1998

Adaptive FEC-Based Error Control for Interactive Audio in the Internet

Jean Bolot
Sophia Antipolis Cedex, France

Adaptive FEC-Based Error Control for Interactive Audio in the Internet*

Jean Bolot   Sacha Fosse-Parisis
INRIA
BP 93
06902 Sophia Antipolis Cedex
France
{bolot,sfosse}@sophia.inria.fr

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

June 1998

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 how to choose the ”best” possible redundant information given all the constraints mentioned above.

We address this issue in the paper, and illustrate our approach using a FEC scheme for packet audio recently standardized in the IETF. We find that the problem 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: i) it optimizes a subjective measure of quality (such as the perceived audio quality at a destination) as opposed to a non subjective measure (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 (and estimated on line with RTCP) loss conditions in the network.

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

* Submitted to *IEEE Infocom '99*.

1 Introduction

The transmission of real time audio, and especially of real time voice, over the Internet has been much in the news recently. An increasingly large number of companies sell Internet telephony software and supporting hardware ("Net phones", gateways [25], phone-like appliances [1]), while traditional voice carriers quietly investigate the matter. Internet telephony is branded by the various parties involved as fitting anywhere between "the 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 [33], ranging from high quality low bit rate codecs to standardized protocols such as RTP [32] or H.323 [15]. Still, Internet telephony is often dismissed as an application that might sweep the nations 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[1].

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 high loss rates or varying delays is to use control mechanisms. These mechanisms adapt the behavior of the audio application so as to eliminate or at least minimize the impact of packet loss, delay 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 [27, 20]. 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 making sure 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 control, 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 in one of two classes. 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 (at least 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[2].

FEC is an attractive alternative to ARQ for providing reliability without increasing latency [34, 3]. 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 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

[1]We note, though, that a mediocre audio quality might also be caused by problems having little to do with the network service. The experience accumulated over the past few years (e.g. with the audiocasting of IETF meetings) suggests that badly tuned or set up microphones and speakers are responsible for many such problems. However, all these can be addressed by users at their own sites. Furthermore, their impact is expected to decrease as users become familiar with the tools and the tools themselves become more user friendly.

[2]However, they would be appropriate in low delay environments, or with relatex end to end delay constraints [9].

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 for the FEC scheme recently standardized in the IETF [23] which implements the solution, and evaluate the performance of the algorithm in realistic Internet environments with a real Internet audio/telephony tool.

The paper is organized as follows. In Section 2, 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 3 our main contribution, namely an adaptive algorithm for the IETF FEC scheme which incorporates the constraints of rate control and playout delay adjustment schemes, 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 4 to illustrate the performance of the algorithm.

2 A simple FEC-based error control scheme

The loss process of audio packets

We mentioned in the Introduction that the characteristics of the loss process of audio packets are important to determine which type of error control scheme (ARQ or FEC) to use for error recovery. Thus, it is not surprising to find previous work in that area. The main result, obtained using analytic models [31] and confirmed with measurements over the Internet [5, 39, 13], has been 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) but it does not appear to have any specific structure.

Unfortunately, the result above does not say much about the characteristics of the loss process because it only mentions the marginal distribution of the process, but it says nothing about the correlation structure of that process. For example, it could be claimed that the result above is consistent with a Bernoulli or a Gilbert loss process (i.e. 1- and 2-state Markov processes with geometrically distributed residence times). And indeed, it is true that the distribution of the number of packets lost with a Bernoulli or a Gilbert model is geometric, but the converse (a geometric distribution implying a Bernoulli or Gilbert process) is not true. However, recent results using model order estimation with entropy and minimum description code approaches [19] do indeed show that a Gilbert model, i.e. a 2-state Markov process, is a good model for the loss process observed in traces[3] (e.g. [40]). This is consistent with other, more general, results on end to end Internet characteristics (e.g. [22]).

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 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. Then the residence times for states 0 and 1 are both geometrically distributed with means $1/p$ and $1/q$, respectively. The probability that n consecutive packets are lost is equal to $(1-q)q^{n-1}$, and thus the residence time for state 1 is geometrically distributed. Refer to

[3] Note that this does not say that a Gilbert model is the model that best fits the traces, but rather that a Gilbert model best strikes a balance between better fit and increased model complexity (or equivalently increased number of states).

Figure 2. Note that when $p + q = 1$, the model turns into a (1-state) Bernoulli model.

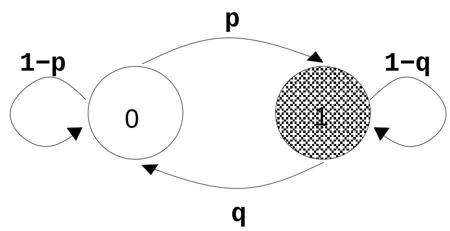

Figure 1: The Gilbert model

In any case, the results mentioned above show that the median number of consecutively lost packets is small, and thus that FEC is particularly well suited for live audio applications over the Internet. A large variety of FEC mechanisms have been proposed in the literature. Many such mechanisms involve exclusive-OR operations, the idea being to send every nth packet a redundant packet obtained by exclusive-ORing the other n packets [34]. This mechanism can recover from a single loss in a n packet message. It is a very simple mechanism, but it increases the send rate of the source by a factor of $1/n$, and it adds latency since n packets have to be received before the lost packet can be reconstructed. More sophisticated algorithms such as Reed-Solomon codes are not well suited to interactive audio because they require that data be broken up into blocks, which in practice would be of large size. Thus, the use of such schemes would add a non negligible "block delay" to the end to end delay, thereby decreasing the quality of interactivity between participants.

