

Packet Scheduling Algorithm for Wireless Video Streaming¹

Sang H. Kang and Avidesh Zakhor
Video and Image Processing Lab, U.C. Berkeley
E-mail: {sangk7, avz}@eecs.berkeley.edu

Abstract

We propose a class of packet scheduling algorithms for video streaming over wireless channels by applying different *deadline thresholds* to video packets (VP) with different importance. The importance of a VP is determined by its relative position within its group of pictures (GOP) and motion-texture context. We assume wireless channels with fixed round trip time, and with a feedback channel available from receiver to sender. Based on the deadline threshold, the sender schedules video packets in a different order from the original playback order. In trace-driven simulations, we evaluate the performance in terms of PSNR, and observe improvements over the conventional earliest-deadline-first schemes.

1. Introduction

The increasing number of mobile users and the growing demand for new multimedia streaming services have spurred a great deal of research on video streaming over wireless networks. The third generation (3G) wireless systems, e.g. CDMA2000 and Universal Mobile Telecommunications System (UMTS), are becoming available, which, together with the advance of low bit-rate video compression technology, will lead to wireless video streaming era in the near future.

Wireless channels have been shown to be band limited, error prone, and time varying, resulting in significant amount of loss and delay, and hence causing missed playback deadlines [1]. Video coding standards such as MPEG and H.263 have different types of frames such as I, P and B with different impact on video quality, when lost due to either corruption or missed deadlines. Furthermore, header, motion and texture fields within a frame have different levels of importance. Therefore, it is highly desirable to apply unequal error protection levels to different frame types and different fields.

There have been many studies on quality of service (QoS) protection for wireless video transmission. In [2], different Reed-Solomon codes are selected for video packet transmission in order to achieve unequal Forward Error Correction (FEC) based on the channel status, frame type, frame delivery deadline, etc. An FEC-based error control scheme is proposed in [3], where different levels of FEC are applied to different fields of MPEG-4 frames. Since sole use of FEC is ineffective for burst errors in fading wireless channels, there is a trend to use more intelligent Automatic Repeat Request (ARQ) techniques to ensure video delivery [4,5]. In [5], different video frame fields are protected by unequal ARQ schemes with different retransmission policies together with a frame scheduling algorithm. In [6], a packet scheduling algorithm is proposed for layered video streams in two-state Markovian wireless channels.

Current video streaming applications on the Internet usually use quite large, e.g. 5-20 seconds, pre-roll buffer to cope with the channel fluctuation and the inherent variable-bit-rate (VBR) nature of coded video sequences [7]. In this paper, we develop a class of packet scheduling algorithms for video streaming over wireless channels with dedicated fixed bandwidth and fixed round trip time. The basic idea is to re-order the transmission schedule of the video packets so as to use the available buffer at the receiver effectively. In our scheduling algorithm, we propose to apply different *deadline thresholds* to video packets with different importance in terms of the VP position within a GOP and motion-texture context, in order to obtain unequal loss ratios.

The rest of this paper is organized as follows. In Section 2, we develop a scheduling algorithm based on

¹ This work was supported by NSF under Grants CCR-9979442 and ANI-9905799, and by AFOSR contract F49620-00-1-0327

deadline thresholds for video packets with and without motion-texture discrimination. In Section 3, we evaluate our proposed system by trace-driven simulations. Finally, we have a discussion in Section 4.

2. Scheduling Algorithm

MPEG-4 achieves high compression ratio with several provisions for error resilience/concealment: resync-marker, data partitioning for P frames, and reversible variable length coding (RVLC) [8]. A frame is composed of several video packets (VP) separated by resync-markers. Using data partitioning mode, a VP may be further separated into motion and texture fields by the motion marker. In this paper, we use video packet mode with resync-marker with and without data partitioning mode. We assume no compression and/or expansion of total display time, i.e., we enforce that the playback duration at the receiver to be equal to the duration of the original video clip. Suppose we have frame sequence $\{F_0 F_1 F_2 \dots\}$ to be displayed at frame rate of f frames per second. If the receiver starts to display the first frame F_0 at time $t = 0$, then the n -th frame, F_n , is expected to be displayed at its deadline, i.e., at $t = n/f$. If a VP is not available at its expected display time at the receiver, it misses its deadline, and the receiver applies error concealment by copying corresponding macroblocks from the previous frame.

Scheduling of VPs involves determining their sending order, or selecting the next VP to be sent. One basic criterion for deciding sending order is the ‘deadline’ of VPs; this means the sender sends the VP with earliest deadline first (EDF). In this conventional EDF case, the dwelling time of VPs in the receiver’s buffer is minimized, and as a result, we achieve minimum required buffer size at the receiver. However, with predictive video coding, it is conceivable to add another criterion, namely, the relative ‘importance’ of data in the encoded video stream. For example, the loss impact of earlier frames within a group of picture (GOP) on video quality is much greater than that of the later frames within it. Furthermore, The motion part is more important than the texture part within a frame. Thus it is desirable to send the more important parts of video prior to the less important ones in order to ensure more opportunities for retransmissions in case of channel errors.

