OPTIMIZING SP-FRAMES FOR ERROR RESILIENCE IN VIDEO STREAMING

Wai-tian Tan, Gene Cheung

Mobile & Media Systems Lab, Hewlett-Packard Laboratories

ABSTRACT

SP-frame is a new picture type of H.264 that can be perfectly reconstructed using one of several reference frames. In this paper, we discuss how this property of SP-frame can be exploited for controlling error propagation caused by packet losses. We first illustrate the benefits of the scheme through example. We then present results for optimized streaming where PSNR performance of proposed usage of SP-frame is compared to that of P-frames only. Results show that SP-frames can noticeably reduce distortion caused by packet losses compare to schemes based on P-frames only.

1. INTRODUCTION

When there is sufficient bandwidth and time, the ideal approach to packet loss recovery is retransmission. When retransmission becomes impractical due to bandwidth or latency constraints, measures to control error propagation caused by packet losses are needed. To this end, common approach includes the use of intra frames and blocks, multiple description coding, and NewPred in MPEG-4. Each of these approaches has its merits and drawbacks. Intra coding incurs large penalty in coding performance. Multiple description coding often requires multiple disjoint paths to be effective. NewPred requires a live encoder and is not applicable to stored content. As we will discuss later, the use of SP-frames offer an interesting alternative for controlling error propagation.

Beyond the traditional I-frame and P-frame, a new frame type *SP-frame* is introduced in H.264. Readers interested in detailed coverage of SP-frames are encouraged to read [3, 2]. A key characteristic of SP-frames is that they permit identical reconstructions of a picture from one of several possible reference frames. In Fig 1, SP-frames SP'_4 have identical reconstructed picture as that of SP-frame SP_4, even though SP_4 predicts from P_3, and SP'_4 predicts from P_1. The SP-frame SP_4 is referred to as a *primary* SP-frame, while SP'_4 is referred to as a *secondary* SP-frame.

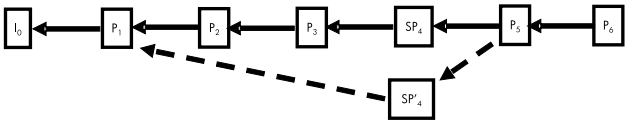

Fig. 1. Example usage of SP-Frames for Error Resilience. The original sequence is transmitted as $I_0P_1P_2P_3SP_4P_5P_6$. The sequence $I_0P_1SP'_4P_5P_6$ allows perfect reconstruction of P_5 and effectively stopping error propagation due to loss of P_2, P_3 or SP_4.

In this paper, we investigate a novel approach in which SP-frames are used to achieve two objectives, namely enhancing error-resilience by limiting error propagation, and providing rate-scalability for adaptation to time-varying channels. In Section 2, we first discuss how SP-frames can achieve objectives described above. Then, we compare the PSNR performance of optimized streaming using SP-frames to that of using P-frames only. The optimization procedures are covered in Section 3, while simulation results are presented in Section 4. We then conclude with a summary. A shorter version of this paper is presented in [5].

2. USING SP-FRAMES FOR ERROR RESILIENCE

Video coded using only P-frames has limited flexibility to address packet losses when transmitting under bandwidth or delay constrained environments. With SP-frames, an interesting alternative is illustrated in Fig 1, where a video sequence is compressed and transmitted as $I_0P_1P_2P_3SP_4P_5P_6$. Some frames, say P_2 and P_3, may be lost. Instead of retransmitting P_2 and P_3, the secondary SP-frame SP'_4 may be sent. The use of SP'_4 has two advantages. First, it offers possible bandwidth savings for bandwidth constrained environments. In particular, the byte-size of SP'_4 may be smaller than the byte-size sum of P_2 and P_3. Second, it extends transmission deadline for latency constrained environments. Specifically, SP'_4 has a later deadline than both P_2 and P_3.

