SCHEDULING TRANSMISSIONS OF REAL-TIME VIDEO CODED FRAMES IN VIDEO CONFERENCING APPLICATIONS OVER THE INTERNET

*Jiangbo Lu** *and Benjamin W. Wah*†

*Advanced Digital Sciences Center, Singapore
†University of Illinois at Urbana-Champaign, USA
*jiangbo.lu@adsc.com.sg, †wah@illinois.edu

ABSTRACT

This paper presents a novel mechanism for scheduling packet transmissions of real-time video conferencing over the public Internet. By scheduling transmissions of video packets at different priorities in the presence of packet losses and delays, we aim to deliver videos with perceptually pleasing and consistent quality at receivers. Fundamental to this new mechanism is a realization of uneven packet transmission rates (UPTR) in the Internet, which allow an instantaneous packet transmission rate to fluctuate around an average rate, without significantly affecting the average loss rate and delay in transmissions. By exploiting this generic UPTR idea, we propose two application scenarios that provide positive design alternatives for robust real-time video delivery over the lossy Internet. Experimental results clearly show the advantages of our proposed methods over those previous solutions without awareness of UPTR.

Keywords— Internet, video conferencing, video compression, forward error correction, packet scheduling

1. INTRODUCTION

Internet-based video telephony and conferencing are exciting applications that allow users to perform social or business conversations in a visually vivid yet economical way. However, without guarantees on quality of service (QoS), real-time video transmissions over the public Internet are vulnerable to packet losses and delays. Providing perceptually high video quality under such error-prone conditions is, therefore, an important research problem.

To tackle this challenge, there are three classes of research conducted in the past decades [1, 2]. The first class is on the development of error resilient video coding techniques, such as intra macroblock (MB) updates [3] and redundant slices [4]. By taking a transport perspective, techniques in the second class investigate how video packets can be reliably delivered with methods like forward error correction (FEC) [5] and audio redundancy coding [6]. Schemes in the last class aim at mitigating the negative effects of lost packets at receivers by various loss concealment methods like spatial interpolation [7] and motion-compensated temporal prediction [8].

Research supported by the Advanced Digital Sciences Center (ADSC) under a grant from the Agency for Science, Technology and Research of Singapore. ADSC is a related organization in Singapore of the College of Engineering at the University of Illinois at Urbana-Champaign.

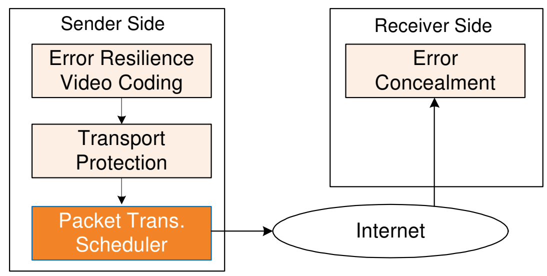

Fig. 1. Integrating our proposed *packet transmission scheduler* in real-time video transmissions over the public Internet.

By combining different types of existing techniques, system designers have large flexibility in compressing and transmitting real-time videos in the lossy Internet. Unfortunately, real-time video presentation quality today is still rather unsatisfactory, especially in the presence of extreme network behaviors. For instance, transmissions in some inter-continental connections or wireless links can experience over 30% loss rate and over 250 ms one-way delay [9]. In this case, unconcealable losses are unavoidable, even though good design choices have been made in combining the best existing techniques. Due to heavy inter-frame dependencies in predictive video coding, the loss of a single video frame may result in perceptible artifacts over a number of subsequent frames.

