosse-Parisis
INRIA
2004 route des Lucioles
06902 Sophia Antipolis Cedex
France
www.inria.fr/rodeo/{bolot,sfosse}

Don Towsley
Department of Computer Science
University of Massachusetts at Amherst
Amherst, MA 01003-4610
USA
towsley@cs.umass.edu

Abstract—

Excessive packet loss rates can dramatically decrease the audio quality perceived by users of Internet telephony applications. Recent results suggest that error control schemes using forward error correction (FEC) are good candidates for decreasing the impact of packet loss on audio quality. With FEC schemes, redundant information is transmitted along with the original information so that the lost original data can be recovered at least in part from the redundant information. Clearly, sending additional redundancy increases the probability of recovering lost packets, but it also increases the bandwidth requirements and thus the loss rate of the audio stream. This means that the FEC scheme must be coupled to a rate control scheme. Furthermore, the amount of redundant information used at any given point in time should also depend on the characteristics of the loss process at that time (it would make no sense to send much redundant information when the channel is loss free), on the end to end delay constraints (destination typically have to wait longer to decode the FEC as more FEC information is used), on the quality of the redundant information, etc. However, it is not clear given all these constraints how to choose the "best" possible redundant information.

We address this issue in the paper, and illustrate our approach using a FEC scheme for packet audio recently standardized in the IETF. We show that the problem of finding the best redundant information can be expressed mathematically as a constrained optimization problem for which we give explicit solutions. We obtain from these solutions a simple algorithm with very interesting features, namely i) ***the algorithm optimizes a subjective measure*** **(such as the audio quality perceived at a destination) as opposed to an objective measure of quality (such as the packet loss rate at a destination), ii)** ***it incorporates the constraints of rate control and playout delay adjustment schemes*****, and iii)** ***it adapts to varying loss conditions in the network*** **(estimated online with RTCP feedback).**

We have been using the algorithm, together with a TCP-friendly rate control scheme and we have found it to provide very good audio quality even over paths with high and varying loss rates. We present simulation and experimental results to illustrate its performance.

I. Introduction

The transmission of real time audio, and especially of real time voice, over the Internet has been much in the news recently. Traditional voice carriers, so-called next-gen telcos, as well as manufacturers of gateways, phone-like appliances, and routers, have all become involved in some way or another with Internet telephony. Internet telephony is branded by the various parties as fitting anywhere between "the old-telco killer app" and "a toy for long distance lovers". In any case, it is clear that the field of packet voice over the Internet has matured and that the basic building blocks are available [25], ranging from high quality codecs to standardized packetization and signaling protocols such as RTP [24], H.323 [11], or SIP [26]. Still, Internet telephony has often been dismissed as a "real" application because of the mediocre quality experienced by many users of Internet voice software.

Audio quality problems are not so surprising because the current Internet provides users with a single class best effort service which does not promise anything in terms of performance guarantees. And indeed, measurements show persistent problems with audio quality caused by congestion in the network, and thus by the impact of traffic in the network on the streams of audio packets. In practice, this impact is felt via high loss rates, varying delay, etc.

In the absence of network support to provide guarantees of quality (such as a maximum loss rate or a maximum delay) to users of audio tools, a promising approach to tackle the problems caused by varying loss rates, delays, or available bandwidth, is to use application level control mechanisms. These mechanisms adapt the behavior of the audio application so as to eliminate or at least minimize the impact of loss, jitter, etc, on the quality of the audio delivered to the destinations.

Efficient playout adjustment mechanisms have been developed to minimize the impact of delay jitter [16]. Much recent effort has been devoted to developing mechanisms to minimize the impact of loss. Rate control mechanisms attempt to minimize the number of packets lost by ensuring that the rate at which audio packets are sent over a connection matches the capacity of the connection [5]. However, they typically do not prevent loss altogether. An error con-

0-7803-5417-6/99/$10.00 ©1999 IEEE.

trol, or loss recovery, mechanism is required if the number of lost audio packets is higher than that tolerated by the listener at the destination.

Typical mechanisms fall into one of two classes [19]. Automatic Repeat Request (ARQ) mechanisms are closed-loop mechanisms based on the retransmission of the packets that were not received at the destination. Forward Error Correction (FEC) mechanisms are open-loop mechanisms based on the transmission of redundant information along with the original information so that some of the lost original data can be recovered from the redundant information. ARQ mechanisms are typically not acceptable for live audio applications over the Internet because they dramatically increase end to end latency[1].

