Rate-distortion Optimized Packet Scheduling and Routing for Media Streaming with Path Diversity

Jacob Chakareski* and Bernd Girod
Information Systems Laboratory, Department of Electrical Engineering,
Stanford University, Stanford, CA 94305
{cakarz,bgirod}@stanford.edu

Abstract

We consider diversity for media streaming in a rate-distortion optimization framework. A sender-driven transmission scenario is investigated, where diversity is achieved by using multiple transmission paths over the network. The proposed framework enables the sender to decide at every instant which packets, if any, to transmit and over which transmission paths in order to meet a rate constraint while minimizing the end-to-end distortion. Experimental results demonstrate the benefit of exploiting packet diversity in rate-distortion optimized sender-driven streaming of packetized media.

1 Introduction

Diversity techniques have been studied for many years in the context of wireless communication. They were introduced in order to exploit the large variability in terms of channel quality when multiple channels are considered for simultaneous transmission. A number of studies have shown that there is an analogous situation in Internet communication: in 30-80% of the cases there is an alternate path that performs significantly better than the default path between two hosts [1]. Performance is measured in terms of round-trip-time, loss rate and bandwidth. These studies have motivated the introduction of packet path diversity for video streaming in [2], where the author proposes to send complementary descriptions of a multiple description (MD) coder through two different Internet paths. The presented experimental results show the potential benefits of the proposed system.

Since then a number of studies have appeared that exploit the concept of packet path diversity in media communication. In [3] the authors employ path diversity in the context of video communication using unbalanced MD coding to accomodate the fact that different paths might have different bandwidth constraints. The unbalanced descriptions are created by adjusting the frame rate of a description sent

*On leave from Rice University, Houston, TX 77005.

over a particular path. In [4] the authors study image and video transmission in a multihop mobile radio network. It is shown that combining MD coding and multiple path transport in such a setting provides higher bandwidth and robustness to end-to-end connections. In [5] a framework for video transmission over the Internet is presented, based on path diversity and rate-distortion optimized reference picture selection. Here, based on feedback packet dependency is adapted to channel conditions in order to minimize the distortion at the receiving end, while taking advantage of path diversity. In [6] the performance of path diversity and multiple description coding in Content Delivery Networks (CDN) is studied. 20-40% reduction in distortion is reported over conventional CDNs for the network conditions and topologies under consideration. Finally, another related work is [7] where the authors consider rate-distortion optimized streaming over networks with DiffServ support.

In this paper, we present a general framework for rate-distortion optimized sender-driven streaming of packetized media over multiple network paths. Media packets are typically characterized by different deadlines, importances and interdependencies. Using this information and the proposed framework, the sender is able to transmit its media packets over multiple paths based on the feedback it receives in a rate-distortion optimized way, that is, minimizing the expected end-to-end distortion subject to a constraint on the expected overall transmission rate over the paths. Such a rate-distortion optimized transmission algorithm, or transmission policy, results in unequal error protection provided to different portions of the media stream. The core step of the optimization framework involves trading off the expected redundancy (the cost used to communicate the packet) for the probability that a single media packet will be communicated in error. The present work derives from the work in [8], which develops a framework for rate-distortion optimized streaming of packetized media over the Internet. By incorporating models for packet path diversity, error concealment and burst loss channels, we have substantially generalized the framework in [8].

2 Packet Loss Probabilities

In a streaming media system, the encoded data are packetized into *data units* and are stored in a file on a media server. All of the data units in the presentation have interdependencies, which can be expressed by a directed acyclic graph. 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. $t_{DTS,l}$ is the *delivery deadline* by which data unit l must arrive at the client, or be too late to be usefully decoded. Packets containing data units that arrive after the data units' delivery deadlines are discarded. The importance Δd_l is the amount by which the distortion at the client will *decrease* if the data unit arrives on time at the client and is decoded.