The top graph of Fig 2 shows the effect of error propagation for video coded using P-frames only. We use the *Sean* sequence with an isolated frame loss at frame 24, and a burst loss of four frames starting at frame 24. We see a long tail of quality degradation despite improvement over time. The corresponding figure for SP-frames is shown in the bottom graph of Fig 2. We employ a primary SP-frame every 16 frames (Δ_{SP} = 16). A secondary SP-frame that predicts from frame 23 and reconstructs the next primary SP-frame (frame 32) is also sent in response to packet loss. We see that the distortion tail is effectively truncated with the reception of the secondary SP-frame. The same termination of error propagation can be

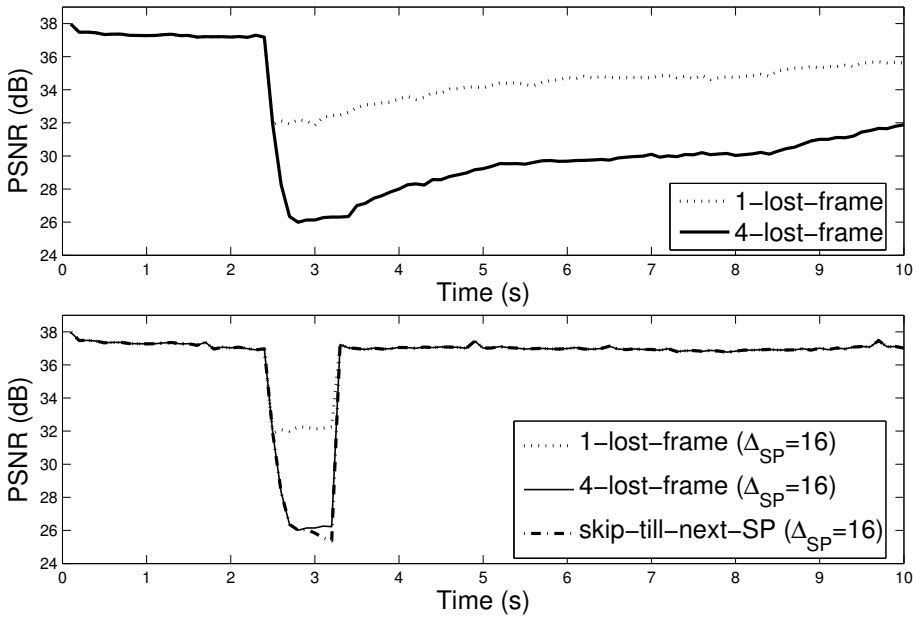

Fig. 2. Packet losses caused error propagation in *Sean* sequence. Top: only P-frames are used. Bottom: SP-frames are used.

achieved by frequent use of I-frames. The use of SP-frames is preferable in that primary SP-frames incur a smaller overhead in coding efficiency compared to I-frames, and secondary SP-frames are only transmitted when losses occur. In contrast, frequent use of I-frame represent a large overhead regardless of whether losses actual occur.

The use of SP-frames costs more bits than P-frames at the same quality. For bandwidth-constrained environments, the additional transmission of secondary SP-frame in response to losses can be justified by omitting the transmission of video until after the next primary SP-frame. This scheme is called *skip-till-next-SP* in Fig 2, and achieves higher bandwidth savings at the expense of higher temporary distortion.

Comparing *skip-till-next-SP* with P-frames only, we see that using SP-frames are clearly preferable under burst loss - somewhat higher transient distortion but much shorter tail. Fewer bytes are being sent using *skip-till-next-SP* scheme as well. For isolated loss, the P-frame only scheme results in a long distortion tail of smaller magnitude, whose distortion sum may be larger or smaller than that of using SP-frames. Visually though, the shorter distortion tail achieved by SP-frames is often preferable to that of P-frames.

In actual transmission, losses can occur in multiple places, and consists of a mixture of isolated and burst losses. Within the constraints of available bandwidth and latency, a sender may attempt limited retransmission in an optimized fashion as well. We would next compare the use of SP-frame with P-frames only in such settings.

3. OPTIMIZED STREAMING WITH S-FRAMES

We first present network and source models used in this paper. We then discuss how SP-frames should be encoded given a storage constraint. This is an off-line optimization before streaming begins. Finally, we discuss how optimized streaming is realized for video with P-frames only, and for video with SP-frames constructed using the off-line optimization.