Other schemes based on block erasure codes [29], convolutional codes [4], interleaving [18], or multiple description codes [37] have been examined. However, we consider in this paper another scheme, referred to as a "signal processing" FEC mechanism [24]. We focus on this particular scheme because it was recently standardized in the IETF [23] and thus we can expect many audio applications to rely on it for robustness with respect to loss in the Internet, and because it appears to provide good subjective results even in the face of high loss rates [14].

That scheme evolved from an earlier scheme [10], in which packet $n + 1$ includes, in addition to its encoded samples, information about packet n which can be used to reconstruct (an approximation to) packet n. The redundant information about packet $n - 1$ considered in [10] includes i) a discretized energy envelope and the number of zero crossings of the waveform encoded in packet $n - 1$, and ii) a discretized energy envelope and the location of zero crossings of the waveform encoded in packet $n - 1$. Not surprisingly, scheme ii) provides slightly better quality at the expense of slightly higher bandwidth requirements.

The IETF scheme relies on an idea similar to that 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 [14].

The scheme described above only recovers from isolated losses. However, it can be modified to recover from consecutive losses as well by using in packet n redundant versions of packets $n - 1$ and $n - 2$, or of packets $n - 1$, $n - 2$ and $n - 3$, or of 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

To choose a good combination of redundant information, we need to know how much benefit we get from adding extra redundant information. To answer this question, we model the loss process in 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

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

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 redundant 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 the table below. To illustrate the results and show how much can be gained by using redundancy, we also show the loss rate after reconstruction for a loss process with parameters $p = 0.05$ and $q = 0.7$.

Redundancy			Loss rate after reconstruction	Example
$n-1$	$n-2$	$n-3$		$p = 0.05, q = 0.7$
-	-	-	$p/(p+q)$	6.7%
X	-	-	$(p(1-q))/(p+q)$	2%
-	X	-	$(p^2q + p(1-q)^2)/(p+q)$	0.83%
-	-	X	$(p(3pq - p^2q - 2q^2p + 1 - 3q + 3q^2 - q^3))/(p+q)$	0.54%
X	X	-	$(p(1-q)^2)/(p+q)$	0.6%
X	-	X	$(p(1-q)(pq + 1 - 2q + q^2))/(p+q)$	0.25%
-	X	X	$(p(1-q)(pq + 1 - 2q + q^2))/(p+q)$	0.25%
X	X	X	$(p(1-q)^3)/(p+q)$	0.18%

Table 1: Loss rates after reconstruction

As expected, adding redundant information decreases the perceived loss rate. We note that if only one piece of redundant information is used, then it is best to use information about packet $n-3$, rather than information about packet $n-1$ or $n-2$. Similarly, if two pieces of redundant information are used, then it is best to use information about packets $n-1$ and $n-3$, or about $n-2$ and $n-3$, rather than information about $n-1$ and $n-2$. In general, it appears best to use information about packet $n-K$, where K characterizes how far back we can go to use redundant information. We will get back to this observation in the next section.

It is also clear from the table that the amount of redundant information should be chosen with care. For example, if the loss rate in the network is low, choosing the last combinations (bottom rows) in the table would be an overkill. Thus, we need a mechanism to adjust the amount of redundancy added at the source based on the loss process in the network. The simplest way to do this 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 knows p and q. Unfortunately, RTCP receiver reports (RRs) [32] 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). 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 evolutions 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%.

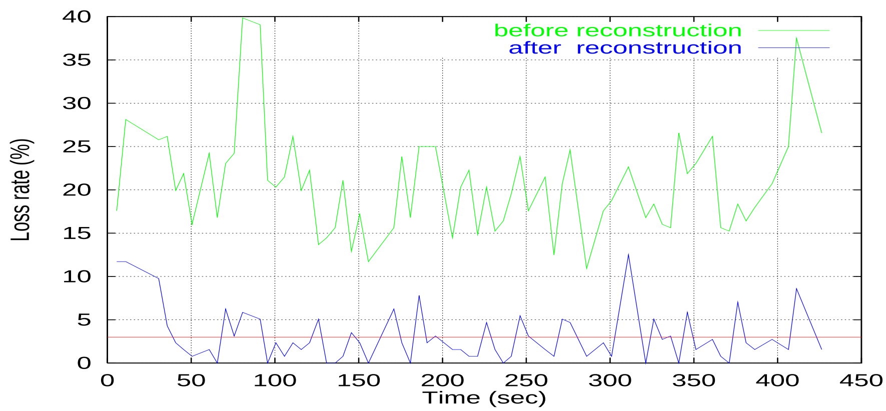

Figure 2: Loss rate with and without reconstruction; target loss is 3%

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 [32].

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.

3 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 $f : \mathbb{R}^+ \to \mathbb{R}$ 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 is equal to 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, $p_{i,j}(t) = P(X_{\tau+t} = j \,|\, X_\tau = i)$. These probabilities are given by ([21, ch. 6])

$$\begin{cases} p_{10}(t) = \mu_1(1-\exp(-(\mu_0+\mu_1)t)/(\mu_0+\mu_1) = \pi_0(1-\exp(-(\mu_0+\mu_1)t)) \\ \\ p_{01}(t) = \mu_0(1-\exp(-(\mu_0+\mu_1)t)/(\mu_0+\mu_1) = \mu_1(1-\exp(-(\mu_0+\mu_1)t)) \\ \\ p_{11}(t) = (\mu_0+\mu_1\exp(-(\mu_0+\mu_1)t)/(\mu_0+\mu_1) = \pi_1+\pi_0\exp(-(\mu_0+\mu_1)t) \\ \\ p_{00}(t) = (\mu_1+\mu_0\exp(-(\mu_0+\mu_1)t)/(\mu_0+\mu_1) = \pi_0+\pi_1\exp(-(\mu_0+\mu_1)t) \end{cases}$$

