RATE-DISTORTION OPTIMIZED SENDER-DRIVEN STREAMING OVER BEST-EFFORT NETWORKS

P.A. Chou
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399 USA

Z. Miao
Signal and Image Processing Institute
University of Southern California
Los Angeles, CA, 90089-2564 USA

Abstract - This paper addresses the problem of streaming packetized media over a lossy packet network, in a rate-distortion optimized way. Out of all the packets that a sender could transmit at a given transmission opportunity, we show how the sender should compute which packets, if any, to transmit in order to meet an average rate constraint while minimizing the average end-to-end distortion. Experimental results show that our system has steady-state gains of 3–7 dB or more over systems that are not rate-distortion optimized.

INTRODUCTION

In this paper, we introduce a framework for distortion-rate optimized streaming of packetized media, and apply it to the scenario of sender-driven transmission over a best-effort network with feedback. Rather than streaming the packetized media in a fixed sequence according to presentation time, the sender chooses a transmission policy for each unit of data that minimizes the expected end-to-end distortion of the entire presentation subject to a transmission rate constraint. The solution to this resource allocation problem is obtained by minimizing a Lagrangian, taking into account the data units' dependence relationships in addition to their different delivery deadlines and basic importances. This minimization results in a form of unequal loss protection, in which *sensitive* data units — data units whose delivery most affects the distortion — are transmitted or retransmitted in preference to less sensitive data units. Consequently the more sensitive data units are often pre-transmitted far in advance of their presentation times, while the less sensitive data units are transmitted later if at all. Furthermore the sensitivity of a data unit is not a static quantity, but varies in response to feedback from the receiver. If a data unit is acknowledged as received, then the data units on which it depends increase in sensitivity while the data units that depend on it decrease in sensitivity, and vice versa. The proper computation is embodied in an iterative descent algorithm we call the sensitivity adjustment (SA) algorithm.

To our knowledge, the most closely related contemporaneous work is that by Miao and Ortega [1], which develops a low-complexity heuristic algorithm for sender-driven scheduling of packet transmissions over a best-effort network. Zhou and Li [2] also develop similar heuristics. The most closely related rigorous work is that by Podolksy, McCanne, and Vetterli [3]. The present paper is a severely shortened version of [4].

PRELIMINARIES

In a streaming media system, the encoded data are packetized into *data units* and are stored in a file on a media server. Regardless of how many media objects (audio, video, etc.) there are in a multimedia presentation, and regardless of what algorithms are used for encoding and packetizing those media objects, all of the data units in the presentation have interdependencies, which be expressed by a directed acyclic graph as illustrated in Figure 1. Each node of the graph corresponds to a data unit, and each edge of the graph directed from data unit l' to data unit l implies that data unit l can be decoded only if data unit l' is first decoded.

Associated with each data unit l is a size B_l, a decoding time $t_{DTS,l}$, and an importance Δd_l. The size B_l is the size of the data unit in bytes. The decoding time $t_{DTS,l}$ is the time at which the decoder is scheduled to extract the data unit from its input buffer and decode it (the decoder timestamp in MPEG terminology). Thus $t_{DTS,l}$ is the delivery deadline by which data unit l must arrive at the client, or be too late to be used. Packets containing a data unit that arrive after the data unit's delivery deadline are discarded. The importance Δd_l is the amount by which the distortion at the receiver will *decrease* if the data unit arrives on time at the receiver and is decoded.

Also associated with each data unit l is a set of $N = N_l$ transmission opportunities $t_{0,l}, t_{1,l}, \ldots, t_{N-1,l}$ prior to $t_{DTS,l}$ at which the data unit may be put into a packet and transmitted. Often this set of transmission opportunities is a single time $t_{0,l}$ (such as a "send time") prior to the delivery deadline, but in general we assume it is a finite set of times $t_{0,l}, t_{1,l}, \ldots, t_{N-1,l}$, such as the set of times at $T = 50$ ms intervals within a window prior to the delivery deadline.