FEC is an attractive alternative to ARQ for providing reliability without increasing latency. FEC schemes send redundant information along with the original information so that the lost original data can be recovered, at least in part, from the redundant information. There are two main issues with FEC. First, the potential of FEC mechanisms to recover from losses depends in large part on the characteristics of the packet loss process in the network. Indeed, FEC mechanisms are more effective when the average number of consecutively lost packets is small. Second, sending additional redundancy increases the probability of recovering lost packets, but it also increases the bandwidth requirements and thus the loss rate of the audio stream. This means that the FEC scheme must be coupled to a rate control scheme. Furthermore, the amount of redundant information used at any given point in time should also depend on the characteristics of the loss process at that time (it makes no sense to send redundant information when the channel is loss free), on the end to end delay constraints (destination typically have to wait longer to decode the FEC as more FEC information is used), on the quality of the redundant information, etc. The problem, then, becomes a constrained optimization problem, namely: given constraints of the rate control mechanisms (i.e. given a total rate at which the source can send), find the combination of main and redundant information which provides the destination with the best perceived audio quality. It is precisely the goal of this paper to formalize this problem, solve it, derive a practical algorithm, apply it to the FEC scheme recently standardized in the IETF [18], and evaluate the performance of the algorithm in realistic Internet environments using a real Internet audio/telephony tool.

The paper is organized as follows. In Section II, we first briefly review recent results on the the loss process of audio packets in the Internet. We then describe a simple FEC scheme which uses these results to minimize an objective function (the loss rate after packet reconstruction) at the destination. However, that scheme turns out to have a number of drawbacks. We describe in Section III our main contribution, namely an adaptive algorithm for the IETF FEC scheme which incorporates the constraints of rate control and playout delay adjustment, which adapts to varying loss conditions, and which maximizes a subjective measure of quality (such as the perceived audio quality at a destination) as opposed to a measure such as the packet loss rate at a destination which does not reflect the quality perceived by the receiver. We present simulation and experimental results in Section IV to illustrate the performance of the algorithm.

[1]However, they would be appropriate in low delay environments, or with relaxed end to end delay constraints [6].

II. A simple FEC-based error control scheme

The loss process of audio packets

We mentioned in Section Ithat the characteristics of the loss process of audio packets are important in determining which type of error control scheme (ARQ or FEC) to use for error recovery. Recent work has shown that the correlation structure of the loss process of audio packets can be modeled by a low order Markov chain, such as a two state Gilbert model, and that the distribution of the number of packets lost in a loss period is approximately geometric, or, rather, that the head of the distribution is geometric, and that the tail includes a few events (which might contribute significantly to the overall loss rate, since a single event in the tail indicates that a loss period with a large number of lost packets) which appear not to have any specific structure [23], [4], [29], [31]. This is consistent with other, more general, results on end to end Internet loss characteristics (e.g. [3], [17]).

We will use in the rest of the paper a few basic results about the Gilbert model. Therefore, recall that a Gilbert model is a 2-state Markovian model in which one state (which we refer to as state 1) represents a packet loss, and the other state (which we refer to as state 0) represents a packet reaching the destination. Let p denote the probability of going from state 0 to state 1, and let q denote the probability of going from state 1 to state state 0. The probability that n consecutive packets are lost equals $(1-q)q^{n-1}$, and, thus, the residence time for state 1 is geometrically distributed. Refer to Figure 1. Note that when

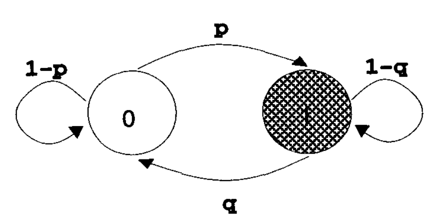

Fig. 1. The Gilbert model

$p+q=1$, the model turns into a Bernoulli model.

The results on the loss process of audio packets in the Internet mentioned above show that the median number of consecutively lost packets is small, and thus, FEC is particularly well suited for live audio applications over the Internet. A large variety of FEC schemes have been proposed in the literature [19], based on parity and block erasure codes [21], convolutional codes [2], interleaving [14], or multiple description codes [28]. We consider in this paper another scheme, which was recently standardized in the IETF [18] and which, thus, we can expect many Internet audio applications to rely on for robustness with respect to packet loss. Furthermore, it appears to provide good subjective results even in the face of high loss rates [10].

This scheme evolved from an earlier scheme [7], in which a packet, say n, includes, in addition to its encoded samples, information about the previous packet, $n-1$. If packet $n-1$ is lost but packet n is not lost, then the destination can use that information to reconstruct (an approximation to) packet $n-1$. The information about packet $n-1$ considered in [7] includes a discretized energy envelope as well as the number and/or the location of zero crossings of the waveform encoded in packet $n-1$.

The IETF scheme relies on an idea similar to the above, i.e. packet n includes, in addition to its encoded samples, a redundant version of packet $n-1$. However, the redundant information about packet $n-1$ is now obtained with a low bit rate encoding of packet $n-1$. Consider for example the case when audio is sent using PCM encoding. Then LPC, GSM, or CELP coders could be used to obtain the redundant information. Clearly, the mechanism can be used to recover from isolated losses: if packet n is lost, the destination waits for packet $n+1$, decodes the redundant information, and sends the reconstructed samples to the audio driver. With redundant LPC audio, the output consists of a mixture of PCM- and LPC-coded speech. Somewhat surprisingly, the subjective quality of this reconstructed speech has been found to be quite good [10].