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:

$$\begin{array}{lll} \text{Maximize} & 1-\pi_1\prod_{k=1}^{K-1}\left(\pi_1+\pi_0 e^{-(\mu_0+\mu_1)t_k}\right), & \\ & & \\ \text{such that} & t_k \geq 0, & k = 1,\ldots,K \\ & \sum_{k=1}^{K} t_k \leq T & \end{array}$$

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. A solution to this problem is equivalent to a solution of the following problem

$$\begin{array}{lll} \text{Minimize} & \sum_{k=1}^{K-1}\log(\pi_1+\pi_0 e^{-(\mu_0+\mu_1)t_k}), & \\ & & \\ \text{s.t.} & t_k \geq 0, & k = 1,\ldots,K \\ & \sum_{k=1}^{K} t_k = T & \end{array}$$

This problem falls in the general category of bit allocation problems with rate contraints [36, 8], and use standard techniques based on Lagrange multipliers to solve it. Recall that the problem of finding an extremum x_0 to the function $f(x)$ subject to the m linear constraints $h_i(x) = b_i$, $1 \leq i \leq m$ can be replaced by that of finding an extremum, *not subject to constraints*, to another function $F(x, \alpha)$ referred to as the Lagrangian. Specifically, we consider

$$F(x, \alpha) = f(x) + \sum_{j=1}^{m} \alpha_j (b_j - h_j(x))$$

where $\alpha = (\alpha_1, ..., \alpha_m)$ are called the Lagrange multipliers, and we solve the $m + 1$ equations

$$\begin{cases} \partial F / \partial x = 0 \\ \\ \partial F / \partial \alpha_j = 0 \quad j = 1, \ldots, m \end{cases}$$

In other words, it is necessary for an extremum of f that the Lagrangian F satisfies the (classical) unconstrained necessary conditions as a function of the $m + 1$ variables x and α_j. It turns out that there exists a Lagrange multiplier such that necessary conditions of the form above hold in the more general case when the constraints are expressed in terms of equalities and inequalities, for example, $h_i(x) = b_i$, $1 \leq i \leq k$ and $h_i(x) \leq b_i$, $k + 1 \leq i \leq m$, $k < m$. Refer to [2] for details.

We can now apply these techniques to solve our minimization problem above. Because the cost function is convex and the constraints define a convex set, the optimal choice of t_k must satisfy the conditions

$$\begin{aligned} \frac{\pi_0 e^{-(\mu_1+\mu_0)t_k}}{\pi_1 + \pi_0 e^{-(\mu_0+\mu_1)t_k}} &= \alpha, \quad t_k > 0, \ k \geq 1 \\ \frac{\pi_0 e^{-(\mu_0+\mu_1)t_k}}{\pi_1 + \pi_0 e^{-(\mu_0+\mu_1)t_k}} &\leq \alpha, \quad t_k = 0, \ k \geq 1. \end{aligned}$$

where α is a Lagrange multiplier chosen so that $\sum_{k=1}^{K} t_k = T$. From this we conclude that t_k does not depend on k, and since $\sum_k t_k = K$, it follows 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).

The next question we ask is

Q2. How many copies do we need in order to achieve a perceived loss probability γ at the destination?

First, it is necessary to check whether a loss probability of γ can be achieved with the constraint that the first and last copy of the voice packet cannot be spaced more than T time units apart. γ is achievable if

$$\left(1 - \pi_1 e^{-(\mu_0+\mu_1)T}\right) < \gamma.$$

If this is the case, then the minimum number of copies is given by $\lceil K_0 \rceil$ where K_0 is the solution of

$$1 - \pi_1 p_{11}(T/(K_0 - 1))^{K_0} = \gamma$$

and thus

$$1 - \pi_1 \left(\pi_1 + \pi_0 e^{-(\mu_0+\mu_1)T/(K_0-1)}\right)^{K_0} = \gamma.$$

Henceforth, we assume that K copies of the sample will be transmitted with the k-th copy transmitted at time $t_k = (k-1)T/(K-1)$, $i = 1, \ldots, K$ (here time has been normalized so that the first copy is transmitted at time

$t_1 = 0$).

We now address the following question:

Q3. 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 such as that described in [7] or a TCP-friendly scheme [38, 11] (in practice, we use the scheme described in [11]). 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 $\mathcal{S} = \{i \,|\, X_i = 0, i = 1, \ldots, K\}$, i.e. the set of copies of a packet that make it across the network. Question **Q3** can be stated mathematically as follows.

$$\begin{array}{lll} \text{Maximize} & \sum_{\mathcal{S} \subseteq \{1,\ldots,K\}} P(\mathcal{S}) \max_{i \in \mathcal{S}} f(x_i), & \\ & & \\ \text{s.t.} & x_i \geq r_0, & i = 1, \ldots, K \\ & \sum_{i=1}^{K} x_i \leq R & \end{array}$$

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. This appears to be, in general, a difficult problem. However, it is easy to solve for small values of K, which fortunately are the values of interest in practice. We describe below a simple way of deriving a general result, and of deriving complete results for the case $K = 2, 3, 4$ (and we conjecture $K \geq 5$). A more rigorous and general approach is presented in the Appendix.

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 focus on the formulation above in this paper because of space constraints, and because it ties in with the schemes proposed in [23].

We now get back to solving the optimization problem. Let a_i denote the probability that the first successfully transmitted copy of the packet is the i-th copy. Then the solution to the optimization problem above is also the solution of the following problem

$$\begin{array}{lll} \text{Maximize} & \sum_{i=1}^{K} a_i f(x_i), & \\ & & \\ \text{s.t.} & x_i \geq r_0, & i = 1, \ldots, K \\ & \sum_{i=1}^{K} x_i \leq R & \end{array}$$

The solution to this problem must satisfy the following relations

$$\begin{array}{rcll} a_i g(x_i) & = & \alpha, & x_k > r_0, i = 1, \ldots K, \\ a_i g(x_i) & \leq & \alpha, & x_k = r_0, i = 1, \ldots, K \end{array}$$

where α is a Lagrange multiplier chosen so that $\sum_{i=1}^{K} x_i = R$.