3.1. Source Model

Similar to our earlier work on reference frame selection [4], we model each frame i, F_i, by a node in an directed acyclic graph as shown in Fig 3. The presence of edge $E_{i,j}$ means frame F_i can use F_j for motion compensation. There is only one edge for a P-frame, but two for a SP-frame. Associated with F_i is a deadline T_i, upon which the frame F_i must be delivered to the client or it will be rendered useless. Associated with each edge $E_{i,j}$ is a rate term $r_{i,j}$ specifying the byte size if F_i is encoded using F_j as reference. Generally, $r_{i,j}$ is large for large temporal distance between F_i and F_j. In addition, we assume an SP-frame is inserted into the video sequence every Δ_{SP} frames, and a secondary SP-frame i uses frame $i - \delta_{SP}$ as reference when performing motion prediction and compensation. Fig 3 shows an example when $\Delta_{SP} = 4$ and $\delta_{SP} = 2$. We will choose the parameters Δ_{SP} and δ_{SP} during an off-line optimization.

Finally, we assume constant frame rate of *FPS* frames per second and an initial client buffering delay of *BUF* seconds.

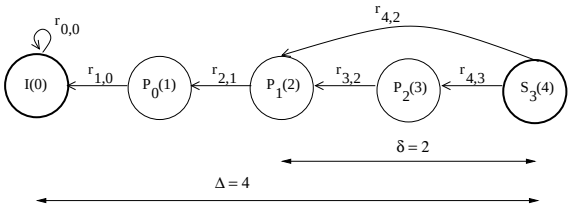

Fig. 3. An example directed acyclic graph source model

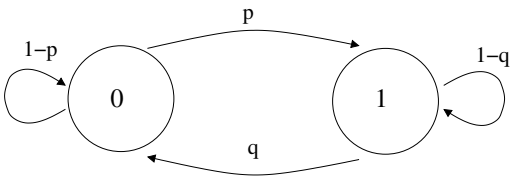

Fig. 4. Gilbert model for packet losses. By changing parameters p and q, different average packet loss rate and burst length can be achieved.

3.2. Network Model

We assume a network with constant bandwidth of C kbps, and a Gilbert packet loss process. A gilbert model of parameters p and q are given in Fig. 4, where the 0 and 1 states corresponds to packet delivery and loss, respectively. The parameters p and q corresponds to the state transition probabilities, and it is known that the average loss rate for such channel is given by $\pi = p/(p+q)$, and the average burst length is given by $1/q$. The effect of a burst loss process on a constant bandwidth channel is time varying achievable throughput. Different levels of the bandwidth constraint are realized by varying C in different experiments. We assume negligible network delays and instantaneous packet losses notification when performing the off-line optimization to determine Δ_{SP} and δ_{SP}. In simulation of optimized streaming, a shifted-Gamma-distributed delay is assumed.

3.3. Some Useful Definitions

Given Gilbert model of parameters p and q, several useful quantities can be computed [1]. For $i \geq 0$, we denote by $P(i)$ the probability of having at least i consecutive delivered packets following a lost packet, and by $p(i)$ the probability of having exactly i consecutive delivered packets between two lost packets. For the Gilbert channel, these quantities are given by:

$$p(i) = \begin{cases} 1-q & \text{if } i=0 \\ q(1-p)^{i-1}p & \text{otherwise} \end{cases}$$

$$P(i) = \begin{cases} 1 & \text{if } i=0 \\ q(1-p)^{i-1} & \text{otherwise} \end{cases}$$

Similar terms $q(i)$ and $Q(i)$ are defined by reversing the role of q and p. The probability of exactly m losses in n packets after an observed lost packet, $R(m,n)$, is given by:

$$R(m,n) = \begin{cases} P(n) & \text{for } m=0 \text{ and } n \geq 0 \\ \sum_{i=0}^{n-m} p(i) R(m-1, n-i-1) & \text{for } 1 \leq m \leq n \end{cases}$$