The scheme described above only recovers from isolated losses. However, it can be modified to recover from consecutive losses as well by including in packet n redundant versions of packets $n-1$ and $n-2$, or packets $n-1$, $n-2$ and $n-3$, or packets $n-1$ and $n-3$, etc. Note that the scheme then can be thought of as some kind of generalized interleaving: interleaving because the information relative to packet n is spread over multiple packets, and generalized because each interleaved chunk can be decoded by itself independent of the others. Also, it is clear that the more redundant information is added at the source, the more lost packets can be reconstructed at the destination. However, it would make little sense to add much redundant information when the loss rate is very low. Thus, we would like to choose the appropriate combination of redundant information given the loss process in the network at any given point in time. We consider this issue next.

A simple adaptive algorithm for the IETF FEC scheme

The choice of redundant information depends on the benefit we get from adding redundant information. To answer this question, we model the loss process in the presence of redundancy so as to find the perceived loss rate after reconstruction.

Recall that in the absence of redundant information, the loss rate is $\pi_1 = \frac{p}{p+q}$. Consider now the case when packet n includes redundant information about packet $n-1$ only: a packet is lost only if it cannot be reconstructed using the redundant information, i.e. the packet is lost and the next packet is lost as well. It is then straightforward to show that the loss rate after reconstruction is now

$$\pi_2 = \frac{p(1-q)}{p+q}$$

The ratio between π_2 and the loss rate without redundancy is equal to $(1-q)$. With q around 0.70 (a value we have typically found in traces collected between European universities), we see that adding one piece of redundancy decreases the perceived loss rate by 70%.

We can carry out a similar analysis and examine cases with two, three, four pieces of redundant information, etc. The results are summarized in Table I. In the column "Redundancy", the notation -1-3, for example, means that redundant information about packets $n-1$ and $n-3$ was sent in packet n.

Redun.	Loss rate after reconstruction
none	$p/(p+q)$
-1	$(p(1-q))/(p+q)$
-2	$(p^2q+p(1-q)^2)/(p+q)$
-3	$p(3pq-p^2q-2q^2p+1-3q+3q^2-q^3)/(p+q)$
-1-2	$(p(1-q)^2)/(p+q)$
-1-3	$(p(1-q)(pq+1-2q+q^2))/(p+q)$
-2-3	$(p(1-q)(pq+1-2q+q^2))/(p+q)$
-1-2-3	$(p(1-q)^3)/(p+q)$

TABLE I

LOSS RATES AFTER RECONSTRUCTION

The simplest way to build a control mechanism is to have a target perceived loss rate (i.e. loss rate after reconstruction) at the destination, and to have the source choose the amount of redundant information that will yield the loss rate closest to the target loss rate. Of course, this requires that the source know p and q. Unfortunately, RTCP receiver reports (RRs) [24] only include information about the mean loss rate, i.e. $p/(p+q)$, but not about p and q separately. There are two ways around this. The first way is to use other fields in RTCP RRs to include p and q (we used the jitter field, which nobody seems to be using or intending to use). The other way is to assume that the loss process is Bernoulli, not Gilbert, i.e. to assume that

$p + q = 1$; then the loss rate p is the rate reported in the RTCP RRs.

Figure 2 shows the evolution with time of the loss rate measured over a connection between INRIA and London, and of the loss rate after reconstruction over the same connection when the algorithm described above is used, the target loss rate being 3%.

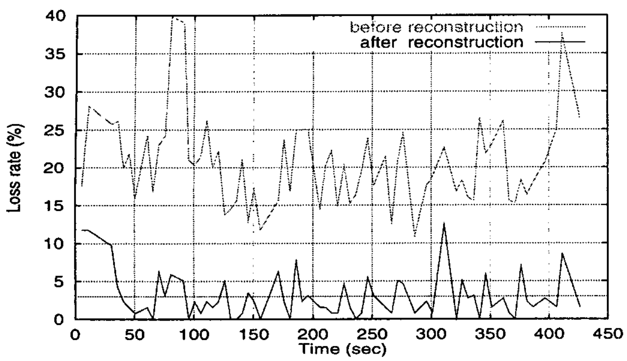

Fig. 2. Loss with and without FEC; target loss is 3%

The algorithm does provide the destination with a perceived loss rate which fluctuates around the desired loss rate, even though the loss rate in the network varies between 12 to 40%. The fluctuations are caused in part because the loss process in the network is not a Bernoulli process, and the value of p only is not enough to capture all of its characteristics, and in part because the RTCP feedback is sent back by the destination only every 5 seconds [24].

The figure above makes the algorithm appear attractive. However, it suffers from two drawbacks. First, it minimizes an objective performance measure (loss rate after reconstruction), instead of a measure tied to audio quality: in practice, it would make little sense to be able to reconstruct all lost packets if the quality of the information used for audio reconstruction (i.e. the quality of the redundant information) is too low to be understandable. Second, adding redundant information increases the bandwidth requirements of the source. Therefore, we need to tie the process of adding redundant information to a rate control scheme. In practice, we combine the rate control and the error control mechanisms into one joint rate/error control mechanism. The goal then is to adjust at the source both the send rate and the amount of redundant information to minimize the perceived loss rate at the destinations. We describe one such scheme next.