We can now derive the values of $a_i, i = 1, \ldots, K$. It is easy to see that

$$a_i = \begin{cases} \pi_0, & i = 1 \\ \pi_0 p_{0,1}(T), & i = 2 \end{cases}$$

for $K = 2$,

$$a_i = \begin{cases} \pi_0, & i = 1 \\ \pi_0 p_{0,1}^2(T/2), & i = 2 \\ \pi_0 p_{0,1}(T), & i = 3 \end{cases}$$

for $K = 3$,

$$a_i = \begin{cases} \pi_0, & i = 1 \\ \pi_0 p_{0,1}(T/3) p_{0,1}(2T/3), & i = 2 \\ \pi_0 p_{0,1}^2(T/3) p_{1,1}(T/3), & i = 3, \\ \pi_0 p_{0,1}(T), & i = 4 \end{cases}$$

for $K = 4$ and

$$a_i = \begin{cases} \pi_0, & i = 1 \\ \pi_0 p_{0,1}^2(T/4) p_{1,1}(T/2), & i = 2 \\ \pi_0 p_{0,1}^2(T/2), & i = 3, \\ \pi_0 p_{0,1}^2(T/4) p_{1,1}^2(T/4), & i = 4, \\ \pi_0 p_{0,1}(T), & i = 5 \end{cases}$$

for $K = 5$.

We now use the definition of the a_is and simple monotonicity properties of the exponential function to derive monotonicity properties of the x_is. For example, comparing a_2 and a_3 for $K = 3$ and remembering that $1 - \exp(-t) \leq (1 - \exp(-t/2))^2$ for $t \geq 0$ implies that $x_3 \geq x_2$. We find that

$$\begin{aligned} x_1 \geq x_2, &\qquad K = 2, \\ x_1 \geq x_3 \geq x_2, &\qquad K = 3, \\ x_1 \geq x_4 \geq x_2 \geq x_3, &\qquad K = 4, \\ x_1 \geq x_5 \geq x_3 \geq x_2 \geq x_4, &\qquad K = 5. \end{aligned}$$

Thus, we find that:
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). This is in agreement with what we saw back in Table 1. However, the result here is far more general since it is valid for any function f and any rate control constraint.
3) for $K = 2, 3, 4, 5$, the results above tell us exactly which copies should be encoded with the better quality schemes.

The explicit results above (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

$$\begin{array}{lll} \text{Maximize} & \sum_{\mathcal{S} \subseteq \{1,\ldots,K\}} P(\mathcal{S}) \max_{i \in \mathcal{S}} f(x_i), & \\ & & \\ \text{s.t.} & x_i \in \mathcal{R}, & i = 1, \ldots, K \\ & \sum_{i=1}^{K} x_i \leq R & \end{array}$$

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 solution to the above problem is a solution to the following problem

$$\begin{array}{lll} \text{Maximize} & \sum_{i=1}^{K} a_i f(x_i), & \\ & & \\ \text{s.t.} & x_i \in \mathcal{R}, & i = 1, \ldots, \\ & \sum_{i=1}^{K} x_i \leq R & \end{array}$$

where a_i was defined earlier. The algorithm in Figure 3 provides a simple and computationally cheap [2] 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;

Figure 3: Discrete rate optimization algorithm.

A Non-concave Utility Function

The analysis so far assumes that the utility function f is concave. However, an important class of reasonable utility functions include functions that are convex near the origin, then concave for $x \geq x_0$ for some x_0 [35]. Let us assume then that f takes the following form

$$f(x) = \begin{cases} 0, & x < r_0, \\ & \\ b(x), & x \geq r_0 \end{cases}$$

where $r_0 > 0$ and b is concave increasing with derivative $g(x)$ (having inverse h).

We can pose the same optimization problem as before. Unfortunately, as stated it is no longer a simple concave non-linear programming problem. However, we pose the following new problem. Let $\mathcal{S} \subset \{1, 2, \ldots\}$ such that $|\mathcal{S}| \leq R/r_0$.

$$\begin{array}{lll} \text{Maximize} & \sum_{i \in \mathcal{S}} a_i b(x_i), & \\ & & \\ \text{s.t.} & x_i \geq r_0, & i \in \mathcal{S} \\ & \sum_{i \in \mathcal{S}} x_i \leq R & \end{array}$$

If we can solve this, then it suffices to determine the appropriate set $\mathcal{S}$ that produces the maximum quality. Although in principle this appears difficult, due to the monotonicity property of a_i, it is possible to show that $\mathcal{S}$ takes the form $\mathcal{S} = \{1, \ldots, n_0\}$ for some n_0, and that the problem becomes that of determining the solution to

$$\begin{array}{lll} \text{Maximize}_{1 \leq n_0 \leq \lfloor R/r_0 \rfloor} & \sum_{i=1}^{n_0} a_i b(x_i), & \\ & & \\ \text{s.t.} & x_i \geq r_0, & i = 1, \ldots n_0 \\ & \sum_{i=1}^{n_0} x_i \leq R & \end{array}$$