From the viewpoint of channel status, if the channel is in good condition without errors, then it is advantageous to use EDF criterion to send VPs in sequential order to obtain minimum average queue length in the receiver’s buffer. However, if the channel condition is poor with large error rates, then it is desirable to send more important VPs within GOPs first in order to achieve lower video distortion. Combining these two criteria we develop the following scheduling algorithm. In Section 2.1 we propose a basic frame based scheduling, and in Section 2.2, we extend it to the case with motion-texture discrimination.

2.1 Frame based Scheduling

We consider frame sequences: $\{F_0 F_1 F_2 \dots\}$, where $F_i = I$ or P frames. Assume that $F_0 = I$. Each GOP is composed of one I and $(M-1)$ P frames. Suppose that F_0 is on display at the receiver at time $t = 0$. Then the deadline of F_n is given by $t = n/f$.

We now define the type α_i of F_i where $\alpha_i = 0, \dots, M-1$ as

$$\alpha_i = i \bmod M \quad (1)$$

In other words, I frames have type 0, P frames immediately after I frames have type 1, and P frames immediately before I frames have type $M-1$. Frame F_i is composed of video packets with $VP_{i,j}$ denoting the j -th VP in F_i . We assign the following *deadline threshold* $d(VP_{i,j})$ in seconds to $VP_{i,j}$:

$$d(VP_{i,j}) = \frac{\beta \alpha_i}{M-1}, [\text{sec}] \quad (2)$$

where β in second is called importance coefficient, and will be explained later. For the frame based scheduling in this section, all VPs in a frame are assumed to have the same deadline threshold. The deadline threshold is larger for frames that occur later in a given GOP than for those that occur earlier, reflecting their relative importance.

Let us consider an example in Fig. 1, where the sender expects that bold-faced P frame F_n to be currently on display at the receiver. The length of the vertical bar above each frame corresponds to the deadline threshold of the VPs in the frame. Some of the future frames to the right of the frame on display are acknowledged as *successfully delivered*, i.e., all VPs in those frames have already arrived at the receiver intact. *Incompletely delivered* frames have VPs that are unsent or are outstanding, i.e., sent but not acknowledged yet.

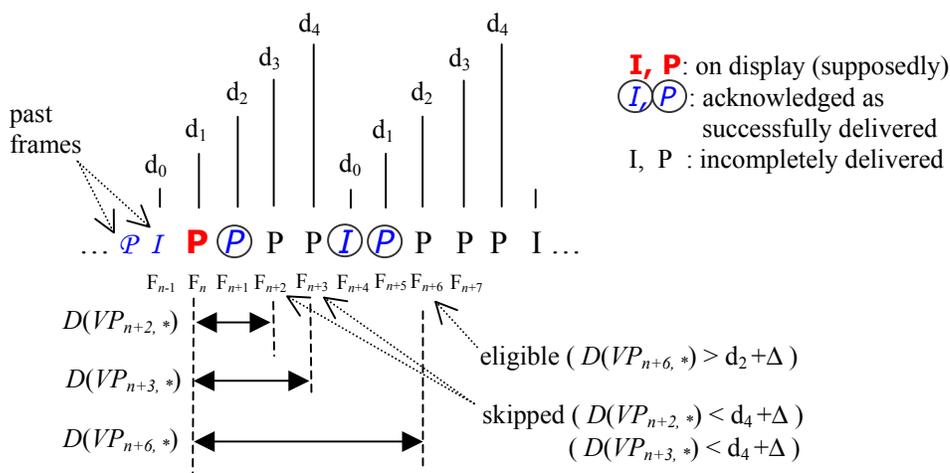


Fig. 1. Scheduling example

Each time the sender selects a VP to transmit, it runs the following scheduling algorithm based on two-step scanning over the frames. Suppose that current time is $t = n/f$, i.e., frame F_n is expected to be on display at the receiver. *In Step 1*, the sender scans $VP_{i,j}$ ($i > n$) to choose a candidate $VP_{i,j}$, with smallest i , which is neither sent nor outstanding and satisfies the following requirement:

$$D(VP_{i,j}) > d(VP_{i,j}) + \Delta, \quad (3)$$

where $D(VP_{i,j})$ denotes the time difference in seconds between current time $t = n/f$ and the deadline of F_i , $t = i/f$, Δ in seconds denotes the latency between the time the sender emits a VP and the time the VP arrives at the receiver. Intuitively, Eqn. (3) ensures that important VPs are sent first when the channel is poor. As shown in Fig. 2(a), if the channel has been poor for a long time with many errors, then few frames have been acknowledged as successfully delivered by this time, and the sender is likely to choose VPs with small $D(*)$ to be sent. When the sender is investigating VPs with small $D(*)$, then unimportant VPs with large $d(VP_{i,j})$ will have less chance to be sent according to Eqn. (3). On the other hand, if the channel status has been good for a long time, then many future frames have already been acknowledged as successfully delivered by this time as shown in Fig. 2(b), and it is likely that the sender transmits frames with large deadline $D(*)$. If $D(*)$ is larger than β , the maximum deadline threshold, then VPs will be sent in sequential order.

Let us re-consider the example in Fig. 1. In the first step of scanning, the incomplete frames, F_{n+2} and F_{n+3} , are skipped, and the next incomplete frame F_{n+6} is selected as a candidate because the situation is too

urgent to send unimportant frames such as F_{n+2} and F_{n+3} .