The probability of exactly m losses in n packets between two lost packets after an observed lost packet, $r(m,n)$, is given by:

$$r(m,n) = \begin{cases} p(n) & \text{for } m=0 \text{ and } n \geq 0 \\ \sum_{i=0}^{n-m} p(i) r(m-1, n-i-1) & \text{for } 1 \leq m \leq n \end{cases}$$

The probability of exactly m losses in n packets after a lost packet and preceding a received packet, $\bar{r}(m,n)$, is:

$$\bar{r}(m,n) = R(m,n) - r(m,n)$$

Quantities $S(m,n)$, $s(m,n)$ and $\bar{s}(m,n)$ are similarly defined, with $Q(n)$ and $q(n)$ in place of $P(n)$ and $p(n)$.

3.4. Optimized Off-line Encoding of SP-frames

Given a storage space limit V^* in bytes, we seek to determine the parameters Δ_{SP} and δ_{SP} that realize the highest expected number of correctly decoded frames under an assumed bandwidth and Gilbert channel. The storage constraint can be written as:

$$r_{0,0} + \sum_{i=1}^{N} r_{i,i-1} + \sum_{i=1}^{\left\lfloor \frac{N}{\Delta_{SP}} \right\rfloor} r_{i\Delta_{SP},\, i\Delta_{SP}-\delta_{SP}} \leq V^* \qquad (1)$$

where the three terms are the size of I-frame, size of P-frames and primary SP-frames, and size of secondary SP-frames, respectively. The number of inter-frames following an I-frame is denoted by N. Intuitively, large Δ_{SP} corresponds to small storage size due to the use of fewer SP-frames. In contrast, large δ_{SP} corresponds to large storage size due to large temporal distance from the reference frame.

Our off-line objective is to maximized the expected number of correctly decodeable frames:

$$\max_{\Delta_{SP}, \delta_{SP}} \left\{ \sum_{i=1}^{N} D_i \right\} \qquad (2)$$

where D_i, the successful *decoding* probability of frame F_i, can be expressed as:

$$D_i = \begin{cases} L_i & \text{if } F_i \text{ is I-frame} \\ L_i D_{i-1} & \text{if } F_i \text{ is P-frame} \\ L_i D_{i-1} + L_i^{(2)} D_{i-\delta_{SP}} & \text{if } F_i \text{ is SP-frame} \end{cases} \qquad (3)$$

where L_i denote the successful *delivery* probability of F_i, and $L_i^{(2)}$ denote the delivery probability of secondary version SP-frame F_i. For sequences with an I-frame followed by P-frames only, (3) reduces to $D_i = \Pi_{j=0}^{i} L_j$. For SP-frames, (3) corresponds to the two mutually exclusive cases with and without using secondary frames.

Central to the computation of L_i is the random variable ω_i, the number of available packet transmission opportunities for F_i after successful delivery of frames up to and including F_{i-1}. Initially, the starting number of transmission opportunities for the first frame is given by:

$$\bar{\omega}_0 = \lfloor BUF \times (1000 \times C/8)/s_{pkt} \rfloor \tag{4}$$

where s_{pkt} is the average packet size. In general, ω_i is a random variable with probability mass function $P_i(\omega_i)$. The crux of the off-line optimization is the derivation of $P_i(\omega_i)$ using a trellis, which we discuss next.

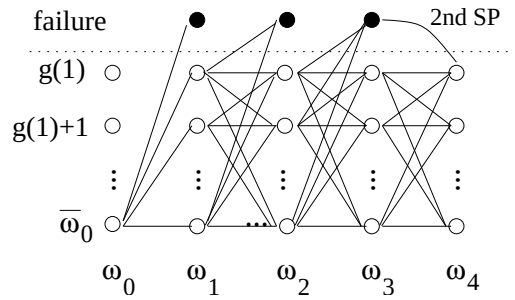

Fig. 5. Transmission Opportunity Trellis

Transmission Opportunity Trellis: We can track $P_i(\omega_i)$ using a trellis. The trellis for the source model of Fig 3 is illustrated in Fig 5. At stage 0 of the trellis, ω_0 equals $\bar{\omega}_0$ with probability 1, and L_0 can be calculated simply as:

$$L_0 = \sum_{j=h_0}^{\bar{\omega}_0} \pi R(j - h_0, \bar{\omega}_0) + (1-\pi) S(j, \bar{\omega}_0) \tag{5}$$

where h_i is the number of packets for F_i. To calculate $P_1(\omega_1)$ at next stage 1 of the trellis, we note that ω_1 receives a replenishment of one frame interval worth of transmission opportunities, denoted by $g(1)$, due to the later playback deadline of F_1 compare to F_0. $g(\tau)$ is defined similarly to (4):

$$g(\tau) = \lfloor \tau/FPS \times (1000 \times C/8)/s_{pkt} \rfloor \tag{6}$$

For ω_1 to assume the value $\omega + g(1)$, F_0 must have exhausted $\bar{\omega}_0 - \omega$ opportunities for successful delivery of h_0 packets. This happens if F_0 used exactly $\bar{\omega}_0 - \omega - 1$ opportunities to deliver $h_0 - 1$ packets, with the last packet transmitted successfully. More generally, to compute $P_i(\omega_i)$ for frame F_i that is not a secondary SP-frame, we write:

$$P_i(\omega_i) = \frac{P_i'(\omega_i)}{\sum_a P_i'(a)} \tag{7}$$

where P_i' is given by:

$$\begin{aligned} P_i'(\omega + g(1)) &= \sum_j P_{i-1}(j)\, \pi\, \bar{r}(j - \omega - h_{i-1},\ j - \omega - 1) \\ &\quad + P_{i-1}(j)\, (1-\pi)\, s(h_{i-1} - 1,\ j - \omega - 1) \end{aligned}$$

Having derived $P_i(\omega_i)$, we can compute L_i, similarly done in (5), as:

$$L_i = \sum_{\omega_i} P_i(\omega_i) \left(\sum_{j=h_i}^{\omega_i} \pi R(\omega_i - j, \omega_i) + (1-\pi) S(j, \omega_i) \right)$$

Computing $L_i^{(2)}$ for secondary SP-frame F_i is more complicated. We first assume that transmission of a secondary SP-frame is triggered when the sender failed to correctly deliver a non-essential frame. We call a frame an *essential* frame if it must be correctly decoded for future frames to be correctly decoded. In Fig 3, the essential frames are $I(0)$, $P_0(1)$, $P_1(2)$ and $S_3(4)$, while $P_2(3)$ is a *non-essential* frame. For an SP-frame F_i, we compute the probability that each of the non-essential frame F_k fails to be delivered $(1 - L_k)$. This will trigger the transmission of secondary frame F_i with $g(i-k)$ transmission opportunities. Writing $\tau = i - k$, we have:

$$\begin{aligned} L_i^{(2)} &= \sum_{k=i-\delta_{SP}+1}^{i-1} (1 - L_k) \left(\prod_{j=i-\delta_{SP}+1}^{k-1} L_j \right) \\ &\quad \left(\sum_{j=h_i^{(2)}}^{g(\tau)} \pi R(g(\tau) - j, g(\tau)) + (1-\pi) S(j, g(\tau)) \right) \end{aligned}$$

where $h_i^{(2)}$ is the size of the secondary SP version of frame F_i.

To complete the analysis, we need to compute the pmf $P_i^{SP}(\omega_i)$ of SP-frame F_i for future frames $F_j, j > i$, whose pmf $P_j(\omega_j)$ depends on $P_i^{SP}(\omega_i)$. $P_i^{SP}(\omega_i)$ is the weighted average of the two possible pmfs, $P_i(\omega_i)$ and $P_i^{(2)}(\omega_i)$, where the weights correspond to the probabilities that the primary and the secondary version of F_i are sent.