The optimum value of n_0 can be obtained by doing a binary search in the interval $[1, \lfloor R/r_0 \rfloor]$ using the following relations to obtain the $\{x_i\}$ for a specific n_0,

$$\begin{aligned} a_i g(x_i) &= \alpha, \quad x_i > r_0, \; i \geq 1 \\ a_i g(x_i) &\leq \alpha, \quad x_i = r_0, \; 2 \leq i \leq n_0 \end{aligned}$$

where α is a Lagrange multiplier that is again chosen so that

$$\sum_{i=1}^{\infty} x_i = R.$$

4 Evaluating the scheme

We have implemented the joint rate/error control scheme described in the previous section in the FreePhone audio tool [12]. Figure 4 shows a screen dump of the redundancy control panel of FreePhone (this will be available in release 3.7). For debugging purposes, it is possible to choose the number and type of redundant information by hand (click on ”Manual”) or to choose among a set of predefined combinations of main and redundant information (click on ”Predef-profiles”). By default, the joint rate/redundancy control scheme takes over. The ”Action” field indicates which type of utility function f is used in the optimization process (this is discussed further below). A small panel (”Receiver”) shows information about the effectiveness of the FEC scheme at the receiver, characteristics of the loss process (estimations of p and q), etc.

We have been using the tool for quite some time now. 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 of

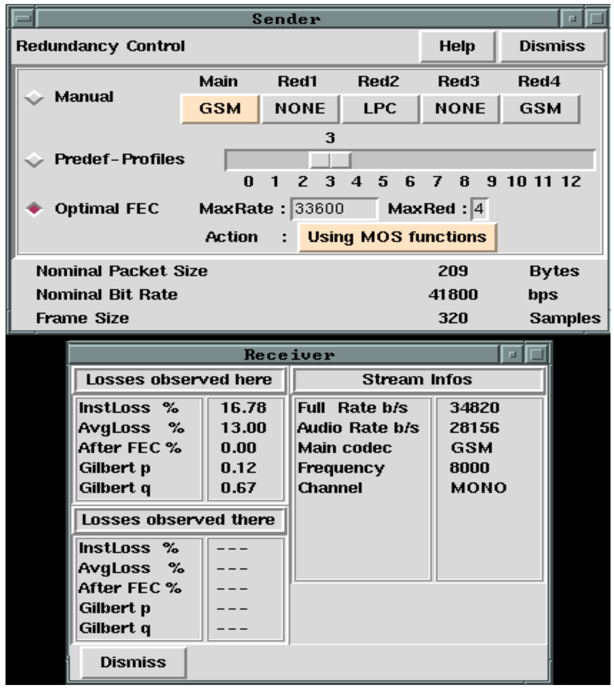

Figure 4: Redundancy control panel of FreePhone3.7

quality that 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 [16, 30]. The signal to noise ratio, on the other hand, is sensitive to the characteristics of the signal, and hence to different sentences being spoken. For example, Figure 5 shows the signal to noise ratio obtained by encoding 8 different samples $s_1 - s_7$ (different sentences in two different languages, 5 in French, 2 in English, spoken by a native French speaker) with 5 different coding algorithms, namely LPC (4.8 kb/s), GSM (13 kb/s), ADM4 (32 kb/s), ADM6 (48 kb/s) and PCM (64 kb/s). The different sentences yield somewhat similar looking curves, however we note that some curves are concave, while other are convex then concave[4].

Thus, in the absence of reliable objective functions, we have considered four sample functions, shown in Figure 6. The first function is defined by $f_0(x) = x$, the second function $f_1(x)$ is defined by the signal/noise ratio curve of sample s_4 in Figure 5, the third curve $f_2(x)$ is obtained from values of MOS available in the literature about the codecs we consider (namely the LPC, GSM, ADM4, ADM6, and PCM coders mentioned earlier), and the last function $f_3(x)$ is defined by $f_3(x) = 1$ for $x \neq 0$ and $f_3(0) = 0$.

We chose f_0 and f_3 the way they are 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[5]. Regarding f_3, note that f_3 is maximum as long as *some* information is received,

[4]In these experiments, the convex then concave curves correspond to sentences in English spoken by the French speaker. We do not have an explanation for this, nor do we know whether the correlation between native/non lative language and concave/non concave curve is significant.

[5]This is easy to derive by replacing $f(x_i)$ with x_i in section 2 and working out the optimization problem directly by hand.

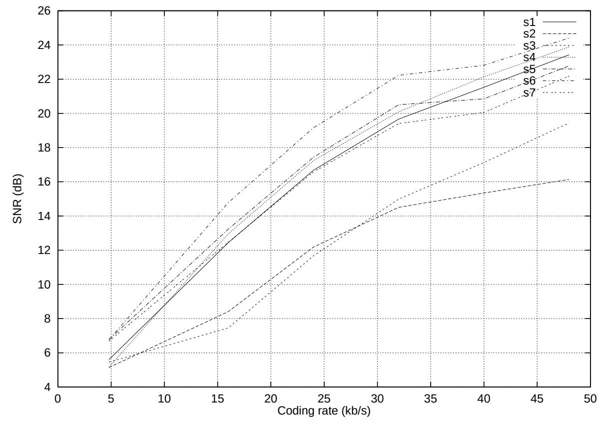

Figure 5: Signal/noise ratio computed for 7 audio files encoded with LPC, GSM, ADM4, ADM6, and PCM coders

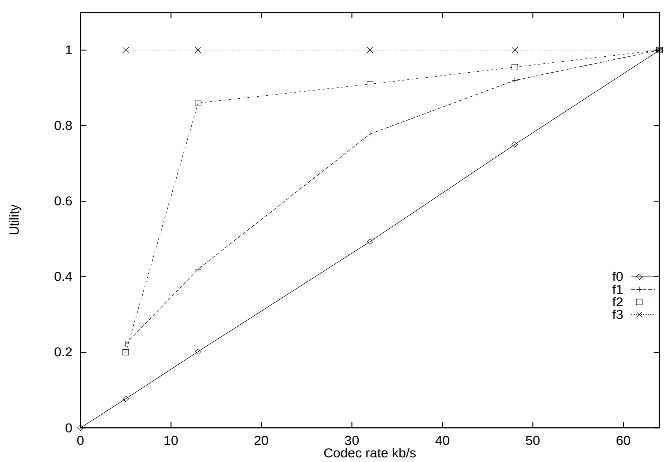

Figure 6: Utility functions

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 2. 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.