We model a network path between a sender and a receiver, typically a media server and a client, as a burst loss channel using a K-state discrete-time Markov model. The forward and the backward channel make state transitions independently of each other every T seconds, where the transitions are described by probability matrices $\mathcal{P}_{(F)}$ and $\mathcal{P}_{(B)}$, respectively. In each state the forward and the backward channel are characterized as an independent time-invariant packet erasure channel with random delay. Hence, they are completely specified with the probability of packet loss $\epsilon^k_{F/B}$ and the probability density of the transmission delay $p^k_{F/B}$, for $k = 1, \ldots, K$. This means that if the media server sends a packet on the forward channel at time t, given that the forward channel is in state k at t, then the packet is lost with probability ϵ^k_F. However, if the packet is not lost, then it arrives at the client at time t', where the forward trip time $FTT^k = t' - t$ is randomly drawn according to the probability density p^k_F. Therefore, we let $P\{FTT^k > \tau\} = \epsilon^k_F + (1 - \epsilon^k_F) \int_\tau^\infty p^k_F(t)dt$ denote the probability that a packet transmitted by the server at time t, given that the forward channel is in state k at t, does not arrive at the client application by time $t + \tau$, whether it is lost in the network or simply delayed by more than τ. Then similarly, $P\{BTT^k > \tau\} = \epsilon^k_B + (1 - \epsilon^k_B) \int_\tau^\infty p^k_B(t)dt$ denotes the probability that a packet transmitted by the client at time t, given that the backward channel is in state k at t, does not arrive at the server by time $t + \tau$, whether it is lost in the network or simply delayed by more than τ. Finally, we are interested in $P\{RTT^{kj} > \tau\}$, which is the probability that the server does not receive an acknowledgement by time $t + \tau$ for a packet transmitted at time t, given that the forward and the backward channel are respectively in states k and j, at t.

To derive $P\{RTT^{kj} > \tau\}$ assume first that the transmission on the forward channel occured immediately after the channel made a state transition. If $FTT^k \leq T$, the packet is received by the client before the backward channel makes the next state transition. Then $P\{RTT^{kj} > \tau | FTT^k \leq T\} = P\{FTT^k + BTT^j > \tau | FTT^k \leq T\}$ as the client sends an acknowledgement while the backward channel is still in the current state j. The probability of this event is $P\{FTT^k \leq T\}$. However, if $lT < FTT^k \leq (l+1)T$, for $l \geq 1$, then the state of the backward channel makes l transitions before the packet actually arrives at the client. The probability of this event is $P\{lT < FTT^k \leq (l+1)T\}$. Here the state on the backward channel when the acknowledgement is sent can be any of the K possible values. Hence we compute the desired quantity as the expected value over all of them, i.e., $P\{RTT^{kj} > \tau | lT < FTT^k \leq (l+1)T\} = \sum_{p=1}^K \mathcal{P}^{(l)}_{jp(B)} P\{FTT^k + BTT^p > \tau | lT < FTT^k \leq (l+1)T\}$. Note that $\mathcal{P}^{(l)}_{jp(B)}$ is the probability of making a transition from state j to state p in l transition intervals. These probabilities are obtained using matrix power, i.e., $\mathcal{P}^{(l)}_{(B)} = \mathcal{P}^l_{(B)}$. Finally, by averaging over all possible outcomes for FTT^k we write

$$
\begin{aligned}
P\{RTT^{kj} > \tau\} &= \sum_{l=0}^{\infty} \sum_{p=1}^{K} \mathcal{P}^{(l)}_{jp(B)} P\{lT < FTT^k \leq (l+1)T, FTT^k + BTT^p > \tau\} \\
&= \sum_{l=0}^{M-1} \sum_{p=1}^{K} \mathcal{P}^{(l)}_{jp(B)} P\{lT < FTT^k \leq (l+1)T, FTT^k + BTT^p > \tau\} \\
&+ \sum_{p=1}^{K} \mathcal{P}^{(M)}_{jp(B)} P\{MT < FTT^k \leq \tau, FTT^k + BTT^p > \tau\} + P\{FTT^k > \tau\}
\end{aligned}
$$

where the first equality follows from Bayes' rule, while the second one holds since $P\{lT < FTT^k \leq (l+1)T, FTT^k + BTT^p > \tau\} = P\{lT < FTT^k \leq (l+1)T\}$, for

$lT \geq \tau$. Finally, $M = \lfloor \tau/T \rfloor$ and $\mathcal{P}_{(B)}^{(0)} = \mathcal{I}$, the identity matrix.