Instead of proposing yet another technique from the previous categories, we propose a novel component to augment the design space explored by video application designers. As shown in Figure 1, we introduce a scheduler that schedules transmissions of video packets of different priorities in order to improve robust video transmissions over the lossy Internet. Our rationale is that uneven packet transmission rates (UPTR) are allowed in the Internet. Experimental results have shown that the instantaneous packet transmission rate (to within some maximum) can fluctuate around an average rate, without significantly affecting the average loss rate and delay in transmissions [9]. Although the Internet does not guarantee in-order delivery, those bursty transmissions sent earlier will have a high chance to arrive ear-

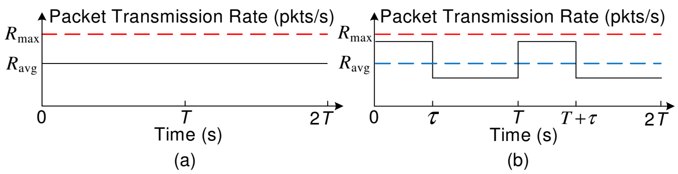

Fig. 2. A comparison of a) the traditional FPTR fixed at a constant packet rate R_{avg}, and b) our proposed UPTR that exploits two different packet rates allowed in the Internet: the maximum tolerable rate $R_{\max}$ and the average rate R_{avg}.

lier at receivers. With UPTR, the landscape of Internet-based real-time video transmission can be changed: some previous methods considered inappropriate or suboptimal can be noticeably enhanced, while others that are widely recognized can be further improved.

Motivated by the UPTR concept, this paper makes two major contributions. After presenting the concept in Section 2 and comparing it to the fixed packet transmission rate, we propose a novel real-time video coding and transmission scheme in Section 3 that out-performs existing approaches. Second, we exploit in Section 4 UPTR for fast transmission of source and redundant FEC packets of inter-coded frames that enable more packets to meet the end-to-end delay requirements.

2. UPTR: UNEVEN PACKET TRANSMISSION RATES IN THE INTERNET

Unlike circuit-switched networks, the Internet is an interconnection of packet-switched networks. This means that the Internet does not enforce a fixed packet transmission rate (FPTR). This flexibility, however, has been overlooked in many real-time designs, especially in video transmissions.

Although the Internet allows variable packet transmission rates in principle, it does have two constraints on the admissible rates in practice. Based on experimental results in the PlanetLab [9], two observations have been made on packet transmissions in the Internet. First, to achieve a stable packet loss rate and average delay, the packet transmission rate averaged over a long period of time should be kept no larger than R_{avg} (say 30 packets/s). Second, without incurring appreciable losses and delays, the instantaneous rate can fluctuate around the average rate to within some maximum $R_{\max}$ (say 50 packets/s) for a short interval. This behavior is due to buffers in intermediate routers that can smooth out irregular arrivals across different connections. Figure 2 illustrates the difference between FPTR and UPTR. Specifically, FPTR fixes the transmission rate at R_{avg}, whereas UPTR sends packets at a bursty rate close to $R_{\max}$ for short intervals, say $[T, T+\tau]$, while regulating the average rate to no larger than R_{avg} in any period T.

Independent of any specific video coding algorithms, UPTR is rather general and can be applied to improve real-time video transmissions in the Internet. The availability of bursty transmissions is beneficial in at least two scenarios. 1) The excess packets sent in the bursty interval τ can be used to carry quality-critical information, which is not intended for immediate use but will only be needed τ sec later. 2) Some packets can be sent earlier in a bursty fashion, as long as they do not carry time-critical source information. In line with this reasoning, we propose two design instantiations in the next sections and demonstrate the strength of UPTR for improving real-time video quality delivered in the Internet.

3. UPTR APPLICATION 1: OVERLAPPED TRANSMISSIONS OF INTRA-CODED FRAMES

To stop error propagations due to unconcealed video packets, inserting intra-coded (I) frames periodically appears to be an effective means. However, the method is widely dismissed in conversational video applications because the traditional FPTR assumption forces multiple subsequent inter-coded (P) frames that depend on the current I frame to be skipped, thereby causing a temporally frozen screen at receivers. For this reason, most existing solutions resort to intra-MB updates [3, 8] done either randomly or in a R-D optimized fashion. Unfortunately, the highly dynamic nature of the Internet renders the selection of optimal coding nontrivial. It becomes even more challenging for live video conferencing applications with a real-time requirement. Moreover, as discussed in Section 1, error propagation still cannot be completely avoided even with optimized intra-MB updates.