Figure 7 shows the evolutions of the parameters p and q of the Gilbert model. Not surprisingly, we find that the variance of q is larger than that of p, since there are relatively few loss events over each averaging period (128 packets), or at least more packets typically make it across the connection than are lost.

Suppose then that utility function f_3 has been chosen as the appropriate utility function, i.e. the goal is to minimize

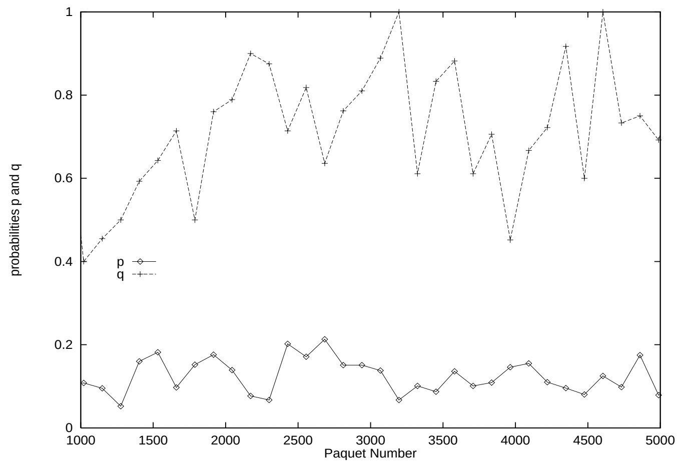

Figure 7: Evolutions of p and q over time

the number of lost packets at the destination. Figure 8 shows the evolutions as a function of time of the loss rate at the destination, computed over intervals of 128 packets, before and after reconstruction. For that experiment, we had $T = 4$.

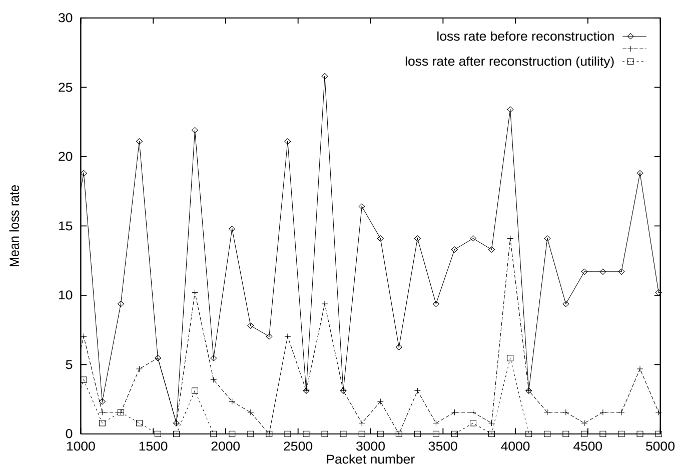

Figure 8: Evolutions of the loss rate and the utility (loss rate after reconstruction) over time, $f = f_3$

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 26%. 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 8 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 1 that 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. However, we have found in practice that the best subjective quality by far is obtained with function f_2, i.e. the utility function that 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 9 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. First, the quality increases dramatically as T goes from 1 to

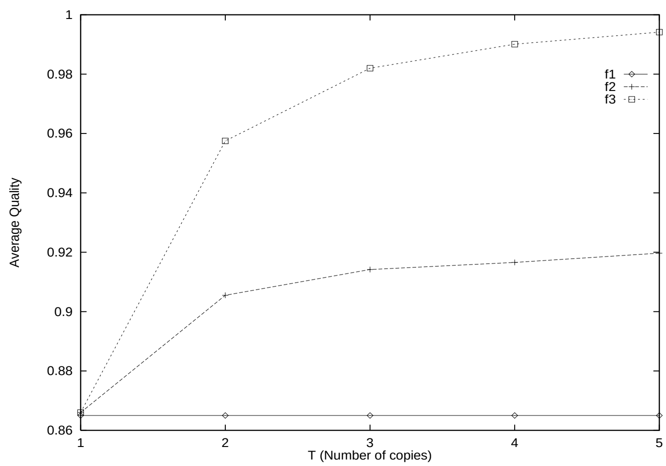

Figure 9: Perceived quality at the destination as a function fo the maximum delay T (or the number of redundant copies)

2. This indicates that adding just one piece of redundant information about packet $n - 1$ in packet n does make a big difference in quality. This is consistent with the subjective results shown in [14]. 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.

Sensitivity analysis

Suppose that f_3 is chosen by the application as being the appropriate utility function, i.e. the goal is to minimize the loss rate after reconstruction. Suppose now that, given the same network conditions (i.e. loss and available bandwidth), another function were chosen. How well would that function fare in terms of perceived loss rate at the destination? The question can be rephrased as: how sensitive is the optimal solution given by the joint rate/error control algorithm with respect to perturbations in the utility function? Note that it is in fact a special case of another, more general, question, namely: how sensitive are the optimal solutions to perturbations in the parameters of the problem (utility function, available rate R, maximum delay T, etc)?

We saw earlier that the perceived quality at the destination depends on T only for some utility functions, but, for a given T, depends significantly on the choice of f. Thus, sensitivity with respect to changes in T is different from sensitivity with respect to changes in f. Unfortunately, while it is easy to compute the optimal solution to our problem using the algorithm shown in the previous section, it is extremely hard to obtain any kind of general sensitivity analysis of the optimal solution. This result is not specific to our problem, but it is true of solutions of constrained optimization problems in general.