$P_i^{(2)}(\omega_i)$ for secondary SP itself is the weighted average of pmfs, where each pmf is the triggered result from the delivery failure event of a non-essential frame F_k:

$$\begin{aligned} P_i'^{(2)}(\omega_i) &= \sum_{k=i-\delta_{SP}+1}^{i-1} (1 - L_k) \left(\prod_{j=i-\delta_{SP}+1}^{k-1} L_j \right) \\ &\quad \Big[\pi\, \bar{r}(g(i-k) - \omega_i - h_i^{(2)}, g(i-k) - \omega_i) + \\ &\quad (1-\pi)\, s(h_i^{(2)} - 1, g(i-k) - \omega_i - 1) \Big] \\ P_i^{(2)}(\omega_i) &= \frac{P_i'^{(2)}(\omega_i)}{\sum_a P_i'^{(2)}(a)} \end{aligned}$$

Primary SP's pmf $P_i(\omega_i)$'s is calculated like a P-frame using (7) and (8). We now write the pmf for the SP-frame as:

$$P_i^{SP}(\omega_i) = P_i(\omega_i)\left(\prod_{j=i-\delta_{SP}+1}^{i-1} L_j\right) + P_i^{(2)}(\omega_i)\left(1 - \prod_{j=i-\delta_{SP}+1}^{i-1} L_j\right)$$

Given this is an off-line optimization, our approach is to employ these formulas to compute the optimal Δ_{SP}^* and δ_{SP}^* that maximize (2) while satisfying (1) via exhaustive search.

3.5. Optimized Real-time Streaming

For video with an I-frame followed by P-frames only, our optimization strategy is to retry transmission at every transmission opportunity until the packet is received, or the display deadline has passed. We then transmit the next packet and perform error concealment via frame-copy. We call this scheme *opt-P*.

For video encoded with SP-frames using Δ_{SP}^* and δ_{SP}^* determined from Section 3.4, our optimized streaming strategy proceeds as follows. For packets before the "switching point", such as $I(0)$, $P_0(1)$ and $P_1(2)$ in Fig 3, the sender would transmit all packets in order, and perform all necessary retransmissions until timeout. In Fig 3, this means $P_0(1)$ and $P_1(2)$ are always retransmitted until ACKed or deadline has passed. When packets after the "switching point", such as $P_2(3)$, is lost, we have two options: 1) retransmit lost packet; and, 2) ignore all current and future non-essential frames until the next SP-frame, and transmit the next secondary SP-frame. Our strategy in this paper is to choose the option with the largest expected number of frames that can be correctly decoded. We call this scheme *opt-SP*.

4. SIMULATION RESULTS

We performed simulations to compare PSNR achieved by *opt-SP* and *opt-P*. We use two ten-seconds QCIF sequences *Sean* and *Foreman* at 10 fps. The *Sean* sequence is a "talking head" sequence with a stationary background while *Foreman* contains complex motion. The sequences are coded using fixed quantization parameter of 27 for I-frames and P-frames, and QP and QS of 24 and 21, respectively, for SP-frames. These parameters yields roughly similar PSNR for both P and SP frames. The bit-rates were about 28 and 82 kbps for *Sean* and *Foreman*, respectively. The parameters Δ_{SP} is chosen from $\{4, 8, 12, 16\}$, and δ_{SP} from 2 to Δ_{SP} using procedure described above with V^* equals to twice the rate for P-only stream. We set the average loss rate of the Gilbert loss process to be 10%, with varying burst lengths. A shifted Gamma distribution with κ=50 ms, α = 4, and λ = 0.2 is used to model packet transmission delay, and a client buffer of 1 second is assumed.

The PSNR comparison for *opt-SP* and *opt-P* for *Sean* and *Foreman* sequences are shown in in Figs 6 and 7, respectively. The performance is shown in PSNR as function of the channel bandwidth C. Each point is averaged over 3000 independent simulated transmission of a 10 seconds clip.