We model the network as an independent time-invariant packet erasure channel with random delays. That means that if the sender inserts a packet into the network at time t, then the packet is lost with some probability, say ϵ_F, independent of t. However, if the packet is not lost, then it arrives at the receiver at sender time t', where the forward trip time $FTT = t' - t$ is randomly drawn according to probability density p_F. Each packet is lost or delayed independently of the other packets. In practice we allow p_F to adapt to the channel state (e.g., "congested" or "not congested") over time, by estimating the parameters of p_F (equivalent to mean and variance) on line using an exponentially weighted moving average as in TCP. For convenience, we combine the packet loss probability and the packet delay density into a single probability measure,

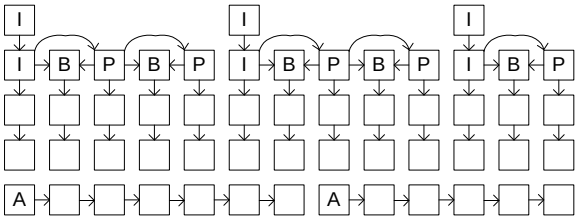

Figure 1: Typical directed acyclic dependency graph for video and audio data units.

by assigning $FTT = \infty$ in the event that the packet is lost. Thus $P\{FTT > \tau\}$ is the probability that a packet sent at time s is not received by time $s + \tau$, whether lost or simply delayed. We assume that the back channel, if available, can be similarly characterized by a probability measure on the backward trip time BTT. The round trip time $RTT = FTT + BTT$ is by definition the sum of forward and backward trip times. In our experiments we use as the density p_F the shifted Gamma distribution with parameters (n_F, α_F) and right shift κ_F.

R-D OPTIMIZATION USING SENSITIVITY ADJUSTMENT

We assume that each data unit l can be transmitted with a different policy π_l selected from a family of policies Π. The family Π is determined by the scenario under consideration. For example, in a forward error correction scenario, Π may correspond to a family of erasure codes having parameters (n, k). Similarly, in a retransmission-based scenario, Π may correspond to family of transmission schedules according to which a data unit is transmitted until an acknowledgement is received. The latter scenario will be investigated in the next section.

Suppose there are L data units in the multimedia session. Let π_l be the transmission policy for data unit $l \in \{1, \ldots, L\}$ and let $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_L)$ be the vector of transmission policies for all L data units. Any given policy vector $\boldsymbol{\pi}$ induces an expected distortion $D(\boldsymbol{\pi})$ and an expected transmission rate $R(\boldsymbol{\pi})$ for the multimedia session. We seek the policy vector $\boldsymbol{\pi}$ that minimizes the Lagrangian $D(\boldsymbol{\pi}) + \lambda R(\boldsymbol{\pi})$ for some Lagrange multiplier $\lambda > 0$, and thus achieves a point on the lower convex hull of the set of all achievable distortion-rate pairs.

The expected transmission rate $R(\boldsymbol{\pi})$ is the sum of the expected transmission rates for each data unit $l \in \{1, \ldots, L\}$:

$$R(\boldsymbol{\pi}) = \sum_l B_l \rho(\pi_l), \tag{1}$$

where B_l is the number of bytes in data unit l and $\rho(\pi_l)$ is the *expected cost* per byte, or the expected number of transmitted bytes per source byte under policy π_l. The expected distortion $D(\boldsymbol{\pi})$ is somewhat more complicated to express, but it can be expressed in terms of the *expected error*, or the probability $\epsilon(\pi_l)$ for $l \in \{1, \ldots, L\}$ that data unit l does not arrive at the receiver on time under policy π_l. Specifically, let I_l be the indicator random variable that is 1 if data unit l arrives at the receiver on time, and is 0 otherwise. Then $\prod_{l' \preceq l} I_{l'}$ is 1 if data unit l is decodable by the receiver on time, and is 0 otherwise. Here, $l' \preceq l$ means that l depends directly or indirectly on l. If data unit l is decodable by the receiver on time, then the reconstruction error is reduced by the quantity Δd_l; otherwise the reconstruction error is not reduced. Hence the total reduction in reconstruction error for the presentation is $\sum_l \Delta d_l \prod_{l' \preceq l} I_{l'}$. Subtracting this quantity from the reconstruction error for the presentation if no data units are received, and taking expectations, we have for the expected distortion

$$D(\boldsymbol{\pi}) = D_0 - \sum_l \Delta D_l \prod_{l' \preceq l} (1 - \epsilon(\pi_{l'})), \tag{2}$$

where D_0 is the expected reconstruction error for the presentation if no data units are received and ΔD_l is the expected reduction in reconstruction error if data unit l is decoded on time. Here we have used the assumption that the data packet transmission processes are independent, and are independent of the source process, in order to factor the expectation.