III. An optimal joint rate/FEC-based error control scheme

Main Results

Consider a voice source that has the flexibility to encode its samples at a rate $x \in [0, \infty)$ (or $[0, D]$ if one prefers). The quality of the encoding of the sample is given by a function $f : \mathbb{R}^+ \rightarrow \mathbb{R}$ which is an increasing concave function with derivative g. Note that g is necessarily non-increasing.

The source transmits voice packets to a receiver over an unreliable network characterized by a two-state continuous time Markov chain $\{X_t\}$ where $X_t \in \{0, 1\}$ If a packet is transferred at time t, then the packet is not lost if $X_t = 0$; the packet is lost if $X_t = 1$. The infinitesimal generator of this Markov chain is

$$\mathbf{Q} = \begin{bmatrix} -\mu_0 & \mu_0 \\ \mu_1 & -\mu_1 \end{bmatrix}$$

The stationary distribution associated with this chain is $\pi = (\pi_0, \pi_1)$ where $\pi_0 = \mu_1/(\mu_0 + \mu_1)$ and $\pi_1 = \mu_0/(\mu_0 + \mu_1)$. Note that π_1 corresponds to the probability that a packet is lost.

We consider the case when we use the FEC-based error control scheme described earlier. Let $K - 1$ denote the maximum number of redundant pieces of information sent along with the main information. Thus, packet n carries information about at most (i.e. a subset of) packets $n - 1$, ..., $n - K + 1$. Therefore, the total number of copies (encoded at different rates, including 0) of a given audio packet sent by the source equals K. In practice, the larger K, the longer the destination has to wait to receive the redundant information to reconstruct missing packets, and thus the longer the end to end delay. We characterize the delay constraint of the interactive audio application of interest here by a delay T, which is the delay between sending the first and the last copy of a given packet.

The first question that we ask ourselves then is

Q1. Given that we will transmit K copies of each voice packet and we have a delay constraint of T by which the last packet can be transmitted, how should we space the packets so as to maximize the probability that at least one packet is received?

Before providing a more precise formulation of this problem, we introduce the following conditional probabilities. Let $p_{i,j}(t)$ denote the probability that the process is in state j at time $t + \tau$ given that it was in state i at time τ, $p_{i,j}(t) = P(X_{\tau+t} = j \mid X_\tau = i)$. These probabilities are easy to derive, for example

$$\begin{aligned} p_{11}(t) &= (\mu_0 + \mu_1 \exp(-(\mu_0 + \mu_1)t)/(\mu_0 + \mu_1) \\ &= \pi_1 + \pi_0 \exp(-(\mu_0 + \mu_1)t) \end{aligned}$$

Let t_k denote the interval between the times at which the kth and $(k+1)$-st copies of a voice packet are sent. The probability that the first $K - 1$ copies of a packet are lost is equal to $\pi_1 \prod_{k=1}^{K-1} p_{11}(t_k)$. Thus, question **Q1** above can be formulated as the optimization problem with linear constraints below:

$$\text{Maximize} \quad 1 - \pi_1 \prod_{k=1}^{K-1} \left(\pi_1 + \pi_0 e^{-(\mu_0+\mu_1)t_k}\right),$$

$$\text{such that} \quad t_k \geq 0, \quad \sum_{k=1}^{K} t_k \leq T$$

It should be clear that the last constraint will always be satisfied with equality because the cost function is an increasing function of t_i. This problem falls in the general category of bit allocation problems with rate constraints [27], and use standard techniques based on Lagrange multipliers to solve it; refer to [1] for details. It is easy to show that $t_k = T/(K-1)$ is the optimal solution. This means that **the K copies must be equally spaced in the interval $[0, T]$ including both endpoints.** This is a welcome result and an *a posteriori* support for the FEC scheme under study here, since redundancy data in that scheme is sent precisely at regular intervals (piggybacked on audio packets).

We now address the following question:

Q2. Given that K copies are to be transmitted equally spaced in an interval of length T, what encoding rates should be used for each copy so as to maximize the quality of the transfer subject to a rate constraint?

Let R denote the rate available to the audio flow of interest. We assume that a value for R is available at any given point to the source, but we do not make any assumption as to how R is computed. In practice, R is obtained as a result of a rate control algorithm. In the current Internet, R might be computed using a linear-increase exponential decrease scheme or a TCP-friendly scheme [30], [15] (in practice, we use a scheme based on the result in [15]). However, R could also be computed using explicit feedback such as ECN bit(s) or the explicit rate messages in ABR.