We see that *opt-SP* generally outperforms *opt-P* over a wide range channel bandwidth and irrespective of burst length. For both sequences, we see that as bandwidth becomes more constrained, the performance improvement of *opt-SP* over *opt-P* increases. This is due to more opportunities in which secondary SP-frames need to be deployed under constrained bandwidth. As discussed in Section 2, the employment of secondary SP-frames causes high transient distortion, but consumes less bandwidth, and provides a relaxed transmission deadline for the secondary SP-frames. Specifically, at very constrained bandwidth, such as 28 kbps for *Sean* and 83 kbps for *Foreman*, *opt-SP* outperforms *opt-P* by 2-3 dB for *Sean*, and about 1 dB for *Foreman*. Since we have 10% loss rate, these channel bandwidth are smaller than the bit-rate for their respective video. At about 31 kbps for *Sean* and 91 kbps for *Foreman*, the average achievable throughput for the channel equals the media bit-rate. At those channel bandwidths, *opt-SP* outperform *opt-P* by about 1.7 dB and 0.7 dB for *Sean* and *Foreman*, respectively.

At about 36 kbps for *Sean* and 104 kbps for *Foreman*, the average throughput of the channel is 15% higher than the media bit-rate. At those channel bandwidths, we see that the performance improvement of *opt-SP* over *opt-P* largely disappears. This is due to the fact that when bandwidth is plentiful, most frames are received correctly for both schemes. We also notice that for channels with larger average lengths of burst losses, the performance improvement of *opt-SP* over *opt-P* is larger. Specifically, at 36 kbps for *Sean*, *opt-SP* outperforms *opt-P* by 0.2 dB when average burst length is 3, but 0.6 dB when burst length is 5. Similarly, at 103 kbps for *Foreman*, *opt-SP* outperforms *opt-P* by 0 dB when burst length is 3, but 0.5 dB when burst length is 5. This can be explained by the fact that when bandwidth is plentiful, a high burst channel is more likely to experience temporary throughput degradation, and therefore more opportunities for *opt-SP* to excel.

We also notice that as the average burst length of the channel increase, the channel bandwidth range in which *opt-SP* outperforms *opt-P* increases. Specifically, for *Sean*, *opt-SP* outperforms *opt-P* for channel bandwidth less than 37 kbps when average burst length is 3. The range is extended to 40 kbps and beyond for average burst lengths of 5 and 8. For the *Foreman* sequence, *opt-SP* outperforms *opt-P* when channel bandwidth is less than 104 kbps when average burst length is 3. The range is extended to 115 kbps and beyond 120 kbps when burst length becomes 5 and 8, respectively. Again, this is due to the fact that a more bursty channel is more likely to suffer from temporary throughput degradation under the con-

dition of high channel bandwidth.

Under otherwise identical conditions, PSNR is generally lower when burst length is longer. This is due to the difficulty in recovering from long bursts.

Results in Figs 6 and 7 shows PSNR averaged over time and different simulation runs. In practice, at the same PSNR, artifacts produced by *opt-SP* have shorter time support and is often preferable to that of *opt-P*.