Suppose that $VP_{i,j}$ is chosen as a candidate in the first step scanning. Then, *in Step 2*, the sender rescans $VP_{k,l}$, $n < k < i$, to see if there is any VP with higher or the same importance as candidate $VP_{i,j}$. Then, among such VP(s), the sender finally chooses a VP with shortest deadline among them. Without such VP, the candidate $VP_{i,j}$ will be finally selected to be sent. In the example in Fig. 1, suppose that F_{n+6} is a successfully delivered frame. Then a VP in F_{n+7} will be selected as a candidate in the first step, but a VP in F_{n+2} will be finally selected to be sent in the second step of scanning. In the first step, the skipped VPs are not dropped permanently but waiting for their second chance in step 2.

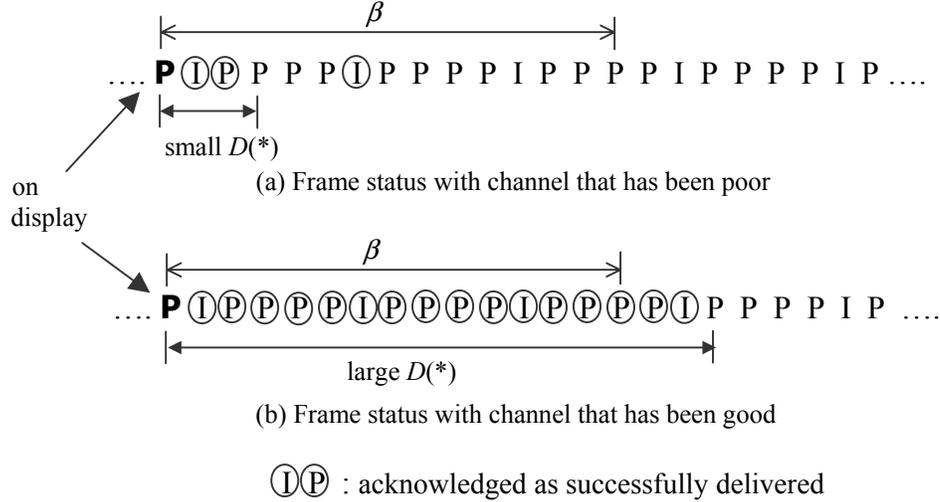


Fig. 2. Examples of frame status

We note that, in Step 1, the ‘importance’ criterion is dominant whereas the ‘deadline’ criterion plays a minor role. Conceptually, once a frame importance is determined in Step 1, ‘deadline’ criterion is applied once more in Step 2 to make up for its minor effect in Step 1. We define a frame to be missing its deadline if at display time more than $\theta\%$ of it is missing. We assume an error-free feedback channel and a limited number of retransmissions in the data link layer. With this assumption, the status of a sent VP can be determined within a fixed time. If a frame is found to have missed its deadline, or most certainly is going to miss its deadline, then the sender drops remainder of its GOP permanently. Outstanding VPs are not considered to have missed their deadline yet.

The importance coefficient β determines the range of frames ‘importance’ criterion is applied to. If the receiver’s buffer has shorter queue length than β , then VPs to be sent have smaller D^* than their deadline thresholds, and as a result ‘importance’ criterion will be applied to them as shown in Fig. 2(a). On the other hand, if the receiver’s buffer has larger queue length than β , then D^* of VPs to be sent is large enough and VPs are sent sequentially. β should be determined as an optimum value which depends on the receiver’s buffer size, initial amount of pre-roll buffer, statistics of burst error, etc.

2.2 Motion-Texture Discrimination (MTD) based Scheduling

In this section, we extend the frame-based scheduling by adding motion-texture discrimination. MPEG-4 supports data partitioning mode by separating the motion and the texture by motion marker inserted between motion and texture information within a VP. If the texture information is lost, this approach utilizes the motion information to conceal errors. Using this feature, we rearrange the data in each frame with *motion* and *texture* blocks; all motion vector fields are gathered in the motion block, and all DCT coefficient fields in the texture block. For frame F_i , the motion block is then divided into video packets

denoted by $VP_{i,j}^{(m)}$ s and the texture block into $VP_{i,k}^{(t)}$ s. We assign deadline thresholds to the VPs as follows:

$$d(VP_{i,j}^{(m)}) = \frac{\beta_m \alpha_i}{M-1}, \quad (4)$$

$$d(VP_{i,k}^{(t)}) = \frac{\beta_t \alpha_i}{M-1}, \quad (5)$$

where β_m and β_t denote importance coefficients for motion and texture, respectively. We choose $\beta_m < \beta_t$ to assign much lower priority to texture than motion. According to Eqns. (4) and (5), VPs in a frame have two different thresholds with motion-texture discrimination. For a frame missing its deadline, the remainder of its GOP is dropped using the same dropping mechanism as in the non-MTD case.

