EFFICIENT PATH AGGREGATION AND ERROR CONTROL FOR VIDEO STREAMING

Omesh Tickoo, Shivkumar Kalyanaraman and John W. Woods

ECSE Department, Rensselaer Polytechnic Institute
Troy, NY - 12180

ABSTRACT

This paper presents an efficient multiplexing and error control system to improve the streaming video performance over path aggregates. While providing the application with increased aggregate bandwidth, the scheme reduces the performance degradation due to high path latencies and loss rates. The reduction in effective loss and delay is achieved by smart multiplexing and exploiting the high latency paths to user's advantage. A novel out-of-order transmission algorithm utilizes the higher latency paths to transfer suitable frames from within the transmit buffer. We present an FEC strategy for our scheme that decouples the transmission of error correction frames from the associated data. This provides protection against correlated losses. Our scheme, while not completely optimized, can provide close to optimal performance at a considerably lower complexity. We verify the performance of our scheme using the *ns-2* simulator.

1. INTRODUCTION

Multimedia transmission over the Internet has gathered considerable interest in the recent years. Apart from the unreliability of best effort Internet, the limited bandwidth and path latencies pose major hurdles to efficient multimedia transport. Previous studies have shown that the use of path aggregates can overcome the bandwidth deficiency ([1] - [4]). Reference [4] proposes use of product codes to unequally protect the video frames while aiming to reduce the overall transmission delay over diverse paths. The previous work in the field aims at improving the performance under a particular constraint (loss, delay etc.) given a set of paths. It is important to consider the effect of a path diversity scheme on performance with the change in one or more of these constraints (path loss, delay, bandwidth, number of paths etc.). Multimedia transmission is highly susceptible to the path loss characteristics and latencies. Thus, it is important that a path aggregation scheme must not be limited to efficient bandwidth aggregation but must optimize the end user experience. This can be achieved by jointly reducing effective loss rate and latency/jitter in addition to maximizing the transmission rate.

In this paper we develop an efficient video multiplexing scheme over path aggregates. Our scheme, although not optimal, increases the video performance by an order of magnitude while keeping the implementation complexity much lower than the optimized case. We present a novel partitioning scheme that takes advantage of unequal packetization of the video and also uses an out of order transmission scheme to utilize high latency paths. Conventionally, the transport level optimizations have been decoupled from the application level information. We demonstrate that better streaming performance is achieved by using a content aware transport scheme. To deal with the short-term temporal dynamism of the best-effort path characteristics, on every path we use a TCP friendly congestion response scheme that is suited to multimedia delivery [5]. Hereafter, in this paper, we refer to our scheme as *Smart Multi-path Capacity Aggregation*, **SMCA**. An extended version of SMCA utilizes the path loss information to generate FEC and to decouple the FEC packets from the associated video packets over the path aggregates.

This work was supported in part by grants from Intel Corp., ARO (grant number DAAD19-00-1-0559) and DARPA (contract number F30602-00-2-0537).

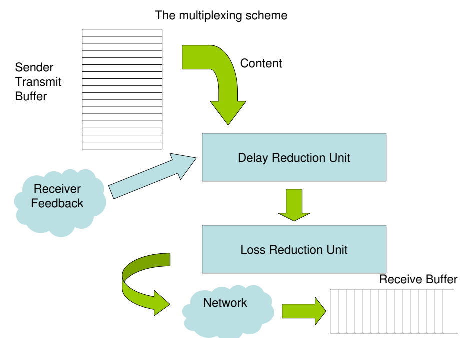

Fig. 1. Scheme Overview

2. DETAILED ALGORITHM

The SMCA architecture is represented in Figure 1. The sender transmit buffer is filled by the application with video packets in a serial fashion. The packets corresponding to the frame to be transmitted/decoded earliest occupy the head of the buffer. SMCA scheme is used to *choose* frames for transmission from the transmit buffer as described in subsections 2.1 and 2.2. There are two main stages in mapping the video frames to the appropriate paths. The first stage assigns the GOPs to a set of paths under delay constraints. This stage is represented by the delay reduction unit in Figure 1. The second stage protects each frame with FEC and maps each video and redundancy frame to a path using the content information. This second stage is represented by the loss reduction unit in Figure 1. The delay reduction unit is so named because it helps minimize effective delay by using out-of-order transmissions on high latency paths. Similarly, content based multiplexing of frames and error correction reduces the impact due to the network losses (hence the name loss reduction unit).

