RATE-DISTORTION OPTIMIZED VIDEO STREAMING WITH MULTIPLE DEADLINES

Mark Kalman, Prashant Ramanathan, Bernd Girod

Information Systems Laboratory, Department of Electrical Engineering
Stanford University
{mkalman, pramanat, bgirod}@Stanford.EDU

ABSTRACT

We extend a recently-proposed framework for the rate-distortion optimized transmission of packetized media. The original framework assumed that each packet has a single arrival deadline and that a packet is useless if it arrives after its deadline. In practice, however, packets may be associated with multiple deadlines. Examples include the case of compressed video that uses bi-directional prediction and the case of decoders that can recover from late packet arrivals through the accelerated retroactive decoding of the dependency chain. We extend the original framework to consider multiple deadlines. In our experimental results for the case of the accelerated retroactive decoding of late packets, the multiple-deadline formulation yields up to a 1.5 dB improvement in rate-distortion performance compared to the original, single-deadline formulation. The results indicate, furthermore, that accelerated retroactive decoding offers significant benefit only when coupled with a scheduler that considers multiple deadlines.

1. INTRODUCTION

Recent work has investigated how to stream packetized media over a lossy channel in a rate-distortion optimized way. Of particular interest is how to schedule packet transmissions in an optimal way when ARQ is used to combat packet losses.

In principle, a scheduler can generate a rate-distortion optimized sequence of packet transmissions by calculating the expected distortion and rate that will result from every possible transmission sequence, and then choosing the sequence that offers the best rate-distortion trade-off. In practice, as noted in [1], the first rigorous work on the topic, this exhaustive approach is too complex.

In the framework presented in [2], the authors introduce an iterative descent algorithm that simplifies the search for an optimal transmission sequence. In this framework, the scheduler chooses packets to transmit at regular intervals in time by optimizing a complete policy governing transmissions that will occur during a horizon of transmission times into the future. At each discrete transmission opportunity, these *transmission policies* are re-optimized to account for any feedback in the form of acknowledgments that has been received. The descent algorithm, which is run at each time step, works by iteratively optimizing the transmission policy of one packet at a time until it converges on an optimized overall policy for all the packets.

For simplicity, the framework in [2] assumes that each packet removes a fixed amount of distortion from the displayed sequence

when it is decoded. This assumption is not always accurate. For instance, in the case when the decoder uses error concealment, the amount of distortion removed when a packet is decoded strongly depends on which other packets are available. In general, a unique distortion may result from every combination of available and unavailable packets. The framework requires extensions, therefore, to allow the accurate calculation of transmission schedules when error concealment is used. In [3], the authors provide an extension of the framework for the case of simple, previous-frame error concealment.

Another assumption in the framework presented in [2] is that a packet is useless if it does not arrive by its single decoding deadline. In some cases, this is not accurate. For instance, consider the case of bi-directional prediction with a sequence of frames I-B-B-B-P. If the P-frame does not arrive by the first B frame's deadline, it may still be useful if it arrives in time to decode the subsequent B frame. Another example is the case when the decoder allows the accelerated retroactive decoding of the dependency chain when a packet arrives late [4]. In the above sequence, for example, if the I-frame arrives after its own decoding deadline, but before the P-frame's deadline, the P-frame may still be decoded.

In this paper we extend the framework in [2] to take into consideration multiple packet deadlines. Our formulation, by considering the probability and resulting distortion of all possible decoding outcomes resulting from a transmission sequence, also allows the accurate calculation of optimized transmission schedules for video with simple error concealment. We restrict the error concealment to simple, previous-frame substitution as in [3] to limit the set of unique decoding outcomes to a tractable size

We begin in Sec. 2 with a brief review of the rate-distortion optimized streaming framework in [2]. In Sec. 3, we show our new formulation, which is directly related to the light field streaming scheme we present in [5]. In Sec. 4, we present experimental results which show the scheduling performance improvements attained with our multiple-deadline formulation for the case of the accelerated retroactive decoding of late packets.