3. Performance Evaluation

We compare our proposed frame-based and motion-texture discrimination (MTD) based schemes with conventional earliest-deadline-first (EDF) scheme. We show the feasibility of the scheduling algorithm with trace-driven simulations. The simulation environment is as follows. We assume a fixed bandwidth wireless channel with random bit error. The channel bandwidth seen by the application is taken to be the same as the average bit rate of video. Channel bit error rate (BER) is given by 0.001. Radio Link Protocol (RLP) is used as data link layer. The round trip time τ seen by the RLP layer is fixed to be 100 ms. We use Microsoft MPEG-4 Visual Reference Software version 2 FDAM1-2.3-001213, to encode/decode video streams. We have added some error recovery/concealment features to the decoder: synchronization recovery based on the resync_marker and replacing a totally corrupted macroblock with the corresponding one in the previous frame.

Our scheduling algorithm is applied to application layer. We generate VBR streams with one I frame and 12 P frames in a GOP. At the receiver, the initial amount of pre-roll buffer length is chosen to be 15 seconds. For frame based scheduling $\beta = 5$ seconds, and for MTD based scheduling $\beta_m = 5$ seconds and $\beta_t = 1.5 \beta_m$. Using UDP-lite protocol, we assume that a UDP packet consists of 9 VPs. As the data link layer, RLP is assumed to be modified such that, an erroneous RLP packet is tagged and sent to upper layer, which makes the upper layer with our proposed scheduling locate corrupted VPs. In the RLP layer, the number of retransmissions is upper bounded to 2. In simulations, a frame with more than $\theta = 30\%$ loss at its display time is considered to be missing the deadline. The transmission channel bandwidth seen by the application layer is the same as the average traffic bandwidth of video.

We run simulations with the following two encoded video data. For each simulation, we repeat the video clip 3 times. Video in simulation 2 has higher motion activity.

Simulation 1 : Video clip: Talk Show ‘Larry King,’ QCIF, 1689 frames, $f = 15$ frame/sec,
Avg. video bit rate = 59.6 Kbps (Qp = 10)

Simulation 2 : Video clip: Movie trailer ‘Preacher’s Wife,’ QCIF, 1871 frames, $f = 15$ frame/sec,
Avg. video bit rate = 189.0 Kbps (Qp = 5)

In Fig. 3 we plot a trace of channel error and the corresponding frame number difference between successive transmissions of VPs as a function of time. In Fig. 3(a), condition 0 denotes good channel, and 1, bad channel. With EDF in Fig. 3(b), the difference is typically 1, i.e., sequential order, except for larger values corresponding to jumps to the next I frame in case of frames missing deadline, and occasional negative values for erroneous frame retransmissions. Figs. 3(c) and 3(d) show the results for frame-based and MTD based scheduling schemes, respectively; the curves are observed to have large deviation from 1,