Define the rv $\mathcal{S}$ to be $\mathcal{S} = \{i \mid X_i = 0, i = 1, \ldots, K\}$, i.e. the set of copies of a packet that make it across the network. Question **Q2** can be stated mathematically as follows.

$$\text{Maximize} \quad \sum_{\mathcal{S} \subseteq \{1,\ldots,K\}} P(\mathcal{S}) \max_{i \in \mathcal{S}} f(x_i)$$

$$\text{s.t.} \quad x_i \geq r_0, \quad \sum_{i=1}^{K} x_i \leq R$$

where x_i is the encoding rate for the packet placed in the i-th position. Here r_0 is the minimum rate used to encode all samples.

It is important to observe that the formulation of the optimization problem above assumes that the different copies of an audio packet cannot be combined to produce a better quality copy of the original packet. Indeed, we measure quality at the destination using only the "best" (i.e. largest $f(x)$) copy of a packet. In other words, if the l-th best quality copy is not lost, it is used in the case that the best, 2-nd best, ... $(l-1)$-best quality copies are lost. The formulation would be different with layered or hierarchically encoded copies (we have examined those cases but do not report on them here because of lack of space). We focus on the formulation above in this paper because of space constraints, and because it ties in with the schemes proposed in [18].

The problem above appears to be, in general, difficult to solve. We do not derive the solution here for lack of space, but instead describe the results, which are as follows:

1) x_1 is greater than all other x_i's, meaning that **the main information should be encoded using the highest quality coding scheme (among those used to encode the main and the redundant information).**

2) $x_1 \geq x_K \geq$ other x_i's, meaning that **it pays to put more quality into the end packets**. In particular, if only two copies of a packet are to be sent, then these copies should be x_1 (main information) and x_K (redundant information that goes as far back as allowed).

3) for $K = 2, 3, 4, 5$, the solution tells us exactly which copies should be encoded with the better quality schemes.

The explicit results (result 3) have been obtained for $K = 2, 3, 4, 5$ only. However, it is important to observe that **results 1) and 2) are valid for any K**. Indeed, they essentially rely on the fact that $p_{10}(t)$ is an increasing function of t.

Discrete Rate Optimization

The analysis above assumes that the encoding rate at each copy of a packet could take on any real value. In practice, of course, there is a countable set of rates available to the encoder, say $\mathcal{R} = \{r_i\}_{i=0}^{n}$. Without loss of generality, we assume that $r_i < r_{i+1}$, $i = 0, 1, \ldots$. Let f remain non-decreasing concave. We now define the "derivative" of f as follows $g(i) = (f(r_i) - f(r_{i-1}))/(r_i - r_{i-1})$. The concavity of f implies that g is non-increasing. Our optimization problem can now be posed as follows

$$\text{Maximize} \quad \sum_{\mathcal{S} \subseteq \{1,\ldots,K\}} P(\mathcal{S}) \max_{i \in \mathcal{S}} f(x_i)$$

$$\text{s.t.} \quad x_i \in \mathcal{R}, \quad \sum_{i=1}^{K} x_i \leq R$$

Again, this is not an easy problem to solve. However, the optimal solution exhibits some of the same properties as the solution to the continuous rate problem. The algorithm in Figure 3 provides a simple and computationally cheap [1] way to find an approximate solution to the above problem. The algorithm provides a non optimal solution, however with reasonable properties, in particular 1) the resulting solution is $x_i = r_{k_{i-1}}$, $i = 1, \ldots, n-1$, and the quality of the solution differs from that of the optimum by at most $(g(r_{k_j}) - g(r_{k_j - 1}))a_j$ where j resulted in the algorithm halting, 2) the solution can be improved by just checking to see if any of the other x_i's can be increased without violating the rate constraint, and 3) if $r_i = i \times r_0$, then the solution is optimal.

$S = \{1\}$; $n = 1$; $k_1 = 0$; $r = R$;
Repeat forever
  Choose $j \in S$ st $g(k_j)a_j$ is maximum;
  if $r_{k_j} - r_{k_j - 1} \leq r$
  then
    $k_j = k_j + 1$;
    $r = r - r_{k_j} - r_{k_j - 1}$
    if $j = n$ **then**
      $n = n + 1$; $k_n = 0$; $S = S \cup \{n\}$;
  else halt;

Fig. 3. Discrete rate optimization algorithm.

We have also examined the case of a non-concave utility function. We do not present it here for lack of space.

IV. Evaluating the scheme

We have implemented the joint rate/error control scheme described in the previous section in the FreePhone audio tool [8]. We next present some experimental results showing how the algorithm fares in practice. In particular, we consider how the optimal FEC allocation varies as a function of the utility function f, of the delay constraint T, and of the rate constraint R.