3 Rate-distortion optimized policy selection

Suppose there are L data units in the media presentation. Let $\pi_l \in \Pi$ 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. Π is a family of policies defined precisely in the next section.

Any given policy vector $\boldsymbol{\pi}$ induces an expected distortion $D(\boldsymbol{\pi})$ and an expected transmission rate $R(\boldsymbol{\pi})$ for the media presentation. We seek the policy vector $\boldsymbol{\pi}$ that minimizes $D(\boldsymbol{\pi})$ subject to a constraint on $R(\boldsymbol{\pi})$. This can be achieved by minimizing the Lagrangian $D(\boldsymbol{\pi}) + \lambda R(\boldsymbol{\pi})$ for some Lagrange multiplier $\lambda > 0$, thus achieving a point on the lower convex hull of the set of all achievable distortion-rate pairs.

We now compute expressions for $R(\boldsymbol{\pi})$ and $D(\boldsymbol{\pi})$. The expected transmission rate $R(\boldsymbol{\pi})$ is the sum of the expected number of bytes transmitted for each data unit $l \in \{1, \ldots, L\}$, $R(\boldsymbol{\pi}) = \sum_l B_l \rho(\pi_l)$, where B_l is the number of bytes in data unit l and $\rho(\pi_l)$ is the expected number of transmitted bytes per source byte (under policy π_l), called the *expected cost*. The expected distortion $D(\boldsymbol{\pi})$ can be expressed in terms of the probability $\epsilon(\pi_l)$ that data unit l does not arrive at the receiver on time (under policy π_l), called the *expected error*. 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. Furthermore, let $N_c^{(l)} = \{1, \ldots, l\}$ be the set of data units that the receiver considers for error concealment in case data unit l is not decodable by the receiver on time. Then let $\Delta d_l^{(l_1)}$, for $l_1 \in N_c^{(l)}$, be the reduction in distortion if data unit l is not decodable and is concealed with a previous data unit l_1 that is received and decoded on time. Note that the decoder always prefers the most recent decodable unit from the concealment set, i.e., for $l_1, l_2 \in N_c^{(l)} : l_2 > l_1$ and both l_1, l_2 decodable on time the decoder always choses l_2 to conceal the missing data unit l. The product $\prod_{j \in A(l_1)} I_j$ is 1 if data unit l_1 is *decodable* by the receiver on time, and is 0 otherwise, where $A(l_1)$ is the set of ancestors of l_1, including l_1. Now given that l_1 is decodable, the product $\prod_{l_2 \in C(l,l_1)} \left(1 - \prod_{l_3 \in A(l_2) \setminus A(l_1)} I_{l_3}\right)$ is 1 if none of the data units $j \in N_C^{(l)} : j > l_1$ are decodable on time, and is 0 otherwise. $C(l, l_1)$ is the set of data units $j \in N_c^{(l)} : j > l_1$ that are not mutual descendants, i.e., for $j, k \in C(l, l_1) : j \notin D(k), k \notin D(j)$, where $D(j)$ is the set of descendants of data unit j. "$\setminus$" denotes the operator "set difference".