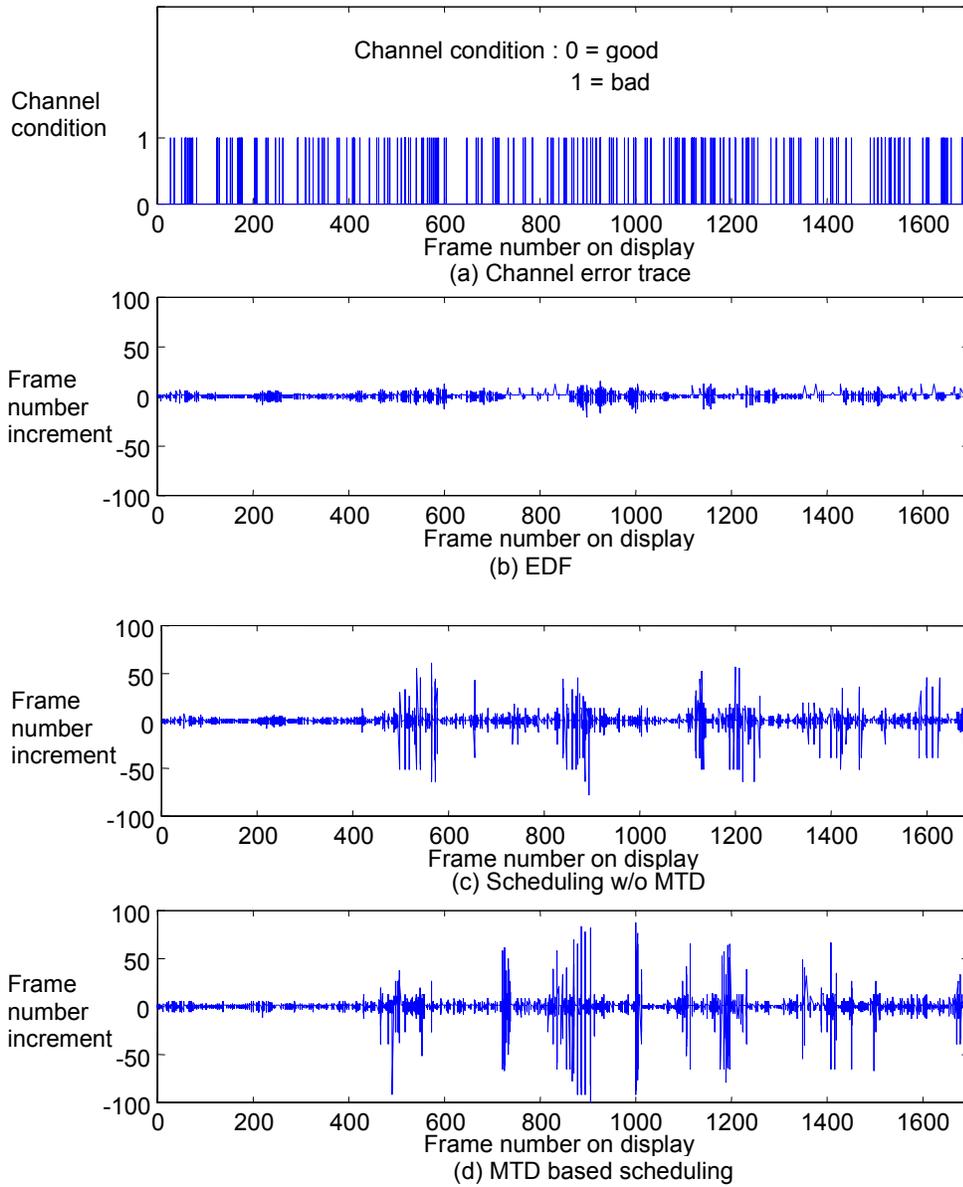


Fig. 3. Traces of channel error and frame number increment in simulation 2

with many negative values, showing that the sending order is shuffled. As seen, scheduling with MTD yields a little more shuffling than without.

3.1 Results for Frame based Scheduling

First we consider the frame based scheduling without motion-texture discrimination. We compare the VP loss rates based on our frame based scheduling scheme to that of EDF, in Fig. 4. As expected, scheduling results in much lower loss rate for earlier, more important frames. The values in parentheses correspond to average VP loss ratio, i.e., the ratio of the corrupted VPs plus missing-deadline VPs to the total transmitted VPs. The total numbers of transmitted VPs are 108,626 and 378,735 in simulations 1 and

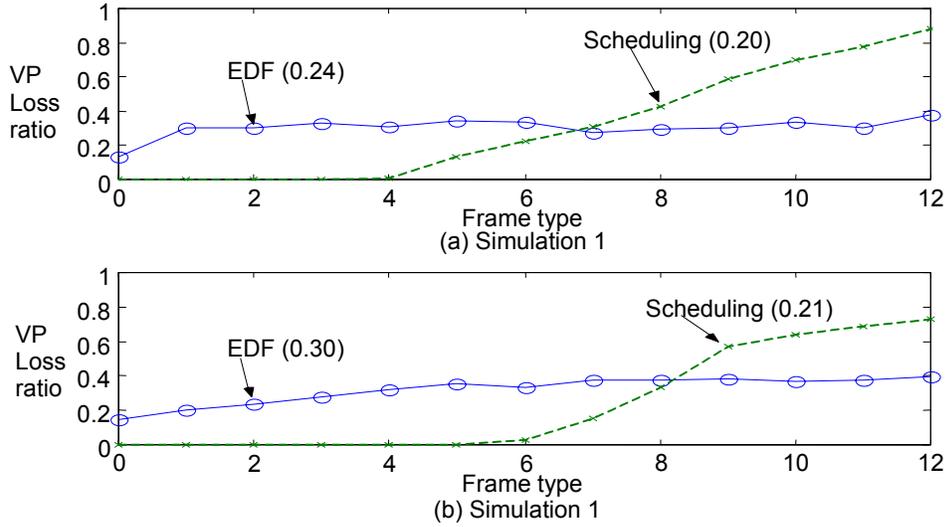


Fig. 4. VP loss rates comparison for frame based scheduling and EDF

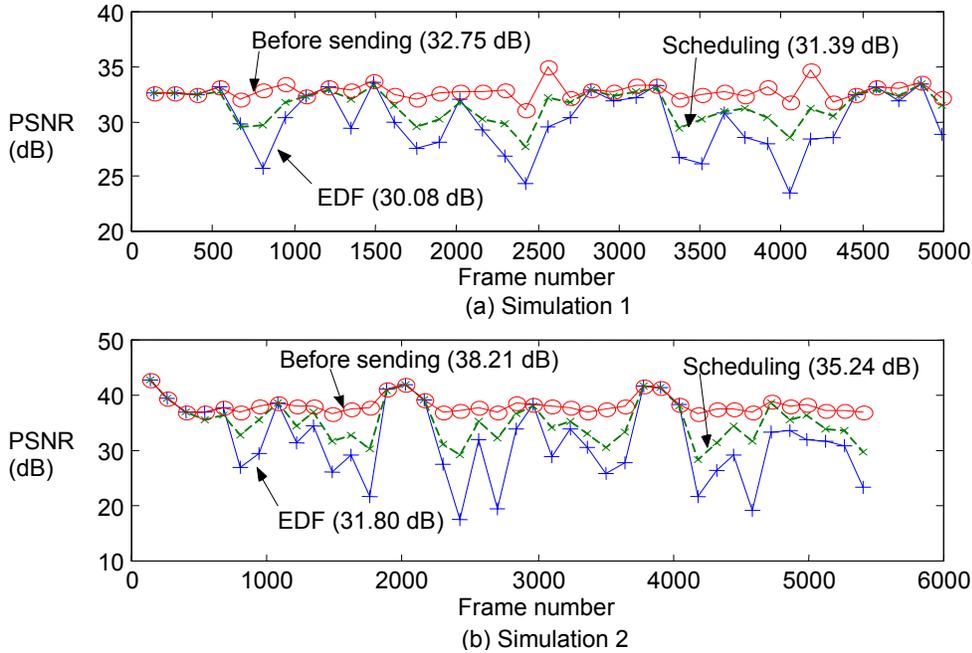


Fig. 5. Performance comparison

2, respectively. On average, scheduling yields lower loss ratio. This is mainly due to high loss ratio of important frames in EDF, which is accompanied by dropping of following VPs within the GOP.