With expressions (1) and (2) for the expected transmission rate and expected distortion for any given policy vector now in hand, we are now able to find the policy vector $\boldsymbol{\pi}$ that minimizes the expected Lagrangian

$$J(\boldsymbol{\pi}) = D(\boldsymbol{\pi}) + \lambda R(\boldsymbol{\pi}) = D_0 + \sum_l \left[\Delta D_l \left(- \prod_{l' \preceq l} (1 - \epsilon(\pi_{l'})) \right) + \lambda B_l \rho(\pi_l) \right]. \quad (3)$$

However, this minimization is complicated by the fact that the terms involving π_l are not independent. We employ an iterative descent algorithm, called the sensitivity adjustment (SA) algorithm, in which we minimize the objective function $J(\pi_1, \ldots, \pi_L)$ in (3) one component at a time while keeping the other variables constant, until convergence. Let $\boldsymbol{\pi}^{(0)}$ be any initial policy vector and let $\boldsymbol{\pi}^{(n)} = (\pi_1^{(n)}, \ldots, \pi_L^{(n)})$ be determined for $n = 1, 2, \ldots$, as follows. Select one component $l_n \in \{1, \ldots, L\}$ to optimize at step n, e.g., $l_n = (n \bmod L)$. Then for $l \neq l_n$, let $\pi_l^{(n)} = \pi_l^{(n-1)}$, while for $l = l_n$, let $\pi_l^{(n)} = \arg\min_{\pi_l} J(\pi_1^{(n)}, \ldots, \pi_{l-1}^{(n)}, \pi_l, \pi_{l+1}^{(n)}, \ldots, \pi_L^{(n)})$, or

$$\pi_l^{(n)} = \arg\min_{\pi_l} S_l^{(n)} \epsilon(\pi_l) + \lambda B_l \rho(\pi_l), \quad (4)$$

where (4) follows from (3) with $S_l^{(n)} = \sum_{l' \succeq l} \Delta D_{l'} \prod_{l'' \preceq l' : l'' \neq l} (1 - \epsilon(\pi_{l''}^{(n)}))$. The factor S_l can be regarded as the *sensitivity* to losing data unit l, i.e., the amount by which the expected distortion will increase if data unit l cannot be recovered at the receiver, given the current transmission policies for the other data units.

The minimization (4) is now simple, since each data unit l can be considered in isolation. Indeed the optimal transmission policy $\pi_l \in \Pi$ for data unit l minimizes the "per data unit" Lagrangian $\epsilon(\pi_l) + \lambda' \rho(\pi_l)$, where $\lambda' = \lambda B_l / S_l^{(n)}$. Thus to minimize (4) for any l and λ', it suffices to know the lower convex hull of the function $\epsilon(\rho) = \min_{\pi \in \Pi} \{\epsilon(\pi) : \rho(\pi) \leq \rho\}$, which we call the *error-cost* function. The error-cost function can be considered as a normalized distortion-rate function pertaining to the transmission of a single dimensionless data unit, and depends only on the transmission scenario and the channel characteristics. The next section investigates the error-cost function for the scenario of sender-driven transmission over a best-effort network with feedback.

SENDER-DRIVEN TRANSMISSION OF A SINGLE DATA UNIT

First consider the scenario without feedback. Let $t_0, t_1, \ldots, t_{N-1}$ be N discrete transmission opportunities for a data unit and let t_{DTS} be its delivery deadline. Repeatedly transmitting the data unit at all N opportunities results in a small expected

error (equal to $\prod_i P\{FTT > t_{DTS} - t_i\}$) but a large expected cost (equal to N). On the other hand transmitting the data unit at none of the N opportunities results in a large expected error (equal to 1) but a small expected cost (equal to 0). Intermediate expected errors and costs can also be achieved and easily computed for any fixed transmission pattern. For example, suppose $a_0, a_1, \ldots, a_{N-1}$ represents a transmission pattern where $a_i = 1$ if a data packet is transmitted at time t_i and $a_i = 0$ otherwise. Then the expected error is equal to $\prod_{i:a_i=1} P\{FTT > t_{DTS} - t_i\}$ while the expected cost is equal to $\sum_{i:a_i=1} 1$ transmitted bytes per source byte.

Now consider the scenario with feedback. Suppose the receiver sends an acknowledgement packet back to the sender the instant that it receives a data packet, and that the sender truncates its transmission pattern upon receipt of the acknowledgement packet. Then although the expected error remains the same, the expected cost is reduced to $\sum_{i:a_i=1}(\prod_{j<i:a_j=1} P\{RTT > t_i - t_j\})$. To see this, consider that each term in parentheses is the probability that none of the previously transmitted packets is acknowledged by time t_i, which in turn is the expected value of the indicator function of the event that a data packet is transmitted at time t_i.