Utility functions

We have taken pain in the paper to consider mechanisms that would optimize a *subjective* measure of audio quality as perceived at a destination. Unfortunately, it is a well known fact that there is no agreed upon objective measure which captures the audio quality perceived by a user as a function of coding rate, loss rate in the network, etc. Subjective measures such as intelligibility, comfort of hearing, and mean opinion score (MOS), are hard to quantify. Objective measures such as loss rate or signal to noise ratio are related in complex and not always clear ways to subjective measures. For example, packet loss has a "generic" negative impact on quality because information is lost. However, it has a more subtle impact on quality depending on which type of coding scheme is used - for example, schemes that require that some state be kept about past packets to encode future packets (such as in G.729) are more sensitive to packet loss than other schemes [12], [22]. The signal to noise ratio, on the other hand, is sensitive to the characteristics of the signal, and hence to different sentences being spoken. Thus, in the absence of reliable objective functions, we have considered four sample functions, shown in Figure 4. The first function is defined by $f_0(x) = x$, the second function $f_1(x)$ was obtained by measurements of signal/noise ratios with the different codecs we consider here (namely the LPC, GSM, ADM4, ADM6, and PCM coders mentioned earlier), the third curve $f_2(x)$ was obtained from values of MOS available in the literature for our codecs, and the last function $f_3(x)$ is defined by $f_3(x) = 1$ for $x \neq 0$ and $f_3(0) = 0$.

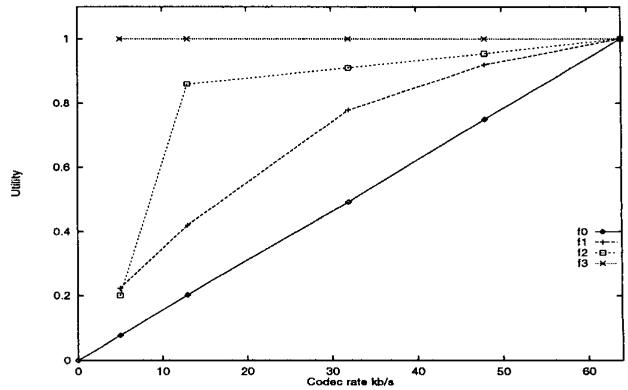

Fig. 4. Utility functions

We chose f_0 and f_3 because they yield two interesting ways of adding redundancy. Specifically, with f_0, the optimal allocation is always to send the main information encoded with the highest possible rate, and to send no redundant information at all[2]. Regarding f_3, note that f_3 is maximum as long as *some* information is received, no matter what the subjective quality of this information. Thus, using f_3 in our algorithm amounts to minimizing the loss rate after reconstruction at the destination, which is what we were doing back in Section II. It is easy to see that, in this case, the optimal policy is to send as many redundant packets as possible no matter how small the coding rate, as long as it fits within the constraints of the rate control mechanism.

How well does it work?

We have used adaptive FEC schemes in FreePhone for quite some time now and we have found them to provide very good average quality. This is illustrated in the figures below, which present measurements obtained over a connection between INRIA in southern France, and London in the UK. The loss rate over that connection is typically high, it was about 13% when the measures were taken. Suppose then that utility function f_3 has been chosen as the appropriate utility function, i.e. the goal is to minimize the number of lost packets at the destination. Figure 5 shows the evolutions as functions over time of the loss rate at the destination, computed over intervals of 128 packets, before and after reconstruction. For that experiment, we had $T = 4$.

[2]This can be derived by replacing $f(x_i)$ with x_i in section II and working out the optimization problem directly by hand.

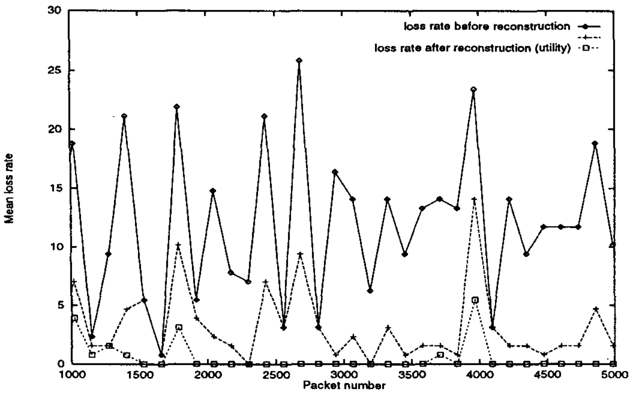

Fig. 5. Evolutions of the loss rate and the utility (loss rate after reconstruction) over time, $f = f_3$

We observe that the loss rate after reconstruction is 0 much of the time, and it remains close to 0 even as the loss rate in the network varies between 1 and more than 25%. Clearly, the adaptive FEC scheme does a good job at improving utility even in the face of high and highly varying loss rate. Note also that the quality of the control scheme in Figure 5 is much better than that obtained with the simpler scheme in Figure 2 because i) the algorithm now takes into account bandwidth and delay constraints instead of just adding redundant information (and thus modifying bandwidth requirements) independent of the bandwidth available in the network for the connection, and ii) it finds the optimal combination of redundant information instead of just picking a combination in Table I which gives a loss rate after reconstruction close to a pre-specified target loss rate.