2. BACKGROUND

In [2], a rate-distortion optimized framework for the streaming of media over a lossy packet network is presented. This framework assumes a compressed media representation that has been assembled into packets or data units. Associated with each data unit is the data unit size, for instance, in bytes, and the deadline by which the data unit must arrive in order for it to be useful for display. There is also a notion of distortion that the user experiences when only a subset of data units is available for decoding and playout.

In order to estimate the distortion in a computationally efficient manner, the authors assume that a single distortion reduction value is associated with each data unit. The distortion reduction value is the amount of distortion that is removed from the reconstructed stream if a particular data unit arrives on time, and all the ancestor data units that it depends upon also arrive on time. These distortion reductions are assumed to be additive, and the overall distortion is computed by using an acyclic directed graph that describes the inter-dependencies among the data units in the media presentation.

A transmission policy π_l is associated with each data unit l. In the case when the R-D framework is used to generate transmission schedules, a policy dictates when a packet will be transmitted in the succession of transmission intervals. For each given transmission policy π_l, there is an associated expected number of transmissions $\rho(\pi_l)$ and error probability $\epsilon(\pi_l)$, the probability that the data unit does not arrive by its deadline.

The goal is to determine the optimal transmission policies for all data units $\boldsymbol{\pi} = [\pi_1 \pi_2 \cdots \pi_N]$, given the deadlines, distortion reductions, network packet loss and delay statistics, feedback, and the transmission history. The policy that minimizes the overall rate-distortion Lagrangian cost,

$$J(\boldsymbol{\pi}) = D(\boldsymbol{\pi}) + \lambda R(\boldsymbol{\pi}), \tag{1}$$

where λ controls the trade-off between rate and distortion, is selected as the optimal policy.

The expected rate $R(\boldsymbol{\pi})$ depends upon the data unit sizes B_l, and the per-unit expected number of transmissions $\rho(\pi_l)$. The expected distortion $D(\boldsymbol{\pi})$ depends upon the error probabilities $\epsilon(\pi_l)$, the distortion reduction values, and the inter-dependency graph for the data units.

With a large number of data units, it is difficult to exactly solve the minimization problem in (1). A reasonable approximate solution is to begin with an initial set of transmission policies, and iteratively find the optimal policy for one data unit at a time, iterating until the overall solution converges. The cost function for data unit l is given by the equation

$$J_l(\pi_l) = \epsilon(\pi_l) + \lambda' \rho(\pi_l) \tag{2}$$

where $\epsilon(\pi_l)$ is the error probability and $\rho(\pi_l)$ is the per-unit cost as before. It can be shown that $\lambda' = \frac{\lambda B_l}{S_l}$, incorporating the rate-distortion trade-off multiplier λ from (1), the data unit size B_l, and S_l, the sensitivity of the overall distortion to not having received data unit l by its deadline. The sensitivity term represents the relative importance of a particular data unit.

3. PROPOSED FORMULATION

In the following subsections we describe how to extend the rate-distortion optimized streaming framework to allow multiple deadlines, and to take into consideration the effects of error concealment that substitutes a missing frame with the most recent decodable frame.

3.1. Multiple Deadlines

In the streaming framework of [2], each data unit has one arrival deadline, one error probability, and one sensitivity. We associate multiple deadlines with a data unit, one deadline for every frame in which the data unit may be used. For example, using the error concealment and retroactive decoding strategies that we consider in this paper, a data unit that contains an I-frame has a deadline associated with the I-frame's display time and deadlines associated with each subsequent frame that uses the I-frame for prediction. The data unit also has deadlines associated with the display times of frames in GOPs that follow, since the I-frame may be used as a substitute if frames are lost.