Figure 2a shows on a log-linear scale the lower convex hull of the error-cost function for this scenario, with each vertex of the convex hull labeled by the sequence of actions $[a_0, a_1, \ldots, a_7]$ for the optimal policy π^*_λ corresponding to the Lagrange multiplier λ for that vertex. Of course the curve is convex on a linear scale.

EXPERIMENTAL RESULTS

Here we investigate the distortion-rate performance for streaming one minute of packetized audio content using different methods. The audio content, the first minute of Sarah McLachlan's *Building a Mystery*, is compressed using a scalable version of the Windows Media Audio codec. The codec produces a group of twelve 500-byte data units every 0.75 seconds for a maximum data rate of 64 Kbps. All twelve data units in the mth group receive the same decoding timestamp, equal to $0.75m$.

We compare several streaming systems. System 1 is has rate control but not error control. Data units are transmitted at most once, in group order. The number of data units transmitted in each group is proportional to the transmission rate. System 2 is similar to commercial systems. Error control is provided by retransmissions, which may occupy up to 20% of the channel bandwidth (equal to the packet loss probability). Data units for which the server receives negative acknowledgements (NAKs) from the client are queued, and are retransmitted from the queue on a space-available basis. The remaining 80% or more of the channel bandwidth is used for first-time transmission of data units in the same manner as in System 1. In our simulation the client is omniscient: for each packet that is lost, the client sends a NAK at precisely the moment that the packet would have arrived at the client if it had not been lost. This provides an upper bound on the performance of any real client. System 3 is rate-distortion optimized as described in the previous sections. Unlike System 2, no NAKs are available; only ACKs are sent back to the server upon receipt of a packet by the client. The Lagrange multiplier λ is fixed for the entire presentation. System 4 is the

same as System 3 with the addition of rate control. In this case we fix a bandwidth limit for each group (instead of fixing λ as in System 3). All of the systems use the same playback delay (420 ms) and the same channel parameters.

As shown in Figure 2b, System 1 saturates in performance as the transmission rate increases. This is because in the absense of error control, base layer packets are being lost 20% of the time, limiting overall performance, regardless of the transmission rate. System 2 outperforms System 1 by three or more dB, while System 3 outperforms System 2 by an addition four or more dB, for a total gain up to seven or more dB over System 1. System 4 pays very little penalty (a fraction of a dB) for imposing a fixed rate constraint on the transmission. Thus, it is clear that the rate-distortion optimized systems obtain superior performance by using the available bandwidth in the most cost-effective way.

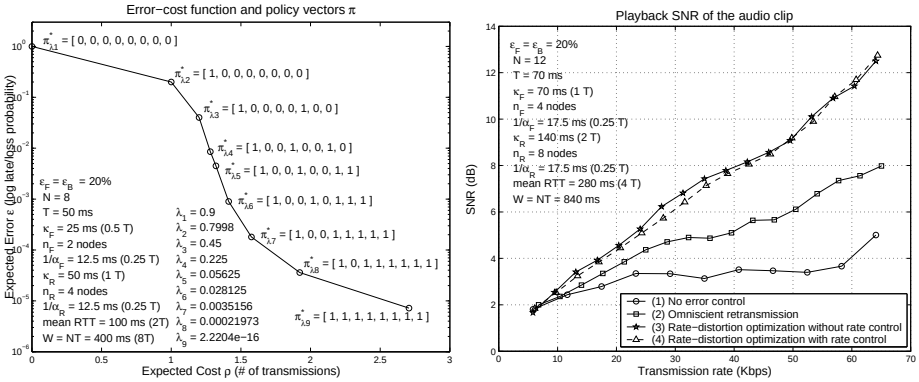

Figure 2: (a) Error-cost function. (b) Distortion-rate performances.

References

[1] Z. Miao and A. Ortega. Optimal scheduling for streaming of scalable media. In *Proc. Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, November 2000.

[2] J. Zhou and J. Li. Scalable audio streaming over the Internet with network-aware rate-distortion optimization. In *Proc. Int'l Conf. Image Processing*, Thessaloniki, Greece, October 2001. IEEE. Submitted.

[3] M. Podolsky, S. McCanne, and M. Vetterli. Soft ARQ for layered streaming media. Technical Report UCB/CSD-98-1024, University of California, Computer Science Division, Berkeley, CA, November 1998.

[4] P. A. Chou and Z. Miao. Rate-distortion optimized streaming of packetized media. Technical Report MSR-TR-2001-35, Microsoft Research, Redmond, WA, February 2001.