SMCA estimates the path characteristics at the sender. This information comprises of loss rate, bandwidth and latency values at different time instants. The estimation is done using the congestion window behavior and acknowledgment information from the transport scheme. For our implementation we use a TCP friendly transport scheme for transmission of video and estimation of the path characteristics. The transport scheme is based on Inverse Increase Additive Decrease (IIAD) algorithm proposed in [5]. For IIAD, the throughput, λ, is related to the loss probability, p, by Equation (1).

$$\lambda = \frac{K}{R\sqrt{\frac{3p}{\alpha}} + T_0 min(1, 3\sqrt{\frac{\beta p}{\alpha}})p(1+32p^2)} \quad (1)$$

where

K : Constant of proportionality

R : Connection's round trip time

α : Constant that determines the magnitude of rate increase

β : Constant that determines the magnitude of rate decrease

T_0 : Loss Interval (Time interval between two successive losses in different RTTs)

2.1. Delay Reduction

The receiver starts the playout after it receives the first GOP completely. Each frame in a GOP should arrive before the preceding GOP has been decoded. Let each GOP take T seconds to decode. After the first GOP has arrived, all the frames of the second GOP should reach the destination within T seconds. The frames of the third GOP must reach within $2T$ seconds and so on. This provides us with an expected arrival time for each frame.

Let the available paths be ranked in the increasing order of the latencies and the i_{th} path in this list be denoted by l_i. The latency associated with l_i is given by $L(l_i)$. The frames to be transmitted are ranked according to their position in the sender's transmit buffer with the frame at the head of the buffer being denoted by f_1 and the frame at position j from the head of the buffer denoted by f_j. The expected time at which the decoder starts processing frame j is denoted by $t(f_j)$ i.e. frame f_j must be present at the decoder buffer at $t(f_j)$ seconds from the current time. The packets are assigned to the paths using the following procedure:

Find the largest n such that

$$L(l_n) \leq t(f_1) \quad (2)$$

Then, considering just the delay requirements, the paths l_1 to l_n are suitable for any of the packets in the transmit buffer. We map the first few GOPs from the head of the buffer to these n paths. The goal is to fill these paths to capacity. If we view each path as a pipeline, the bandwidth-latency product of the path gives an estimate of the amount of data that can be present in the pipeline at any given instant. Thus, if $B(l_i)$ and $L(l_i)$ represent the available bandwidth and the latency associated with path i, the total bits assigned to l_i is equal to the bandwidth-latency product $B(l_i)L(l_i)$. We find the maximum number of integral GOPs from the head of the transmit buffer such that all the frames from these GOPs can be transmitted over paths l_1 to l_n. Thus, the number of frames mapped to paths l_1 to l_n is given by max integer q_1 which satisfies the following conditions:

$$\Sigma_{i=1}^{n}(B(l_i)L(l_i)) \geq \Sigma_{j=1}^{q_1} size(f_j) \quad (3)$$

and the q_1 frames form an integral number of GOPs.

Out of Order Transmission under delay constraints: The packets from $q_1 + 1$ onward are again grouped separately and mapped on to paths l_{n+1} onward. The number of paths, r in this case will be given by the maximum integer r such that

$$L(l_{n+r}) \leq t(f_{q_1+1}) \quad (4)$$

In case $L(l_{n+1}) \geq t(f_{q_1+1})$, we skip the GOPs in the sender's transmit buffer until we reach the start of a GOP (say frame, f_k, $k \geq q_1 + 1$) that satisfies the delay condition $L(l_{n+1}) \leq t(f_k)$. The skipped GOPs between the frames f_{q_1+1} and f_k come after the GOPs between frames f_1 and f_{q_1}. These skipped GOPs can wait for transmission and will be transmitted in the subsequent refresh periods when we re-evaluate frame and path rankings. The second group of paths l_{n+1} to l_{n+r} are assigned q_2 packets for transmission in a similar fashion as q_1 were assigned in the first step. We continue this grouping of paths and assignment of frames for transmission until either all the paths are categorized or we run out of video frames. Each set of paths and associated group of frames is referred to as a Delay Based Subcategory (DBS). The delay reduction unit reduces the overall transmission delay by sending the frames positioned higher up in the transmit buffer over the paths with higher latencies.