We have considered the case when the optimization is done for utility function f_3, because it makes it easier to compare with the earlier results in Figure 2. We have considered other functions as well. However, it is hard to illustrate the impact of subjective functions using objective measures, i.e. without listening to actual conversations or sound files sent over the Internet. We have found in practice that the best subjective quality by far is obtained with function f_2, i.e. the utility function which most closely matches MOS scores. We have also found that there is very little difference in terms of subjective quality between optimizing for f_0 ($f_0(x) = x$) and optimizing for f_1 (SNR). Recall that optimizing for f_0 amounts to not using any redundant information at all not matter what the loss rate; thus it is not surprising that the resulting quality is typically poor. This, however, also means that optimizing for the SNR (f_1) yields a poor quality as well, further proof that the SNR is not a reliable indicator of perceived audio quality.

Impact of the maximum delay T on quality

In the figures above, we had $T = 4$. Clearly, the higher T, the better the quality at the destination, but the larger the delay requirements. We now examine the impact of varying T on the quality achievable at the destination.

Figure 6 shows the average perceived quality at the destination for different values of T, and for the different utility functions described above. We make two observations.

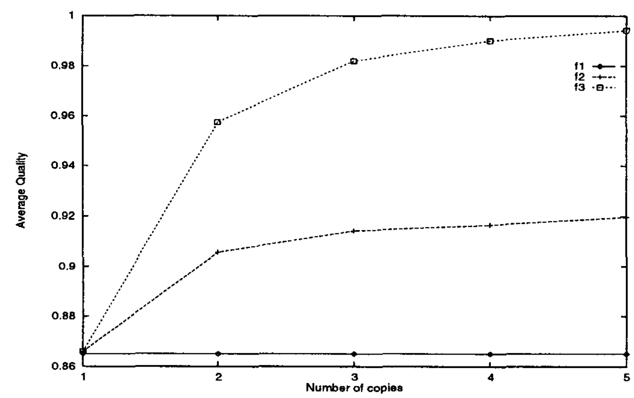

Fig. 6. Perceived quality at the destination as a function of the delay T (or the number of redundant copies)

First, the quality increases dramatically as T goes from 1 to 2. This indicates that adding just one piece of redundant information about packet $n-1$ in packet n makes a big difference in quality. This is consistent with the subjective results shown in [10]. We also observe that the quality perceived at the destination is essentially constant for f_1 now matter how much extra redundant information is added at the source. This is because f_1 is in fact very close to f_0, and f_0 yields an optimal FEC allocation that precisely does not include any redundant information at all (recall our discussion above).

The second observation is that the quality varies dramatically as a function of f. This indicates that i) algorithms that attempt to minimize an objective measure of quality such as the loss rate after reconstruction (i.e. they assume that $f = f_3$) yield very different performance from algorithms that maximize some kind of perceived audio quality, and thus that ii) it is important to get reliable data on subjective quality so as to be able to rely on reasonable curves for f.

V. Conclusion

Various FEC schemes for multimedia applications in the Internet have been proposed recently. However, they have to be handled carefully since adding FEC to a stream generated by a multimedia source increases the bandwidth requirements of that stream. The problem then is, given rate constraints imposed by a congestion control algorithm and given network conditions that can vary over time, to find the FEC information that will provide the destination with the best quality possible at any given point in time.

We have derived in this paper one such adaptive algorithm, which provides very good performance with the "signal processing" FEC scheme for audio recently standardized in the IETF. Of course, even our "optimal" scheme cannot provide guaranteed quality given the best effort service model of the current Internet. However, it puts us one step closer to quasi-constant quality audio even over connections with high or highly varying loss rates.

We are pursuing this work in several directions. One is to develop adaptive FEC schemes suitable for multicast groups (in the spirit of, for example, [13]). Another one is to consider more sophisticated FEC mechanisms, in particular mechanisms based on multiple description codes [28]. Yet another one is to use our technique to solve similar problems in other areas. Indeed, our approach is not restricted to the particular FEC scheme we focused on in this paper, nor to FEC schemes for audio applications only. One very interesting type of applications would be distributed gaming [9]. The idea there would be to use FEC to achieve an "almost reliable" *and* timely delivery of important information such as collisions/explosions or state changes. We can use the results in the paper to send multiple copies of the information encoded at different rates (and thus with different "granularities") so as to maximize the quality perceived by the destinations. Consider for example information about a bridge hit by a rocket. It is important for players to know whether the bridge can be crossed or not, less important to know the status of subparts of the bridge, etc. Thus, the information related to the bridge can be described with varying levels of detail, and thus sent with varying encoding rates (ranging from one bit "the bridge can be crossed or not" to as many bits as required for a full screen, accurate rendition of the bridge). The FEC scheme would then be supplemented by a reliable multicast delivery scheme such as SRM so as to make sure that information eventually gets delivered to all participants. We are currently evaluating such a scheme.

References

[1] D. Bertsimas, J. Tsitsiklis, *Introduction to Linear Optimization*, Athena Scientific, 1997.

[2] R. Blahut, *Theory and Practice of Error Control Codes*, Addison-Wesley, 1983.

[3] J-C. Bolot, "End-to-end packet delay and loss behavior in the Internet", *Proc. ACM Sigcomm '93*, San Fransisco, CA, pp. 189-199, Aug. 1993.