We use these results first to take an expectation over all possible cases of concealment for data unit l and then to sum over all data units in order to obtain

$$D(\boldsymbol{\pi}) = D_0 - \sum_l \sum_{l_1 \in N_c^{(l)}} \Delta d_l^{(l_1)} \prod_{j \in A(l_1)} (1 - \epsilon(\pi_j)) \prod_{l_2 \in C(l,l_1)} \left(1 - \prod_{l_3 \in A(l_2) \setminus A(l_1)} (1 - \epsilon(\pi_{l_3}))\right)$$

where D_0 is the expected reconstruction error for the presentation if no data units are received and $\Delta d_l^{(l_1)}$ is the reduction in reconstruction error if data unit l is not decoded on time, but is concealed with data unit l_1. Note that $\Delta d_l^{(l)} = \Delta d_l$.

To obtain $D(\boldsymbol{\pi})$ we have used an assumption that the packet losses affecting different data units are independent, in order to factor the expectation in $D(\boldsymbol{\pi})$. However, we still account for the dependencies between different packets associated with same data unit, as shown in Section 4. Using the independence assumption between data unit allows us to obtain a mathematically tractable solution for the optimal transmission policies, as shown below. The obtained solution is suboptimum for burst-loss channels, and we might obtain a better solution by taking the dependence into account. However, without the independence assumption we cannot factor the expectations over interdependent data units in $D(\boldsymbol{\pi})$ as products of the individual expectations. This can render the problem of computing the optimal transmission policies too complex to solve even for small sets of interdependent data units, as the solution space is exponential in the number of data units.

Finding a policy vector $\boldsymbol{\pi}$ that minimizes the expected Lagrangian $J(\boldsymbol{\pi}) = D(\boldsymbol{\pi}) + \lambda R(\boldsymbol{\pi})$, for $\lambda > 0$, is difficult since the terms involving the individual policies π_l in $J(\boldsymbol{\pi})$ are not independent. Therefore, we employ an iterative descent algorithm, called Iterative Sensitivity Adjustment (ISA), in which we minimize the objective function $J(\pi_1, \ldots, \pi_L)$ one variable at a time while keeping the other variables constant, until convergence [8]. It can be shown that the optimal individual policies at iteration n, for $n = 1, 2, \ldots$, are given by

$$\pi_l^{(n)} = \arg\min_{\pi_l} S_l^{(n)} \epsilon(\pi_l) + \lambda B_l \rho(\pi_l), \tag{1}$$

where $S_l^{(n)} = \sum_{l_1 : l \in N_c^{(l_1)}} S_{l,l_1}^{+(n)} - S_{l,l_1}^{-(n)} = S_l^{+(n)} - S_l^{-(n)}$ 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 client, given the current transmission policies for the other data units. Note that differently from [8], the sensitivity here consists of two nonnegative terms $S_l^{+(n)}$ and $S_l^{-(n)}$. The first term increases the sensitivity associated with data unit l in case l is in the ancestor set of data unit l_2 used for concealment of a data unit l_1. On the other hand, the second term reduces the sensitivity associated with l in case l is not in the ancestor set of l_2. This result is intuitive and allows us to better model the situations where data unit l is irrelevant for concealment of another data unit. Due to space considerations, we do not provide here the explicit expressions for the two sensitivity terms.

The minimization (1) 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 (1) for any l and λ', it suffices to know the lower convex hull $\epsilon(\rho) = \min_{\pi \in \Pi} \{\epsilon(\pi) : \rho(\pi) \leq \rho\}$ of the function, which we call the expected *error-cost* function. In the next section we show how to compute the expected error-cost function for the family of policies corresponding to sender-driven transmission with packet path diversity.

4 Computing the expected error-cost function

Assume that there are M network paths over which the server can simultaneously send a data unit to the client. Furthermore assume that there are N discrete transmission opportunities $t_0, t_1, \ldots, t_{N-1}$ prior to the data unit's delivery deadline t_{DTS} at which the server is allowed to transmit a packet for the data unit on the forward channel of any $m \leq M$ paths. The server need not transmit a packet at every transmission opportunity. The server does not transmit any further packets after an ACK is received on the backward channel of any of the paths.