The scheduling algorithm is also evaluated in terms of Peak Signal to Noise Ratio (PSNR). In Fig. 5, PSNR as a function of frame number is shown, where the average PSNR is reported every 9 seconds or 135 frames for three videos: the compressed video at the sender ‘before sending,’ the received video with EDF scheme, and the received video with our frame based ‘Scheduling.’ The values in parentheses are average PSNR in dB. Compared with the conventional EDF scheme, our scheduling scheme shows PSNR

improvement of 1.3 dB in simulation 1 and 3.4dB in simulation 2. It is observed that the scheduling scheme has an effect of smoothing out the bursty distortion periods and results in more even PSNR values compared with EDF scheme. For most human observers, distortion bursts result in more significant visual degradation than random distortions.

We compare burst error length statistics in Fig. 6, where we show the distribution of contiguous corrupted frames with at least one erroneous VP. With scheduling, long bursts are rarely observed, while 25-frame bursts or more are observed without scheduling. This effect is more significant in simulation 2, where we have larger traffic bandwidth.

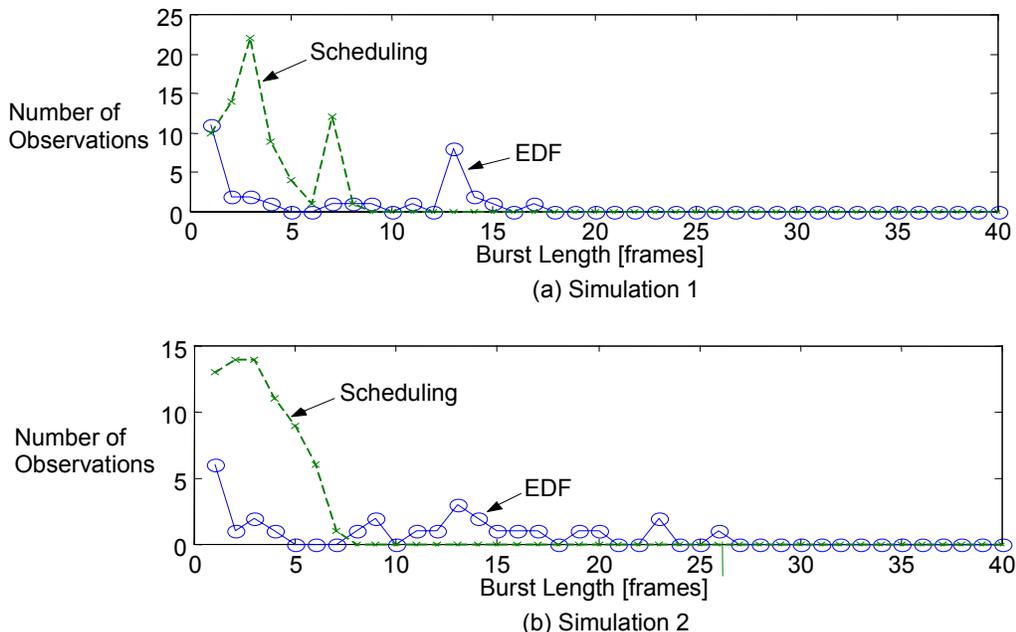


Fig. 6. Burst length statistics for contiguous corrupted frames with at least one erroneous VP for frame based scheduling

3.2 Results for Motion-texture Discrimination based Scheduling

So far, we have shown the effectiveness of our scheduling scheme without using the data partitioning mode of MPEG-4. Now using motion-texture discrimination, we can re-run simulations 1 and 2. We choose $\beta_m = 5$ seconds and $\beta_t = 1.5\beta_m$. With these coefficients, the motion parts of a given P frame have higher priority than the texture parts of up to 3 frames prior to that P frame. Fig. 7 shows the resulting PSNR curves for the scheduling with and without motion-texture discrimination. As seen, MTD results in a smoother curve. This is expected because the deadline thresholds take on more diverse set of values in Eqns. (4) and (5) than in Eqn. (2). Improvements in average PSNR are 0.3 dB in simulation 1 and 0.9 dB in simulation 2. This result is to be expected because simulation 2 corresponds to movie trailer with large motion activities rather than the head-and-shoulder video in simulation 1. Channel errors cause freezing in the case without MTD, and motion blur in the case with MTD.