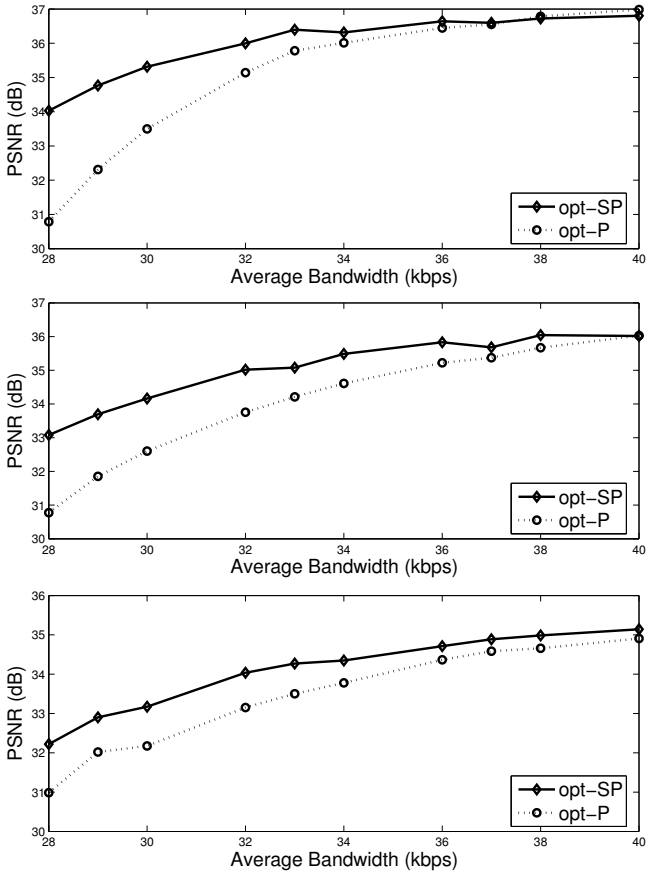

Fig. 6. PSNR performance for *Sean* for average channel burst lengths of 3 (top), 5 (middle) and 8 (bottom).

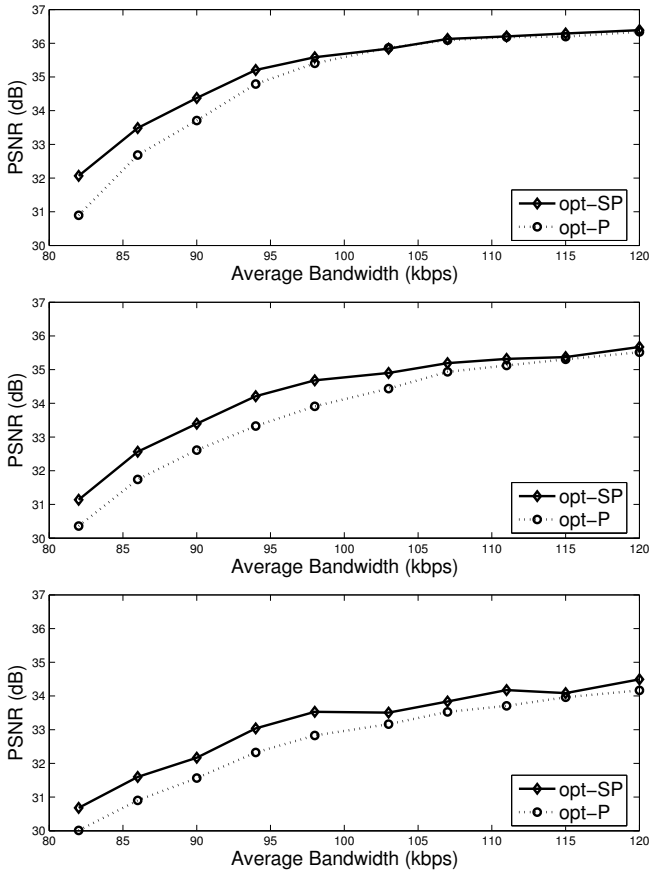

Fig. 7. PSNR performance for *Foreman* for average channel burst lengths of 3 (top), 5 (middle) and 8 (bottom).

5. SUMMARY

In this paper, we proposed a way to use SP-frames to effectively stop error propagation caused by losses. We proposed an off-line optimization procedure to compute some encoding parameters for SP-frames, and then showed via simulations that optimized streaming using SP-frames can achieve higher average PSNR than optimized streaming using P-frames only.

6. REFERENCES

[1] P. Frossard and O. Verscheure. Joint source/FEC rate selection for quality-optimal MPEG-2 video delivery. In *IEEE Trans. Image Processing*, volume 10, no.12, pages 1815–1825, December 2001.

[2] M. Karczewicz and R. Kurceren. The SP- and SI-frames design for H.264/AVC. In *IEEE Transactions on Circuits and Systems for Video Technology*, volume 13, no.7, July 2003.

[3] X. Sun, S. Li, F. Wu, J. Shen, and W. Gao. The improved SP frame coding technique for the JVT standard. In *IEEE International Conference on Image Processing*, Barcelona, Spain, September 2003.

[4] W. t. Tan and G. Cheung. SP-frame selection for video streaming over burst-loss networks. In *IEEE International Symposium on Multimedia*, Irvine, CA, December 2005.

[5] W. t. Tan and G. Cheung. Using SP-frames for error resilience in optimized video streaming. In *IEEE International Conference on Image Processing*, Atlanta, GA, October 2006.