At each transmission opportunity t_i, $i = 0, 1, \ldots, N-1$, the server takes an action $a_i = [a_{i1}, \ldots, a_{iM}]$, where $a_{im} = 1$ means that a packet is sent on the forward channel of path m and $a_{im} = 0$ means that no packet is sent on the forward channel of path m. Then, at the next transmission opportunity t_{i+1}, the server makes an observation o_i, where o_i is the set of acknowledgements received by the server in the interval $(t_i, t_{i+1}]$. For example, $o_i = \{ACK_{j_1}^{m_1}, ACK_{j_2}^{m_2}\}$ means that during the interval $(t_i, t_{i+1}]$, ACKs arrived on the backward channels for the packets sent at time t_{j_1} and t_{j_2} on the forward channels of paths m_1 and m_2, respectively. The history, or the sequence of action-observation pairs $(a_0, o_0) \circ (a_1, o_1) \circ \cdots \circ (a_i, o_i)$ leading up to time t_{i+1}, determines the state q_{i+1} at time t_{i+1}, as illustrated in Figure 1. If the final observation o_i includes an ACK, then q_{i+1} is a final state. In addition, any state at time $t_N = t_{DTS}$ is a final state. Final states in Figure 1 are indicated by double circles.

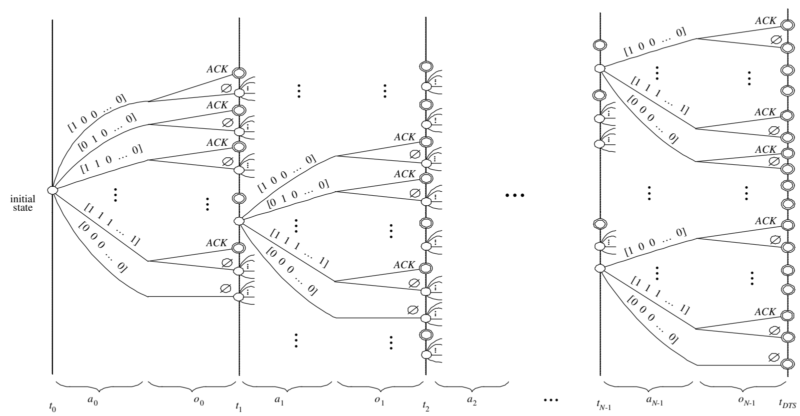

Figure 1: Markov decision tree for a data unit with packet path diversity.

The action a_i taken at a non-final state q_i determines the transition probabilities $P(q_{i+1}|q_i, a_i)$ to the next state q_{i+1}. Formally, a policy π is a mapping $q \mapsto a$ from non-final states to actions. Thus any policy π induces a Markov chain with transition probabilities $P_\pi(q_{i+1}|q_i) \equiv P(q_{i+1}|q_i, \pi(q_i))$, and consequently also induces a probability distribution on final states. Let q_F be a final state with history $(a_0, o_0) \circ (a_1, o_1) \circ \cdots \circ (a_{F-1}, o_{F-1})$, and let $q_{i+1} = q_i \circ (a_i, o_i)$, $i = 1, \ldots, F-1$, be the sequence of states leading up to q_F. Then q_F has probability $P_\pi(q_F) = \prod_{i=0}^{F-1} P_\pi(q_{i+1}|q_i)$, transmission cost $\rho_\pi(q_F) = \sum_{i=0}^{F-1} \sum_{m=1}^{M} a_{im}$, and error $\epsilon_\pi(q_F) = 0$ if o_{F-1} contains an ACK and otherwise $\epsilon_\pi(q_F)$ is equal to the probability that none of the transmitted packets arrives at the client on time, given q_F. Hence, we can express the expected cost and error for the Markov chain induced by policy π: $\rho(\pi) = E_\pi \rho_\pi(q_F) = \sum_{q_F} P_\pi(q_F)\rho_\pi(q_F)$, $\epsilon(\pi) = E_\pi \epsilon_\pi(q_F) = \sum_{q_F} P_\pi(q_F)\epsilon_\pi(q_F)$.