The introduction of multiple deadlines results in changes to expressions for the overall distortion, and changes to the policy minimization algorithm. For each data unit, there is no longer a single error probability, but an entire set $\{\epsilon(\pi_l, t_i)\}$, one for each of the playout deadlines, t_i, associated with that data unit.

We use the same descent algorithm as in the existing framework. The algorithm iteratively minimizes the overall cost with respect to one data unit's transmission policy at a time. The cost function with respect to the transmission policy of a data unit becomes

$$J_l(\pi_l) = \rho(\pi_l) + \sum_{i \in \mathcal{W}} \nu_{t_i} \epsilon(\pi_l, t_i) \tag{3}$$

where $\mathcal{W}$ is the transmission window, the set of frames whose data units are eligible for transmission, i is the frame index, and t_i is the decoding deadline for frame i. Thus, to determine the cost for a data unit l, we consider the deadlines of all frames that may require l and are in the transmission window. The quantity $\epsilon(\pi_l, t_i)$ is the probability that data unit l does not arrive by deadline t_i. The expected number of transmissions $\rho(\pi_l)$ does not depend on the deadlines. The quantity ν_{t_i} is given by $\nu_{t_i} = \frac{S_{l,t_i}}{\lambda B_l}$, analogous to the reciprocal of λ' in (2). Note that the sensitivity term S_{l,t_i} is also indexed by deadline. It is the sensitivity of the overall distortion to the arrival of data unit l by deadline t_i.

The cost (3) is computed for each policy, and the one with the lowest cost is selected as the optimal policy. Since the policy length tends to determine the complexity of the algorithm, using multiple deadlines does not greatly impact the computational complexity of the algorithm.

3.2. State-Dependent Distortion

Each frame in a video sequence is associated with a set of packetized data units that may be used to generate a reconstructed image for display. For f_i, the i-th frame, $i = 1, 2, \ldots$, of a video sequence, let $\mathcal{L}_i$ be the set of L data units that may be used either to directly decode the frame or to reconstruct an error-concealed version of it. There are 2^L possible arrivals and non-arrivals for this set of data units. We consider each such combination to be a state s that belong to set of 2^L possible packet availability states $\mathcal{P}_i$.

For every $s \in \mathcal{P}_i$, we can compute the reconstruction distortion that would result, so that we have a set of 2^L distortion values, one corresponding to each state. We can also calculate the probability that each s occurs, given a transmission sequence and channel statistics. The distortion values and their probabilities can then be used to compute the expected distortion for a transmission policy. Because the number of states is exponential in the number of data units, this state-based approach can quickly become intractable. In the case of previous-frame substitution error concealment, however, subsets of states that lead to the same reconstruction outcome can be grouped together, reducing the number of outcomes that must be considered to a tractable size.

In the following, we formulate the state-based approach assuming a two-layered, SNR-scalable encoding in which GOPs are

structured I-P-P-P ... and each frame consists of a base layer and an enhancement layer. The enhancement layer of a P-frame is dependent both on its base layer and on the enhancement layer of the previous frame. The enhancement layer of an I-frame is only dependent on its base layer.

Assuming this encoding and assuming error concealment that substitutes undecodable frames with the nearest previous frame that has been correctly decoded (using the enhancement layer if it is available), the i-th frame of the sequence can be shown with $2i + 1$ possible display outcomes. Any previous base or enhanced frame may be shown, or in the case when no previous frame has been decoded, no frame is shown.

Let s' belong to the set of $2i + 1$ possible display outcomes, $\mathcal{P}'_i$. Let $D(s', f_i)$ be the distortion that results if frame f_i is shown with outcome s'. At decoding time, the display outcome that occurs depends on which packets are available and which ones are unavailable. We can find the probability of a particular display outcome by recognizing the sets of available and unavailable packets that will lead to the outcome.