5 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 large multicast groups and to extend the work in this paper on source-based adaptive optimal rate/error control schemes to destination-based schemes. Another is to use our technique to solve similar problems in other, related 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. In distributed gaming applications [6], for example, the idea is 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 so as to maximize the quality perceived by the destinations[6]. The FEC scheme could 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.

References

[1] Aplio/Phone, http://www.aplio.com/

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

[3] E. W. Biersack, "Performance evaluation of FEC in ATM networks", *Proc. ACM Sigcomm '92*, pp. 248-257, Baltimore, MD, Aug. 1992.

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

[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] 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 sub-parts of the bridge, etc. Thus, the information related to the bridge can be sent with varying encoding rates, i.e. described with varying levels of detail.

[6] J-C. Bolot, S. Fosse-Parisis, “Adding voice to a distributed game in the Internet”, *Proc. IEEE Infocom ’98*, San Fransisco, CA, April 1998.

[7] D-M. Chiu, R. Jain, “Analysis of the increase/decrease algorithms for congestion avoidance in computer networks”, *Comp. Nets. and ISDN Systs.*, vol. 17, pp. 1-14, June 1989.

[8] J. Danskin, G. Davis, X. Song, “Fast lossy Internet image transmission”, *ACM Multimedia*, San Francisco, Nov. 1995.

[9] 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.

[10] 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.

[11] S. Floyd, K. Fall, “Promoting the use of end to end congestion control in the Internet”, technical report, 1998.

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

[13] M. Handley, “An examination of MBone performance”, USC/ISI Research Report ISI/RR-97-450, Aug. 1997.

[14] V. Hardman et al., “Reliable audio for use over the Internet”, *Proc. INET ’95*, Honolulu, HI, pp. 171-178, June 1995.

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

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

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

[18] 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.

[19] N. Merhav, M. Gutman, J. Ziv, “On the estimation of the order of a Markov chain”, *IEEE Trans. Info. Theory*, vol. 35, no. 5, Sept. 1989.

[20] S. B. Moon, J. Kurose, D. Towsley, “Packet audio playout delay adjustments: performance bounds and algorithms”, to appear in *ACM Multimedia Systems*.

[21] P.M. Morse. *Queues, Inventories, and Maintenance.* John Wiley, 1958.

[22] V. Paxson, “End-to-end Internet packet dynamics”, *Proc. ACM Sigcomm’97*, Cannes, France, Sept. 1997.

[23] C. Perkins et al., “RTP payload for redundant audio data”, RFC 2198, Sept. 1997.

[24] 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.

[25] J. Pulver, Voice on the Net, www.pulver.com.

[26] L. R. Rabiner, R. W. Schafer, *Digital Processing of Speech Signals*, Prentice Hall, 1978.

[27] R. Ramjee, J. Kurose, D. Towsley, “Adaptive playout mechanisms for packetized audio applications in wide-area networks”, *Proc. IEEE Infocom’94*, pp. 680-688, Toronto, June 1994.

[28] R. Rejaie, M. Handley, D. Estrin, “RAP: An end-to-end rate-based congestion control mechanism for realtime streams in the Internet”, ISI Research Report, 1998.

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

[30] J. Rosenberg, “G.729 error recovery for Internet telephony”, Columbia University report, spring 1997.

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

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

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

[34] N. Shacham, P. McKenney, “Packet recovery in high-speed networks using coding and buffer management”, *Proc. IEEE Infocom ’90*, San Fransisco, CA, pp. 124-131, May 1990.

[35] S. Shenker, “Fundamental design issues for the future Internet”, *IEEE JSAC*, Nov. 1995.

[36] 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.

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

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

[39] M. Yajnik, J. Kurose, D. Towsley, "Packet loss correlation in the MBone multicast network," *IEEE Global Internet Conf.*, London, UK, Nov. 1996.

[40] M. Yajnik, J. Kurose, D. Towsley, "Packet loss correlation in the MBone multicast network: Experimental measurements and Markov chain models", Technical Report UM-CS-95-115, University of Massachusetts, August 1995.

Appendix

Recall that we assume that K copies of a packet will be transmitted with the k-th copy transmitted at time $t_k = (i-1)T/(K-1), i = 1, \ldots, K$ (the time is normalized so that the first copy is transmitted at time $t_1 = 0$). Furthermore, we now focus on the loss process embedded at the transmission times, $t_1, \ldots, t_K$. Because they are evenly spaced, this can be described by a two-state *discrete-time* (this is unlike the continuous time approach taken in the body of the paper) Markov chain $\{X_i\}$ where $X_i = X_{t_i}, i = 1, \ldots$ with transition probability matrix

$$P = \begin{bmatrix} 1-p & p \\ q & 1-p \end{bmatrix}$$

where

$$\begin{aligned} p &= \mu_1(1-\exp(-(\mu_0+\mu_1)t)/(\mu_0+\mu_1), \\ q &= \mu_0(1-\exp(-(\mu_0+\mu_1)t)/(\mu_0+\mu_1). \end{aligned}$$

We will derive some interesting properties of the optimum solution $\mathbf{x}^* = (x_1^*, \ldots, x_K^*)$ which will permit us to solve the problem for the case $K = 2, 3, 4$, and to describe a conjecture for $K \geq 5$.