We wish to find the policy π^* that minimizes $\epsilon(\pi) + \lambda'\rho(\pi)$, as discussed in the previous section. We do that by enumerating all possible policies π, plotting the error-cost performances $\{(\rho(\pi), \epsilon(\pi))\}$ in the error-cost plane, and producing an operational error-cost function for our scenario. At every transmission opportunity t_i we find π^*, where $\{(\rho(\pi), \epsilon(\pi)) : \pi \in \Pi\}$ is calculated conditioned on q_i and all the policies under consideration are consistent with the history $(a_0, o_0) \circ (a_1, o_1) \circ \cdots \circ (a_{i-1}, o_{i-1})$ leading up to state q_i at time t_i. Then, a_i is set to the first action $\pi^*(q_i)$ of π^*, and the procedure is repeated at each successive transmission opportunity until a final state is reached.

In the following we provide expressions for $\epsilon(\pi)$ and $\rho(\pi)$. Let t_i be the current transmission opportunity and let $C^F_{jm}, C^B_{jm} \in \{1, \ldots, K\}$ be respectively the states on the forward and the backward channel of path $m = 1, \ldots, M$ at transmission opportunity $t_j : j \leq i$. We assume that the sender has this information available. This is a reasonable assumption, as any congestion control mechanism employed by a streaming media system will include some kind of channel estimation. The error-cost expressions for a policy π in this scenario generalize those for a sender-driven streaming over a single network path [8]. Moreover, contrary to [8], the channel state associated with a path is time-varying here. Therefore, the contribution to the error-cost of prospective transmissions in π at opportunities $t_j : j > i$ can be accounted for only as an expected value over all possible channel states at t_j. Hence we write

$$
\begin{aligned}
\epsilon(\pi) &= \left(\prod_{j<i,\, m\,:\, a_{jm}=1} P\{FTT^{C^F_{jm}} > t_{DTS} - t_j | RTT^{C^F_{jm} C^B_{jm}} > t_i - t_j\} \right) \times \\
&\quad \prod_{j \geq i,\, m\,:\, a_{jm}=1} \sum_{k=1}^{K} \mathcal{P}^{(j-i)}_{C^F_{im} k(F)} P\{FTT^k > t_{DTS} - t_j\} \\
\rho(\pi) &= \sum_{j \geq i,\, p\,:\, a_{jp}=1} \left(\prod_{l<i,\, m\,:\, a_{lm}=1} P\{RTT^{C^F_{lm} C^B_{lm}} > t_j - t_l | RTT^{C^F_{lm} C^B_{lm}} > t_i - t_l\} \right) \times \\
&\quad \prod_{i \leq l < j,\, m\,:\, a_{lm}=1} \sum_{k_1=1}^{K} \sum_{k_2=1}^{K} \mathcal{P}^{(l-i)}_{C^F_{im} k_1(F)} \mathcal{P}^{(l-i)}_{C^B_{im} k_2(B)} P\{RTT^{k_1 k_2} > t_j - t_l\}
\end{aligned}
\tag{2}
$$

where the first product term in both $\epsilon(\pi)$ and $\rho(\pi)$ accounts for previous transmissions (if any).

5 Experimental results

Here we investigate the end-to-end distortion-rate performance for sender-driven streaming of packetized video content over a single and over multiple network paths. The video content is a two layer SNR scalable representation of the sequence *Foreman*. Using H.263+ the first 130 frames of QCIF *Foreman* have been encoded into a base and enhancement layer with corresponding rates of 32 and 64 Kbps. The frame rate is 10 fps and the size of the Group of Pictures (GOP) is 10 frames, consisting of an I frame followed by 9 consecutive P frames. Performance is measured in terms of the luminance peak signal-to-noise ratio (Y-PSNR) in dB of the end-to-end perceptual distortion, averaged over the duration of the video clip, as a function of the available bit rate on the forward channel(s) of the network path(s).