2.2. Loss Reduction

The second step involves the exact mapping of frames to paths within DBSs.

Smart Multiplexing: Equation (1) gives us an estimate of the available capacity given the loss rate of a path. We use the following algorithm for the transmission of I, B and P frames for a given set of GOPs within a DBS. The available paths are ranked in the increasing order of loss rates. The loss rates are measured using the information about number of acknowledgments received for a given number of packets sent over an interval [5]. The multiplexing scheme is a simple frame type based prioritizing mapping wherein the I frames are mapped to the available paths with low loss rates followed by the P frames. The B frames are then mapped to the remaining paths in increasing order of the loss rates.

The complexity of this smart multiplexing scheme is $O(N^2 + Q)$ where N is the number of available paths and Q is the number of frames per GOP. We compare the complexity of SMCA smart multiplexing with an opportunistic packet mapping scheme. The opportunistic packet mapping scheme transmits the packets from the head of the application's transmission buffer onto the paths that are ready to accept a packet at any instant of time. This greedy scheme does not utilize the knowledge of the content priorities nor does it exploit the path diversities. Hereafter, we refer to this scheme as *Opportunistic Packet Mapping Scheme* or **OPMS**. OPMS is an extended version of the scheme presented in [7] without video rescheduling. In case of OPMS the scheme simply sends the next frame on the next available path and thus the scheme is linear (in both number of frames and number of paths) with complexity of the order of $O(N + Q)$. Another multiplexing scheme that we compare the smart multiplexing complexity with is a completely optimized scheme like a pruned tree based approach [2]. The worst case complexity of a pruned tree based approach is $O(N^Q)$.

Including the complexity of the delay reduction unit makes the complexity of SMCA to be $O(N^4 + Q^2)$. The increase in complexity from OPMS is offset by the performance gains provided by using the novel out-of-order transmission scheme as a part of the delay reduction unit (Section 3).

Introducing Error Control: We now present an error control strategy for the path diverse video transmission. We design an FEC scheme based on the RS(n,k) [8] codes. The constraints for designing effective FEC are used to develop a simple error correcting scheme for non-prioritized data transmission. We extend this *uniform* FEC scheme to unequally protect hybrid video like MPEG and H.26x.

We need to partition the available capacity for transmission of both data and the associated FEC. Let the Q frames within a GOP be divided into k packets of average size s bytes each. Suppose that the total capacity of paths under consideration (path l_1 to path l_n) is m packets of s bytes each. In presence of error control protocol like FEC, the m packets will consist of k data (video) packets and $m - k$ FEC packets. It can be shown that for the packets to not exceed the carrying capacity of the n paths:

$$sm \leq \Sigma_n(B(l_i)L(l_i)) \tag{5}$$

Equation (5) along with the constraint that the average path loss must be countered by the FEC yields the Equation (6).

$$k \leq \frac{1}{s}(1 - P_a)\Sigma_n(B(l_i)L(l_i)) \tag{6}$$

where P_a is the average loss rate of the paths l_1 through l_n.

Due to space constraints Equations (5) and (6) are presented without the relevant derivations. Readers are referred to [9] for the derivations. Equation (6) provides the upper limit on the amount of the data (video) that may be transmitted reliably on the group of n identified paths. The rest of the capacity is used by the FEC packets. We present two different techniques to use this bandwidth. The first technique uses uniform error control to protect all the application data while the second technique exploits the content prioritization of hybrid video to unequally protect the most important video frames.

Uniform Error Correction: In case of uniform error correction all the data packets are treated alike and the data is protected with RS(m,k) FEC as given by Equations (5) and (6). The FEC packets are treated with the same priority as the data they protect. Note that since the data and FEC are *clumped* together, they will be exposed to similar and probabilistically correlated path conditions. We refer to the uniform error correcting schemes as SMCA-UFEC.

Unequal Error Correction: Prioritized error correction involves unequally protecting the video packets according to their relative importance within the GOP. For the case of MPEG and H.26x video, the relative importance of I, P and B frames dictates the amount of FEC allocated to the video frames per GOP. Of course, the total FEC allocated cannot exceed the limits posed by Equations (5) and (6). To make the video transmission robust to path loss correlations, the FEC for a frame is sent on a path that is *far* from the path (within the same DBS) on which the frame is sent. The scheme uses unequal protection of I and P frames and no protection for B frames. The FEC is *decoupled* from the transmission of the data by reserving the last paths in the DBSs (the paths with higher loss rates) for transmission of FEC. In this case the FEC packets have the lowest transmission priority within the GOP.