In this section, we propose a novel real-time video coding and transmission technique by exploiting UPTR. The key idea is two-fold. First, we utilize the bursty packet interval in UPTR to overlap the transmission of the quality-critical I frame with the usual P frames in the packet stream. To extend the real-time constraint of the current I frame, the concurrent P frames delivered during the overlapped bursty interval [see Figure 3(a)] do not depend on the current I frame. Instead, they are designed to reference the previous I frame sent ahead of time, also sent in an overlapped transmission manner. Clearly, our proposed method does not need to stop the subsequent P frame transmissions whenever an I frame is sent. This ensures smooth video playback without the displeasing frozen-screen effects. In addition, because the real-time constraint for I frames has been relaxed, better error protection (such as FEC) can be afforded for these frames, under the same end-to-end delay constraint. When compared to intra-MB updates that treat each video frame equally, our new scheme effectively enables unequal error protection for video packets of different priorities over the error-prone Internet.

The second key idea is to predictively encode each P frame in the same period (say $[\tau, T+\tau]$ in Figure 3) while using a common I frame as the reference. Removing the sequential dependency previously existing between two successive P frames, such a design will eliminate error propagation between P frames, while achieving better trade-offs between video coding efficiency and error resilience. In comparison to video coding schemes that force all video frames to be intra-coded (such as Motion-JPEG), our method removes sequential prediction dependency and gains desirable error resilience without substantially compromising coding efficiency. On the other hand,

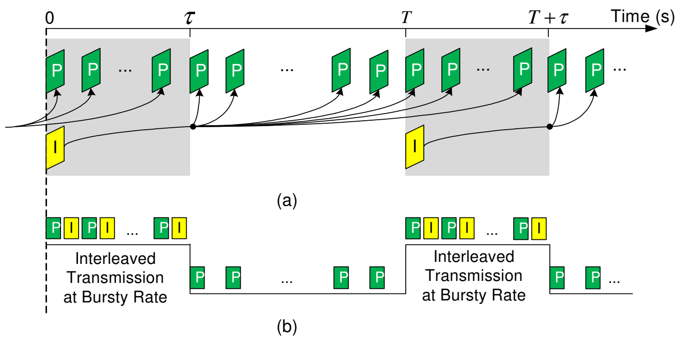

Fig. 3. Proposed real-time video coding and transmission method based on UPTR. a) Prediction dependency structure and overlapped transmission of I frames. b) Scheduling the transmission of packets at two different rates that match the new video coding strategy.

in comparison to the traditional sequential P frame coding, the coding efficiency of the proposed method drops gracefully, as the prediction performance of the common I frame, though temporally distant, is still good enough. This is true because video conference applications feature head-and-shoulder scenes with limited motions, typically captured against a stationary background. More importantly, perceived image distortions caused by error propagation is arguably far more disturbing than limited quality degradation due to the proposed coding scheme.

Next, we present the details of our proposed video coding and transmission method based on UPTR. There are two parameters for configuring the scheme: T, the I frame period, and τ, the overlapped bursty transmission interval. In this paper, we empirically set T to 1 sec, as a smaller value will leave insufficient IP packets for P frame coding and protection (constrained by R_{avg}). However, a larger value is not preferred because more P frames will have to reference a temporally distant I frame. The duration of the bursty transmission interval τ depends on two user-specified parameters: F, the target video playback frame rate, and N_I, the total number of packets allocated to encode and protect an I frame. Further, as shown in Figure 3(b), the transmission of packets containing I frames should always give way to the concurrent P traffic because the real-time constraint on P frames is not extended. Taking all these into account as well as the double-rate bounds in UPTR, the minimum value for τ is as follows.

$$\tau = \left\lceil \frac{F \cdot N_I}{R_{\max} - R_{\text{avg}} + N_I} \right\rceil \cdot \frac{1}{F}\,, \qquad (1)$$

where $\lceil x \rceil$ is the ceiling function of x. We will discuss how to select N_I under high packet loss conditions in Section 5.