One possible reason for poor PSNR performance improvement with MTD is that the MPEG-4 standard treats intra macroblocks after the motion-marker within a VP as texture part. Hence in our current implementation, VPs corresponding to these macroblocks are not given the high priority they deserve. This effect is particularly dominant during scene change with a large number of intra blocks. To partially alleviate this problem, we use a simple algorithm to detect such frames based on their size, and proceed to protect them by assigning β_m , rather than β_t , to all information corresponding to such frames. In our

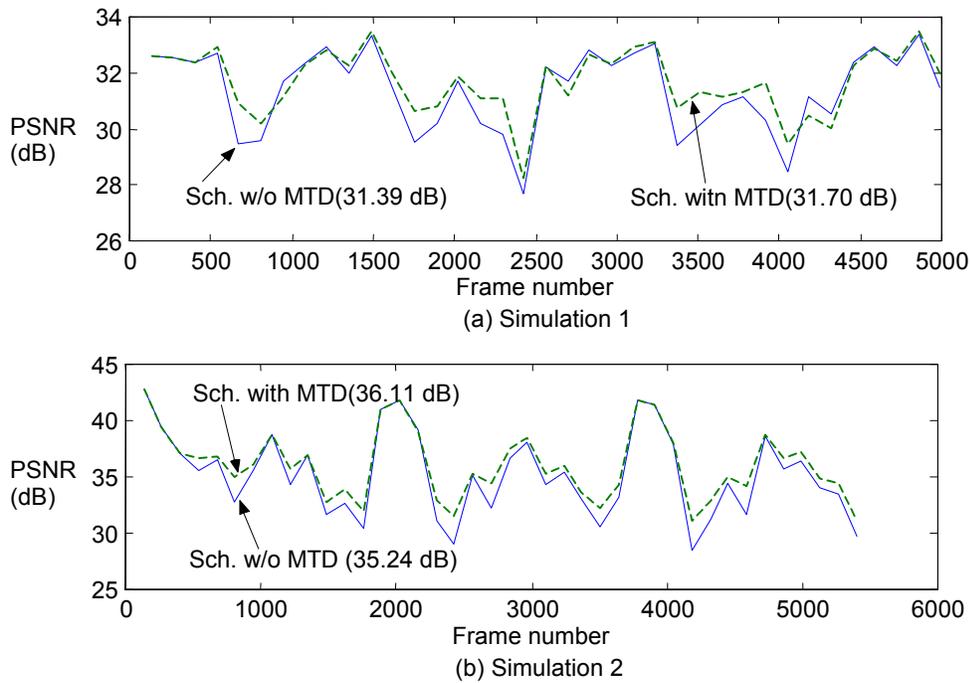


Fig. 7. Performance comparison between non-MTD and MTD based scheduling

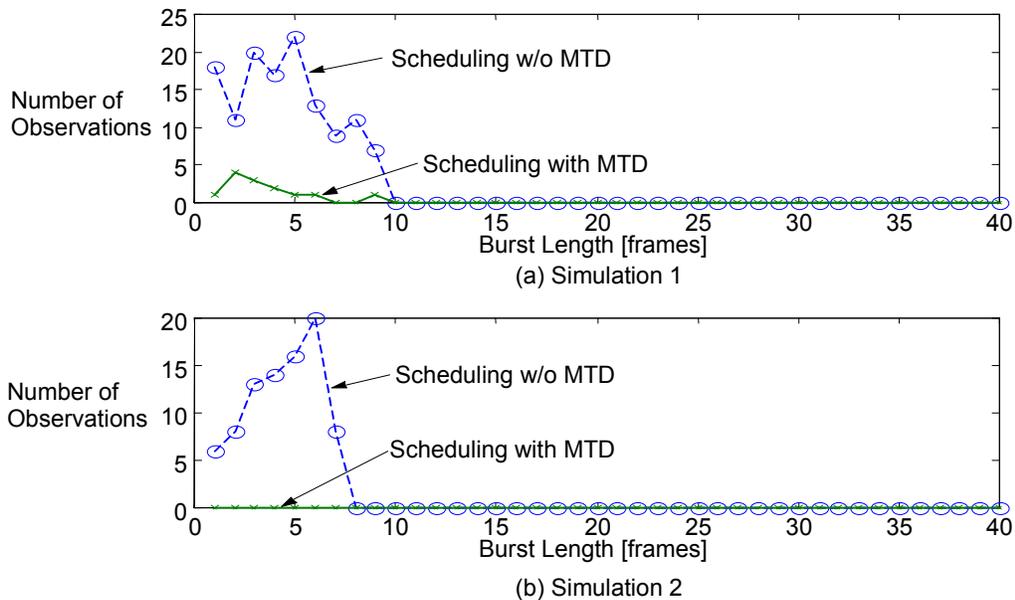


Fig. 8. Burst length statistics with completely missing frames with and without MTD

simulations, a P frame with 4 times larger size than the previous P frame is considered as a scene change.

In Fig. 8, we plot burst of contiguous frames that are completely missing with and without MTD. With MTD the sender tries to send motion vectors first which results in much reduced number of entirely missing frames. Notably no completely missing frames are observed in simulation 2.

Finally, in Fig. 9, we show the loss ratio for motion and texture VPs when using MTD based scheduling.