[4] J-C. Bolot, A. V. Garcia, "The case for FEC-based error control for packet audio in the Internet", to appear in *ACM/Springer Multimedia Systems*, 1999.

[5] J-C. Bolot, A. V. Garcia, "Control mechanisms for packet audio in the Internet", *Proc. IEEE Infocom '96*, San Fransisco, CA, pp. 232-239, Apr. 1996.

[6] B. Dempsey, J. Liebeherr, A. Weaver, "On retransmission-based error control for continuous media traffic in packet-switching networks", *Computer Networks and ISDN Systems*, vol. 28, no. 5, pp. 719 - 736, March 1996.

[7] N. Erdöl, C. Castelluccia, A. Zilouchian, "Recovery of missing speech packets using the short-time energy and zero-crossing measurements", *IEEE Trans. Speech and Audio Proc.*, vol. 1, no. 3, pp. 295-303, July 1993.

[8] FreePhone http://www.inria.fr/rodeo/fphone

[9] L. Gautier, C. Diot, "Design and evaluation of MiMaze, a multiplayer game on the Internet", *Proc. IEEE Multimedia Systems Conf.*, Austin, TX, June 1998.

[10] V. Hardman et al., "Reliable audio for use over the Internet", *Proc. INET '95*, Honolulu, HI, pp. 171-178, June 1995.

[11] ITU-T, *Recommendation H.323 - Visual Telephone Systems and Equipment for Local Area Networks which Provide Non-Guaranteed Quality of Service*, Feb. 1996.

[12] N. Jayant, S. W. Christensen, "Effects of packet losses in waveform-coded speech", *IEEE Trans. Comm.*, vol. COM-29, pp. 101-109, Feb. 1981.

[13] G. Karlsson, "Layered error-control coding for IP multicast", *Proc. HIPPARCH workshop*, Uppsala, Sweden, June 1997.

[14] A. Lapidoth, "The performance of convolutional codes on the block erasure channel using various finite interleaving techniques", *IEEE Tran. Information Theory*, vol. IT-40, pp. 1459-1473, Sept. 1994.

[15] M. Mathis, J. Semke, J. Mahdavi, T. Ott, "Macroscopic behavior of the TCP congestion avoidance algorithm", *Computer Comm Review*, vol. 27, no. 3, July 1997.

[16] S. B. Moon, J. Kurose, D. Towsley, "Packet audio playout delay adjustments: performance bounds and algorithms", *ACM/Spinger Multimedia Systems*, vol. 5, pp. 17-28, Jan. 1998.

[17] V. Paxson, "End-to-end Internet packet dynamics", *Proc. ACM Sigcomm'97*, Cannes, France, Sept. 1997.

[18] C. Perkins et al., "RTP payload for redundant audio data", RFC 2198, Sept. 1997.

[19] C. Perkins, O. Hodson, "A survey of packet loss recovery techniques for streaming media", *IEEE Network*, Sept/Oct 1998.

[20] M. Podolsky, C. Romer, S. McCanne, "Simulation of FEC-based error control for packet audio on the Internet", *Proc. IEEE Infocom'98*, San Fransisco, CA, April 1998.

[21] L. Rizzo, "Effective erasure codes for reliable computer communication protocols", *Computer Communication Review*, April 1997.

[22] J. Rosenberg, "G.729 error recovery for Internet telephony", Columbia University report, spring 1997.

[23] H. Schulzrinne, J. Kurose, D. Towsley, "Loss correlation for queues with bursty input streams", *Proc. IEEE ICC '92*, Chicago, IL, pp. 219-224, 1992.

[24] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A transport protocol for real-time applications", RFC 1889.

[25] H.Schulzrinne, J. Rosenberg, "Internet telephony, architecture and protocols: an IETF perspective", to appear in *Computer Networks and ISDN Systems*.

[26] H.Schulzrinne, J. Rosenberg, "Signaling for Internet telephony", *Proc. ICNP '98*, Austin, TX, Oct. 1998.

[27] Y. Shoham, A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers", *IEEE Trans. Acoustics, Speech, Signal Proc.*, vol. 36, no. 9, pp. 1445-53, Sept. 1988.

[28] V. A. Vaishampayan, "Design of multiple description scalar quantizers", *IEEE Trans. Info. Theory*, vol. 39, no. 3, pp. 821-834, May 1993.

[29] D. Veitch, "Understanding end-to-end Internet traffic dynamics", *IEEE Globecom 98*, Sydney, Australia, Nov. 1998.

[30] L. Vicisano, L. Rizzo, J. Crowcroft, "TCP-like congestion control for layered multicast data transfer", *Proc. IEEE Infocom '98*, San Francisco, CA, March 1998.

[31] M. Yajnik, S. B. Moon, J. Kurose, D. Towsley, "Measurement and modeling of the temporal dependence in packet loss", *IEEE Infocom'99*, New York, NY, March 1999.