Below, we present an approach that significantly reduces N_I^S, the number of packets needed for encoding the source information of an I frame, where $N_I^S \leq N_I$. The basic idea is as follows. Assuming the very first I frame has been successively received,[1] all subsequent I frames can be inter-coded by using the first I frame as the temporal reference. We call this new coding mode "inter-I mode" to differentiate it from the previous "intra-I mode." Our experiments show that, with a similar coding distortion, the new inter-I mode can reduce the packets transmitted by 30% to 60% for video sequences of different scene complexity. This normally results in less than 5 packets for encoding a CIF-sized I frame, namely, $N_I^S \in [1, 5]$. The advantage of reducing N_I^S is obvious: either better protection can be made for I frames under the same N_I, or more packets can be allocated to P frames by keeping the same redundancy degree for I frames. The latter may be less important because the protection of P frames is not as demanding as in traditional methods after removing sequential prediction dependencies.

4. UPTR APPLICATION 2: FAST TRANSMISSIONS OF FEC PACKETS OF INTER-CODED FRAMES

To provide acceptable one-way video quality in the presence of high packet losses, a sufficient amount of redundant information has to be sent to protect the source information. In this case, the end-to-end delay constraint Γ will have to be relaxed accordingly; otherwise, the additional protection information can hardly be received before the scheduled deadline. However, increasing Γ indiscriminately may result in an unfavorable perception of interactivity. To this end, it is important to identify a suitable playout delay Γ that achieves a good balance between one-way signal quality and the perceived interactivity. (Interested readers may refer to a recent study [9].) Besides delay-quality trade-offs, another advantageous direction is to investigate whether one-way video quality could be further improved under the same Γ and the same network condition, or equivalently, whether more redundant packets can have a higher chance to be received before the given deadline. In this section, we focus on addressing this question and again show UPTR as a positive application.

Consider a sequence of inter-coded (P) frames to be transmitted to a receiver under high packet losses. Suppose the source information of each P frame is encoded and encapsulated in N_P^S data packets, and FEC channel coding is applied to generate $N_P - N_P^S$ redundant packets [1]. These lead to a total of N_P packets for each P frame. To correctly reconstruct all the original N_P^S packets (or the full video frame), any N_P^S packets in the N_P-packet block must be received. Under a given end-to-end delay constraint Γ, it is clear that sending the FEC packets earlier will improve the likelihood of receiving the required N_P^S packets. This means that, for a P frame captured at t_i, more of its packets will have a higher chance of arriving before the scheduled playout time $t_i + \Gamma$.

As shown in Figure 4(a), the traditional FPTR assumption on the Internet restricts the packet transmission rate to R_{avg}. This means that, after starting the first packet transmission for P frame i at time t_i, every next transmission of its data or redundant packet has to wait for a minimum time of δ_a $(= 1/R_{\text{avg}})$ sec. Consequently, the addition of four redundant packets will

[1] This is easy to guarantee by a retransmission and acknowledgment mechanism in a short initial setup time before the real video conferencing session begins. In fact, Wu *et al*. [8] have also made the same assumption in their optimal coding-mode selection.

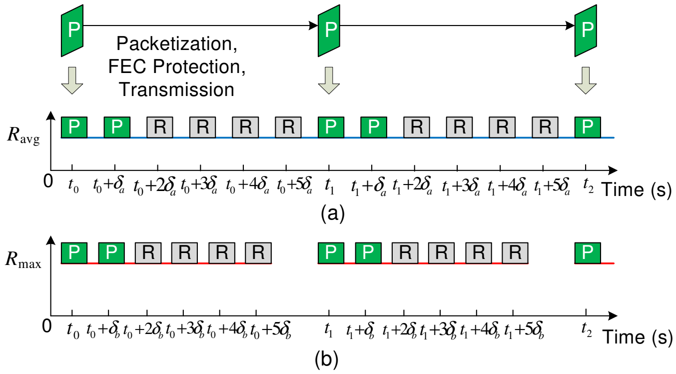

Fig. 4. Comparison of a) the traditional packet transmission scheme for source and redundant packets of inter-coded frames, and b) our proposed fast packet transmission scheme based on UPTR. As an example, each P frame is coded with two source packets ($N_P^S = 2$) and protected by four redundant packets ($N_P = 6$).

incur at least $4\delta_a$ sec before their potential loss-concealment features can be fully realized.