For any outcome s', we can define three sets of packets which govern whether the state will occur. It is required that a certain set $\mathcal{A}_{s'}$ of data units is available, a certain set $\mathcal{U}_{s'}$ of data units is not available, and there may also be a set $\mathcal{M}_{s'}$ of data units in which at least one element must be missing (unavailable). For example, for the outcome in which a P-frame is substituted with the previous frame's base layer, $\mathcal{A}_{s'}$ consists of all the base layer packets in the GOP up to and including the previous frame's base layer, $\mathcal{U}_{s'}$ contains only the P-frame that must be substituted, and $\mathcal{M}_{s'}$ consists of all the enhancement layer data units of the GOP up to and including the previous frame.

We can now calculate, for a given policy $\boldsymbol{\pi}$, the expected distortion for the frames in the transmission window

$$D(\boldsymbol{\pi}) = \sum_{i \in \mathcal{W}} \left[\sum_{s' \in \mathcal{P}'_i} D(s', f_i) \Pr\{s'\} \right] \quad (4)$$

where the probability of a display outcome $\Pr\{s'\}$ is given by

$$\Pr\{s'\} = \begin{cases} \prod_{l \in \mathcal{A}_{s'}} (1 - \epsilon(\pi_l, t_i)) \prod_{l \in \mathcal{U}_{s'}} \epsilon(\pi_l, t_i) \\ \quad \cdot \left(1 - \prod_{l \in \mathcal{M}} (1 - \epsilon(\pi_l, t_i)) \right) & \text{if } \mathcal{M} \neq \emptyset, \\ \\ \prod_{l \in \mathcal{A}_{s'}} (1 - \epsilon(\pi_l, t_i)) \prod_{l \in \mathcal{U}_{s'}} \epsilon(\pi_l, t_i) & \text{otherwise.} \end{cases} \quad (5)$$

As in the existing framework the expected rate is given by

$$R(\boldsymbol{\pi}) = \sum_l B_l \rho_l(\pi_l). \quad (6)$$

In order to minimize the overall Lagrangian cost (1) as before, we iteratively minimize the Lagrangian cost (3) for each data unit. From (4) we can derive the expression for the sensitivity of the distortion to the arrival of a data unit l by deadline t_i

$$S_{l,t_i} = \begin{cases} \sum_{\substack{s' \in \mathcal{P}'_i \\ s': l \in \{\mathcal{U}_{s'} \cup \mathcal{M}_{s'}\}}} S'_{s',l,t_i} - \sum_{\substack{s' \in \mathcal{P}'_i \\ s': l \in \mathcal{A}_{s'}}} S'_{s',l,t_i} & \text{if } l \in \mathcal{L}_i, \\ 0 & \text{otherwise,} \end{cases} \quad (7)$$

where

$$S'_{s',l,t_i} = D(s', f_i) \cdot$$
$$\begin{cases} \prod_{\substack{l' \in \mathcal{A}_{s'} \\ l' \neq l}} (1 - \epsilon(\pi_{l'}, t_i)) \prod_{\substack{l' \in \mathcal{U}_{s'} \\ l' \neq l}} \epsilon(\pi_{l'}, t_i) \\ \quad \cdot \left(1 - \prod_{\substack{l' \in \mathcal{M}_{s'} \\ l' \neq l}} (1 - \epsilon(\pi_{l'}, t_i)) \right) & \text{if } \mathcal{M} \neq \emptyset, \\ \\ \prod_{\substack{l' \in \mathcal{A}_{s'} \\ l' \neq l}} (1 - \epsilon(\pi_{l'}, t_i)) \prod_{\substack{l' \in \mathcal{U}_{s'} \\ l' \neq l}} \epsilon(\pi_{l'}, t_i) \\ \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \text{otherwise.} \end{cases}$$
$$(8)$$

4. EXPERIMENTAL RESULTS

In this section we show simulation results that demonstrate the rate-distortion improvements that can be achieved, in the case of accelerated retroactive decoding, by incorporating multiple deadlines. To demonstrate the effectiveness of our scheme we also include simulation results for a non-optimal heuristic scheme, as a basis for comparison.