Reordering at receiver: The final step involves reordering of frames at the receiver before decoding. This is done by a simple in-place buffer filling scheme that inserts frames in the playback buffer at their correct playout position thus achieving reordering.

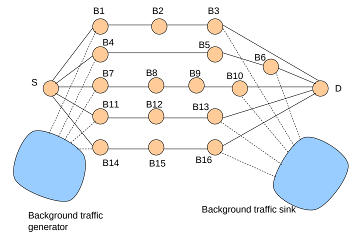

Fig. 2. Simulation Topology

3. PERFORMANCE MEASUREMENT

3.1. Simulation Set-up

Figure 2 shows the set-up used for verifying SMCA performance. Video source S multiplexes video traffic destined for destination D over multiple paths constituted by the hosts/router $B1$ to $B16$. The background traffic generators consist of sources transmitting FTP and constant bit rate (CBR) traffic. The bandwidth of each link varies between $300kbps$ to $1Mbps$. A 20 packets buffer is provided at each transmit interface of the source and the nodes $B1$ to $B16$. We compare the performance of our scheme with OPMS and a Pruned Tree (**PT**) approach (Subsection 2.2). The PT algorithm was implemented for 5 paths (N=5) of Figure 2 and 16 frames (Q = 16) within a GOP. We used the *Flower Garden* video test sequence in the simulations. The sequence was SIF resolution (352 x 240 pixels) at 30 fps. The Flower Garden sequence was encoded using H.26L encoder. The GOP had 16 frames in the following order: IBBPBBPBBPBBPBBP. The bit-rate was 1.7 Mbps and the average packet size was 700 bytes. The sender transmit buffer length was set equal to the average length of 4 GOPs of video frames.

In the following subsection we present our performance evaluation results. It is important to note that the average delay values we quote in the following subsection correspond to the values that were administratively configured in the associated topologies. The delays due to intermediate node buffer occupancy are in addition to the values we quote.

Due to the space constraints we present the results for the uncorrelated topology of Figure 2 only. SMCA gives similar performance improvements with correlated paths also. The readers are referred to [9] for results with correlated topology. [9] also presents the improvements obtained by using SMCA-UFEC and SMCA-UEP. It is observed that while SMCA-UEP outperforms SMCA and SMCA-UFEC, the improvements increase with the increase in the average loss rate of the paths. This is expected since SMCA-FEC provides robustness in presence of high losses by protecting the important data and making sure that the important data and the associated FEC is decoupled in case of correlated losses.

3.2. Results

Table 1 presents the gains in PSNR achieved with changing number of paths. The substantial gain of more than $7dB$ when the network resources are diversified among 5 paths shows that SMCA uses path diversity to user's advantage. The total number of paths was varied from one path ($n = 1$) to five paths ($n = 5$) while

Paths	1	2	3	4	5
PSNR(dB)	20.98	22.48	25.42	26.02	28.34

Table 1. Average PSNR Variation with Number Of Paths

Avg. Loss Prob.	SMCA PSNR(dB)	PT PSNR(dB)	OPMS PSNR(dB)
0.05	29.32	31.82	26.06
0.1	29.03	29.02	24.43
0.35	26.32	26.86	18.21
0.4	22.78	20.31	11.64

Table 2. Gains with Loss Variation

keeping the aggregate bandwidth fixed at $1.3Mbps$, average loss probability fixed at 0.1 and average path delay fixed at $30ms$.

The gains in performance with varying loss characteristics are shown in Figure 3 and Table 2. The average path delay was set at 30ms. For each curve in Figure 3, the variation in the individual path loss probabilities was set at a maximum of 50% from the average. All the PSNR values are averaged over 30 runs of the simulation. We note that the SMCA scheme performs much better than the OPMS under conditions of high average path loss. The gains in performance with varying delay characteristics are shown in Figure 4 and Table 3. The average path loss probability was set at 0.1. Again, all the PSNR values are averaged over 30 runs of the simulation. We note that the SMCA scheme performs much better than the OPMS under conditions of high average delay.