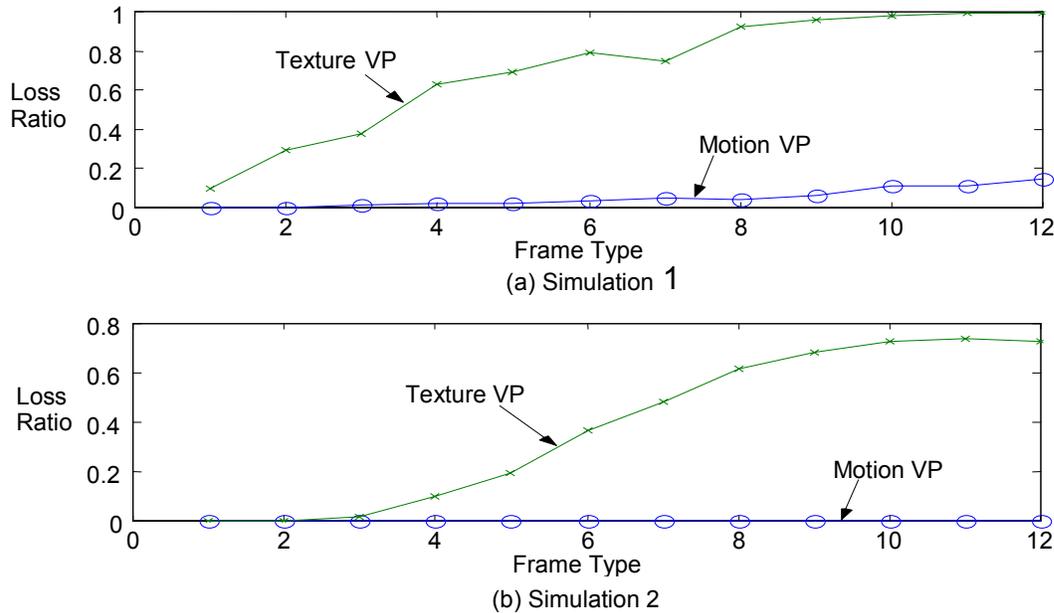


Fig. 9. Loss ratio of motion and texture VPs for MTD based scheduling

Since I frames have no motion part, we only plot for the P frames and thus the horizontal line has from frame types 1 through 12. We observe that the scheduling with MTD results in more protection for motion VPs.

4. Discussion

We have proposed a scheduling algorithm based on unequal deadline threshold for wireless video streaming. First, the proposed video packet scheduling algorithm is efficient in achieving unequal loss rate between VPs with different importance. This efficiency is shown in terms of PSNR. From a subjective point of view, the scheduling results in improvement in video quality with less overall distortion as well as more evenly distributed temporal distortion. Second, the motion-texture discrimination is more efficient for large motion clips and small quantization step, i.e., large size of texture fields. A key assumption in developing this algorithm has been fixed round trip time in order for the sender compute packet deadline and hence to decide as to when to drop packets. While this is a reasonable assumption in circuit switched wireless networks with error prone channels, in wired or wireless packet switched networks, round trip times vary as a function of time due to statistical multiplexing of the packets. Thus a future area of research is extending our algorithm to packet switched networks with time varying round trip times. Other directions for future research include extension to scalable video, and optimizing design parameter such as β for deadline thresholds.

References

- [1] M. Zorzi, "Some Results on Error Control for Burst-Error Channels Under Delay Constraints," *IEEE Trans. on Vehicular Tech.*, vol. 50, no. 1, pp. 12 – 24, Jan. 2001.
- [2] D. Qiao and K. G. Shin, "A Two-Step Adaptive Error Recovery Scheme for Video Transmission over Wireless Networks," *IEEE Infocom2000*, pp. 1698 – 1704, 2000.
- [3] J. Cai, Q. Zhang, W. Zhu, and C. W. Chen, "An FEC-Based Error Control Scheme for Wireless MPEG-4 Video Transmission," *IEEE WCNC 2000*, vol. 3, pp. 1243 – 1247, 2000.
- [4] M. Zorzi, "Performance FEC and ARQ Error Control in Bursty Channels under Delivery Constraints," *IEEE*

VTC'98, Ottawa, Canada, May 1998.

- [5] C.-C. Liu and S. S. Chen, "Providing Unequal Reliability for Transmitting Layered Video Streams over Wireless Networks by Multi-ARQ Schemes," *ICIP'99*, vol. 3, pp. 100 – 104, 1999.
- [6] Z. Jiang and L. Kleinrock, "A Packet Selection Algorithm for Adaptive Transmission of Smoothed Video over a Wireless Channel," *Journal of Parallel and Distributed Computing*, vol. 60, pp. 494 – 509, 2000.
- [7] G. Conklin, et al, "Video Coding for Streaming Media Delivery on the Internet," *IEEE Trans. on Circuits and Sys. for Video Tech.*, vol. 11, No. 3, pp. 269 – 281, March 2001.
- [8] ISO/IEC JTC 1/SC 29, "Information Technology – Coding of audio-visual objects – Part 2: Visual," Jan. 2000.
- [9] ISO/IEC JTC1/SC29/WG11, "Overview of the MPEG-4 Standard," Oct. 2000.
- [10] ETSI TS 100 946, "Digital Cellular Telecommunications System (Phase 2+)," v.7.1.0, Jan. 2000.
- [11] IETF RFC 2508, "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links," Feb. 1999.