4.1. Simulations

Our results are for the *Foreman* and *Mother-Daughter* video sequences encoded with a 2-layer, SNR scalable H.263+ at 10 fps, with a GOP length of 10. Each layer of each frame is placed into an individual packet. We model the packet network as in [2], with delay and loss events statistically independent from transmission to transmission. Packets are randomly delayed both in the forward and reverse directions according to a shifted-Γ distribution with shift $\kappa = 10$ ms, mean $\mu = 50$ ms, and standard deviation 23 ms. The packet loss probability in both directions is 0.20.

We assume that our transmission scheduler has perfect knowledge of the channel statistics. The scheduler has a transmission interval of 100 ms. It uses a fixed-size transmission window positioned such that a frame's data units become eligible for transmission 400 ms before the frame's arrival deadline. Playout begins at the client 400 ms after the first transmission. The results we show are averaged over 100 random seeds.

4.2. Heuristic Scheduler for Comparison

As a basis for comparison, we include the performance of a non-optimizing heuristic scheme that uses prioritized ARQ. Data units are appended to a transmission queue when they enter the transmission window. When a transmitted data unit is not acknowledged by the 90% point of the round-trip time cdf, it is again appended to the transmission queue. Priority for transmission in decreasing order is: base layer re-transmissions, base layer transmissions, enhancement layer re-transmissions, and enhancement layer transmissions. To achieve a fixed transmission rate R_h, each time a data unit l is transmitted another data unit is not transmitted for B_l/R_h seconds. Data units in the transmission queue are discarded when the transmission time is within a mean forward-trip time of their deadlines.

4.3. Results

In Fig. 1 we see results for the *Foreman* sequence. First, we observe that the rate-distortion performance of the heuristic scheme

is far worse than that of the optimizing schedulers over most of the range of bit-rates. A major shortcoming of the heuristic is that it will always retransmits packets for which no acknowledgment is received. Since acknowledgments are lost 20% of the time, significant bandwidth is wasted. At the highest bit rates, the heuristic outperforms the optimizing schedulers by a small margin. This is because of the short transmission window length compared to the policy length. The framework develops transmission policies without taking into consideration the new data units that will enter the window at subsequent transmission times, thus making the scheduling of future transmissions suboptimal.

Of the optimizing schedulers, the system that does not use accelerated retroactive decoding exhibits the poorest performance. The system that uses retroactive decoding, but whose optimizing scheduler considers only a single deadline performs only slightly better. By allowing the scheduler to consider multiple deadlines, however, a system that uses retroactive decoding performs significantly better. In this case, we observe a rate-distortion improvement of up to 1.5 dB relative to the single-deadline scheduler.

In Fig. 2, we see similar results for the *Mother-Daughter* sequence. Here, the maximum performance gain due to multiple deadlines is 0.8 dB. The performance gain is less dramatic for this sequence because there is less motion.

For both sequences, the performance gains due to multiple deadlines collapse suddenly as bit-rates drop. The reason is that at lower bit rates the R-D optimization begins to exploit the temporal scalability of the encodings by dropping the later frames in the GOPs. Retroactive decoding becomes less beneficial because the latter parts of decoding dependency chains are not used.
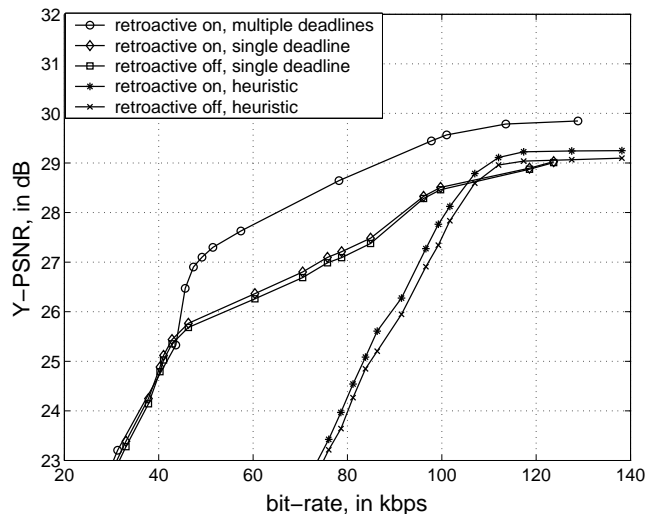