Suppose that the optimal solution $\mathbf{x}^*$ exhibits the following relation:

$$x_{k_1}^* \geq x_{k_2}^* \geq \cdots \geq x_{k_K}^*$$

for some ordering $\mathbf{k} = (k_1, \ldots, k_K)$. Note that this implies that

$$f(x_{k_1}^*) \geq f(x_{k_2}^*) \geq \cdots \geq f(x_{k_K}^*)$$

since f is increasing convex. We can think of the operation of the receiver as using the ordering $\mathbf{k}$ to determine the priority placed on the packet transmitted in the k-th position at time t_k. In other words, if packet k_l is not lost, it is used in the case that packets $k_1, \ldots, k_{l-1}$ are lost. This suggests a different statement of the problem, namely to determine the priority vector $\mathbf{k}^*$ and the vector of packet encoding rates $\mathbf{x}^*$ associated with this priority vector that solves the following problem

$$\begin{array}{lll} \text{Maximize} & \sum_{l=1}^{K} a_l(\mathbf{k}) f(x_{k_l}), & \\ \text{s.t.} & x_k \geq r_0, & k = 1, \ldots, K \\ & \sum_{k=1}^{K} x_i \leq R & \\ & \mathbf{k} \in \mathcal{K} & \end{array}$$

where a_l is the prob. that the l-th transmission is the first successful transmission given that the order of importance is $k_1, k_2, \ldots, k_K$, i.e.

$$a_l(\mathbf{k}) = P(\sum_{j=1}^{l-1} X_{t_{k_j}} = 0,\ X_{t_{k_l}} = 1) \tag{2}$$

and $\mathcal{K}$ is the set of all possible priority vectors (permutations of $(1, 2, \ldots, K)$). Note that a_l can be computed simply as follows. Let $\hat{k}_1, \ldots, \hat{k}_{i_0}, \ldots, \hat{k}_l$ be an ordered version of $k_1, \ldots, k_l$ in increasing order where $\hat{k}_{i_0} = k_l$. Then

$$a_l(k) = \prod_{j=1,\ldots,l; j \neq i_0, i_0+1} p_{0,0}^{(\hat{k}_j - \hat{k}_{j-1})} p_{0,1}^{((\hat{k}_{i_0} - \hat{k}_{i_0-1})} p_{1,0}^{((\hat{k}_{i_0+1} - \hat{k}_{i_0})}$$

where $p_{i,j}^{(m)}$ is the m step transition probability of going to state j from i.

Suppose we consider the simpler problem of determining the optimal $\mathbf{x}^*(\mathbf{k})$ associated with a specific priority vector $\mathbf{k} \in \mathcal{K}$.

$$
\begin{array}{lll}
\text{Maximize} & \sum_{l=1}^{K} a_l(\mathbf{k}) f(x_{k_l}), & \\
\text{s.t.} & x_k \geq r_0, & k = 1, \ldots, K \\
 & \sum_{k=1}^{K} x_i \leq R &
\end{array}
$$

The solution to this problem can be shown to satisfy the following relations

$$
\begin{aligned}
a_k g(x_k^*(\mathbf{k})) &= \alpha, x_k^*(\mathbf{k}) > r_0, \quad k = 1, \ldots K, \\
a_k g(x_k^*(\mathbf{k})) &\leq \alpha, x_k^*(\mathbf{k}) = r_0, \quad k = 1, \ldots, K
\end{aligned}
$$

where α is a Lagrange multiplier chosen so that $\sum_{k=1}^{K} x_k = R$. As a consequence of the definition of a_i, we have the following monotonicity property,

$$
a_{k_1}(\mathbf{k}) \geq a_{k_2}(\mathbf{k}) \geq \cdots \geq a_{k_K}(\mathbf{k})
$$

which implies that

$$
x_{k_1}^*(\mathbf{k}) \geq \cdots \geq x_{k_K}^*(\mathbf{k}).
$$

Hence, we are faced with a combinatorial problem where we must choose the appropriate priority vector $\mathbf{k} \in \mathcal{K}$.

We now state two properties that allow us simplify this combinatorial search. The first of these properties relies on the following definition.

Definition 5.1 *Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^K$. Vector $\mathbf{x}$ is said to be larger than $\mathbf{y}$, written as $\mathbf{x} \geq \mathbf{y}$ if*

$$
\sum_{i=1}^{k} x_i \geq \sum_{i=1}^{k} y_i \qquad k = 1, \ldots K.
$$

Property 1. Take $\mathbf{k}, \mathbf{j} \in \mathcal{K}$. Form the vectors $\mathbf{a}(\mathbf{k}) = (a_{k_1}(\mathbf{k}), \ldots, a_{k_K}(\mathbf{k}))$ and $\mathbf{a}(\mathbf{j}) = (a_{j_1}(\mathbf{j}), \ldots, a_{j_K}(\mathbf{k}))$. If $\mathbf{a}(\mathbf{k}) \geq \mathbf{a}(\mathbf{j})$, then

$$
\sum_{l=1}^{K} a_{k_l}(\mathbf{k}) f(x_{k_l}(\mathbf{k})) \geq \sum_{l=1}^{K} a_{j_l}(\mathbf{j}) f(x_{j_l}(\mathbf{j})).
$$

Property 2. For any priority vector $\mathbf{j} \in \mathcal{K}$ such that either $j_1 \neq 1$ or $j_2 \neq K$, there exists an priority vector $\mathbf{k}$ with $k_1 = 1$ and $k_2 = K$ such that

$$
\sum_{l=1}^{K} a_{k_l}(\mathbf{k}) f(x_{k_l}(\mathbf{k})) \geq \sum_{l=1}^{K} a_{j_l}(\mathbf{j}) f(x_{j_l}(\mathbf{j})).
$$

This property follows from the fact that $p_{1,0}(t)$ is an increasing function of t.

Consequences of these properties are that the optimal priority vectors for the case $K = 2, 3, 4$ are $(1, 2)$, $(1, 3, 2)$, and $(1, 4, 2, 3)$ respectively. Of course, this is in agreement with the result obtained in the paper. Note that in the $K = 5$, we need only consider priority vectors $(1, 5, 3, 2, 4), (1, 5, 2, 4, 3)$.