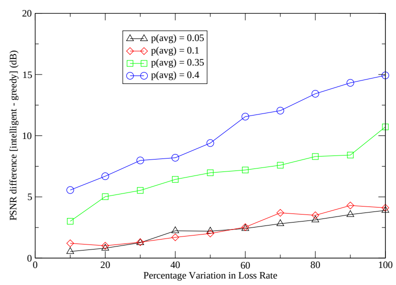

Fig. 3. Gains with Loss Variation

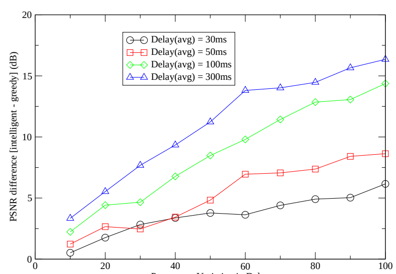

Fig. 4. Gains with Delay Variation

From Tables 1 and 2, we observe that the SMCA performance is comparable to the completely optimized PT approach under favorable network conditions (low loss rate and delay). As the conditions become unfavorable SMCA outperforms the PT approach. This improvement in performance can be attributed to the out of order transmission scheme adopted by SMCA. This allows SMCA to utilize the paths are rejected by the PT transmission scheme under delay constraints.

Avg. Delay	SMCA PSNR(dB)	PT PSNR(dB)	OPMS PSNR(dB)
30ms	30.12	31.83	27.96
50ms	28.32	29.46	24.33
100ms	25.12	24.21	19.19
300ms	21.78	18.73	11.03

Table 3. Gains with Delay Variation

4. CONCLUSION

A route aggregation scheme, SMCA, that exploits the diversity in network paths to satisfy real-time application's transmission requirements was presented. SMCA uses a novel out-of-order transmission strategy to exploit high latency paths for transferring suitable packets from the transmit buffer. While utilizing the otherwise useless bandwidth, the out-of-order transmission scheme also helps reduce the overall transmission delay. A smart content based multiplexing scheme is used by SMCA to counter the effects due to network loss. The multiplexing scheme, though sub-optimal, provides gains at much lower complexity than a fully optimized multiplexing scheme. This scheme can be used to multiplex both video and associated FEC in a decoupled manner to avoid performance degradation due to correlated network losses. The simulation results show that SMCA performs better than an opportunistic packet mapping scheme and comparable to a full optimized multiplexing scheme. Performance improvements gained using SMCA increase with the path diversity and higher values of average path loss and latency.

5. REFERENCES

[1] J. G. Apostolopoulos, "Reliable Video Communication over Lossy Packet Networks using Multiple State Encoding and Path Diversity," *Visual Communications and Image Processing*, January 2001.

[2] W. Xu, S. S. Hemami, "Efficient Partitioning of Unequal Error Protected MPEG Video Streams for Multiple Channel Transmission," *Proc. IEEE Int. Conf. on Image Processing*, Rochester, NY, Sept. 2002.

[3] T. Nguyen and A. Zakhor, "Path Diversity with Forward Error Correction System for Packet Switched Networks," *INFOCOM*, April 2003.

[4] W. Xu, S. S. Hemami, "Delay-Optimized Robust Transmission of Image over Multiple Channels," *Proc. IEEE Int. Conf. on Multimedia and Expo*, Baltimore, MD, Jul. 2003.

[5] D. Bansal and H. Balakrishnan, "Binomial Congestion Control Scheme," *Proc. of IEEE INFOCOM*, Tel-Aviv, Israel, March 2000.

[6] I. V. Bajic, O. Tickoo, A. Balan, S. Kalyanaraman, and J. W. Woods, "Integrated end-to-end buffer management and congestion control for scalable video communications," *Proc. IEEE ICIP 2003*, Barcelona, Spain, September 2003.

[7] J.C. Chen and S.-H. Chan, "Multipath Routing for Video Unicast over Bandwidth-Limited Networks," in Proceedings of IEEE Globecom'01, Vol. 3, pp. 1963-1697, San Antonio, Texas, 25-29 November, 2001.

[8] S. B. Wicker and V. K. Bhargava, "Reed-Solomon Codes and Their Applications,"*Wiley-IEEE Press*,, September 1999.

[9] http://networks.ecse.rpi.edu/∼tickoo/SMCA.pdf