Fig. 1. Rate-Distortion performance of schedulers for the *Foreman* sequence. In the case of accelerated retroactive decoding, a performance gain of up to 1.5 dB is observed for the optimizing scheduler that considers multiple deadlines compared to the one that considers a single deadline

5. CONCLUSIONS

We have extended a scheme for the rate-distortion optimized streaming of packetized video in order to allow the incorporation
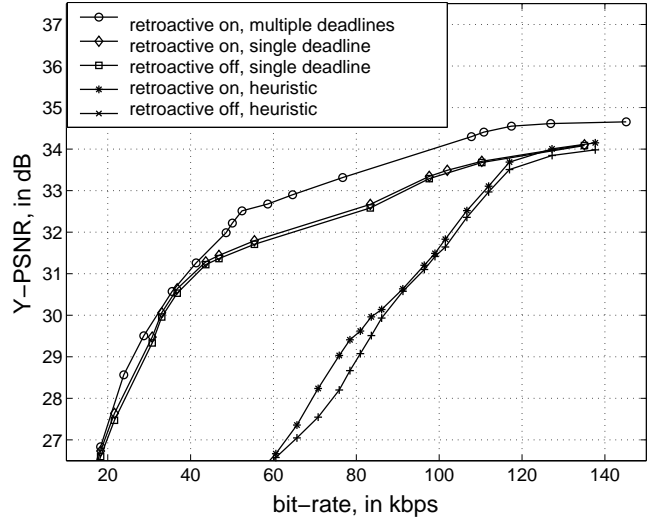

Fig. 2. Rate-Distortion performance of schedulers for the *Mother-Daughter* sequence. Again, in the case of accelerated retroactive decoding, improved performance is observed for the optimizing scheduler that considers multiple deadlines compared to the one that considers a single deadline

of multiple arrival deadlines for data units, and the incorporation of basic error concealment. We show that the incorporation of multiple deadlines can significantly improve streaming performance. In the case of accelerated retroactive decoding, we have demonstrated a 1.5 dB improvement in rate-distortion performance for an optimizing scheduler that considers multiple deadlines compared to one that does not. Furthermore, we have demonstrated that accelerated retroactive decoding only offers significant benefit when coupled with a scheduler that considers multiple deadlines.

6. REFERENCES

[1] Matthew Podolsky, Steven McCanne, and Martin Vetterli, "Soft ARQ for layered streaming media," Tech. Rep. UCB/CSD-98-1024, University of California, Computer Science Department, Berkeley, CA, November 1998.

[2] Philip A. Chou and Zhourong Miao, "Rate-distortion optimized streaming of packetized media," Tech. Rep. MSR-TR-2001-35, Microsoft Research, February 2001, (also submitted to IEEE Transactions on Multimedia).

[3] Jacob Chakareski and Bernd Girod, "Rate-distortion optimized packet scheduling and routing for media streaming with path diversity," in *Proceedings of the Data Compression Conference*, Snowbird, Utah, USA, April 2003, submitted.

[4] M. Ghanbari, "Postprocessing of late cells for packet video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 6, pp. 669–678, December 1996.

[5] Prashant Ramanathan, Mark Kalman, and Bernd Girod, "Rate distortion optimized streaming of compressed light fields," in *IEEE International Conference on ImageProcessing*, Barcelona, Spain, April 2003, submitted.