In the experiments we use $T = 100$ ms as the time interval between transmission opportunities and 600 ms for the playback delay. Furthermore, we employ a $K = 2$ state Markov model for each path. The model parameters are kept same over all paths and are specified in Table 1. In particular, in Table 1a we specify the delay

	Loss	Delay		
	ϵ (%)	κ (ms)	μ (ms)	σ (ms)
State 1	3	25	75	50
State 2	15	25	275	250

(a) Loss and delay parameters.

	π_2	τ_2 (ms)
Model 0	0	0
Model 1	0.2	200
Model 2	0.5	1000
Model 3	0.8	2000

(b) State transitions.

Table 1: Network path characterization.

and loss characteristics for a channel state. We keep the same characteristics for the forward and the backward channel. The delay density is modeled using a shifted Gamma distribution specified with three parameters: shift κ, mean μ and standard deviation σ. Finally, the state transitions are modeled using two parameters: the stationary probability of being in State 2, π_2, and the expected duration of stay in State 2, τ_2, once a transition is made to this state. We employ four sets of values for these parameters denoted Model 0 - 3 in Table 1b. Due to the selected values Models 0 - 3 cover a range of possibilities in terms of the loss and delay characteristics exhibited on a network path.

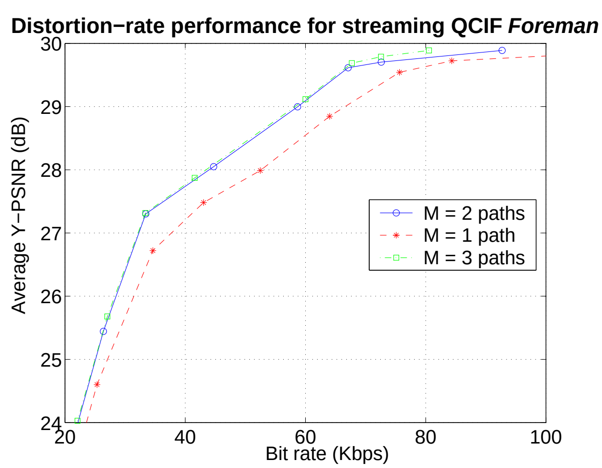

Figure 2: R-D performance for streaming over $M = 1, 2$ and 3 paths.

We first study the performance of the proposed framework as a function of the number of paths available. The state transitions are generated using Model 2 in these experiments. It can be seen from Figure 2 that streaming over 2 network paths can improve performance compared to the case of streaming over a single network path. An improvement is observed over the whole range of available rates. The gains in performance are most significant for the range of rates 30 - 70 Kbps and reach up to 0.65 dB. The difference in performance decreases as we move towards very low or very high transmission rates. The improved performance is due to the fact that having an alternate path for transmission reduces dramatically the probability of having to transmit on a forward channel that features degraded quality (State 2) at transmission. This ultimately contributes to a higher likelihood of delivering the media packets on time. Furthermore, it can be seen from Figure 2 that using further paths for streaming does not provide additional gains in performance, since the performances of $M = 3$ and of $M = 2$ are almost identical. As explained above having 2 network paths reduces substantially the likelihood of facing a degraded channel at transmission on every path. Therefore, adding one more path as an alternative does not provide further benefits, given the selected path model.

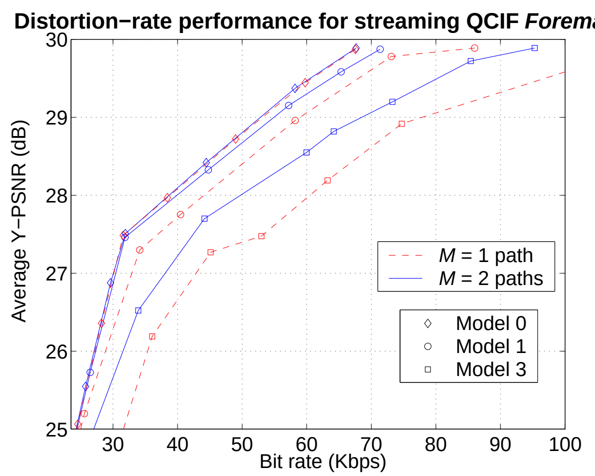

Figure 3: R-D performance for $M = 1, 2$ and different state transition models.

Next, we study the performance of the framework as a function of the quality of the network paths. As explained earlier depending on the state transition model, a path cah exhibit different levels of quality in terms of the loss and delay characteristics of packet transmissions. In Figure 3 we show the performance for streaming over $M = 1$ and over $M = 2$ network paths in case of Models 0, 1 and 3. It can be seen that streaming over two paths does not offer any advantages in case of Model 0. This is expected, as a path here does not switch between states and hence there is no need for an alternative routing of packets over another path. However, as the frequency of state transitions and the duration of stay in State 2 for a path increase on the average, the need for an alternative routing in order to avoid a bad quality path steadily increases. Thus, the difference in performance between $M = 2$ and $M = 1$ is largest when the state transitions on a path are governed by Model 3.

6 Conclusions

A framework has been presented that incorporates network path diversity in a rate-distortion optimized sender-driven streaming of packetized media. Using our framework a sender can exploit the availability of multiple paths over which media packets can be transmitted in order to obtain an improved performance over the case when only a single path is used. Experimental results for streaming video content demonstrate the benefit of using the proposed framework. The gains in performance are dependent on the quality of the paths in terms of loss and delay. In a follow-up work [9] we explore the concept of server diversity in a rate-distortion optimization framework for receiver-driven streaming of media.

References

[1] S. Savage, A. Collins, E. Hoffman, J. Snell, and T. Anderson, "The end-to-end effects of internet path selection," in *Proc. Data Communication, Ann. Conf. Series*. Cambridge, MA, United States: ACM, Aug. 1999, pp. 289–299.

[2] J. Apostolopoulos, "Reliable video communication over lossy packet networks using multiple state encoding and path diversity," in *Proc. Visual Communications and Image Processing*. Seattle, WA: SPIE, Jan. 2001, pp. 329–409.

[3] J. Apostolopoulos and S. Wee, "Unbalanced multiple description video communication using path diversity," in *Proc. Int'l Conf. Image Processing*. Thessaloniki, Greece: IEEE, Oct. 2001.

[4] N. Gogate, D.-M. Chung, S. Panwar, and Y. Wang, "Supporting image and video applications in a multihop radio environment using path diversity and multiple description coding," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 12, no. 9, pp. 777–792, Sept. 2002.

[5] Y. Liang, E. Setton, and B. Girod, "Channel-adaptive video streaming using packet path diversity and rate-distortion optimized reference picture selection," in *Proc. Workshop on Multimedia Signal Processing*. St. Thomas, US Virgin Islands: IEEE, Dec. 2002.

[6] J. Apostolopoulos, W.-T. Tan, and S. Wee, "Performance of a multiple description streaming media content delivery network," in *Proc. Int'l Conf. Image Processing*. Rochester, NY, USA: IEEE, Sept. 2002.

[7] A. Sehgal and P. A. Chou, "Cost-distortion optimized streaming media over DiffServ networks," in *Proc. Int'l Conf. Multimedia and Exhibition*, vol. 1. Lausanne, Switzerland: IEEE, Aug. 2002, pp. 857–860.

[8] P. A. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," *IEEE Trans. Multimedia*, 2001, submitted.

[9] J. Chakareski and B. Girod, "Server diversity in rate-distortion optimized streaming of multimedia," in *Proc. Int'l Conf. Image Processing*. Barcelona, Spain: IEEE, Sept. 2003, submitted.