On the other hand, by exploiting the flexibility of the bursty transmission rate $R_{\max}$ in UPTR, it is possible to send the data and redundant packets of frame i earlier, except for the first data packet of frame i, which must always be anchored to time t_i. This allow the minimum waiting time between two neighboring packets to be shortened to δ_b $(= 1/R_{\max})$ sec [see Figure 4(b)]. By alternating between a bursty packet-transmission period and an idle period, the proposed scheme enables packets to be sent earlier, without exceeding the average packet transmission rate R_{avg} in the steady state. By reducing the unconcealed packet loss rate at the receiver, the new scheme will improve one-way video quality under the same end-to-end delay constraint.

5. EXPERIMENTAL RESULTS

We have conducted experiments to evaluate the performance of our two proposed schemes. In our experiments, we limit $R_{\max}$ to 50 packets/sec and R_{avg} to 30 packets/sec. As we only consider wire-line IP links in this paper, we assume MTU to be 1,500 bytes [6], which results in a payload of 1,460 bytes after excluding UDP and IP headers. We have implemented the proposed UPTR-based methods on the standard H.263+ video codec [10] because of the real-time complexity requirement, although they can be tested on other video codecs as well. H.263+ TMN-8 rate control has been enabled to control the resulting coding bit rates under the given packet resource budget. In our experiments, we have used the standard video sequences *Akiyo* and *Paris* in CIF format, which represent typical videoconference scenarios with distinct scene complexity and movements of participants.

To generate a common Internet delay trace file for our experiments, we have simulated network delays using a mixture of two independent uniform distributions. A mixture probability of 90% is assigned to the major uniform distribution, which takes delay values from $[100, 300]$ ms. The minor uniform distribution models high network jitters in a range of $[300, 600]$ ms with a probability of 10%. Figure 5 depicts a network delay trace simulated using this mixture model.

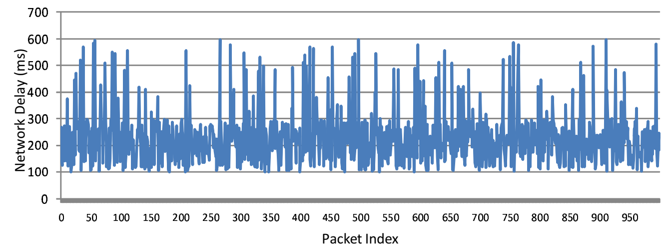

Fig. 5. Network packet delays generated by a probabilistic mixture model of two uniform distributions.

Table 1. Parameter settings in the experiments

	Input					Computed	
Sequence	F	Inter-I	N_I^S	N_I	N_P^S	N_P	τ (ms)
Akiyo	10	OFF	5	10	1	2	400
Paris	5	ON	3	10	2	4	400

Real-time video coding and transmissions. We first evaluate the proposed overlapped-I method presented in Section 3 under high packet-loss conditions. Together with an average random packet loss rate of 30%, Table 1 lists the major parameters used in our experiments. Using the same packet trace file, we compare our proposed method to a random intra-MB-updating scheme in the H.263+ codec. Clearly, more advanced intra-MB-updating schemes will improve error resilience of real-time video delivery, but the error propagation effects still cannot be fully avoided. We choose to report our results using this simple intra-MB-updating scheme because low-complexity implementations are often favored in real-time video conferencing and the scheme does not require modeling the fast changing and non-stationary behavior of the Internet. For fair comparisons, we have applied FEC protection to this traditional solution, under the same targets F and N_P^S and resource constraints R_{avg}. In addition, the intra-MB-updating rate was empirically set to 5 in the previous method, which forced an intra-MB refreshed every five coded macroblocks.

Figure 6 shows the PSNR values measured at the receiver for *Akiyo* and *Paris*, respectively. Similar to [6], PSNR values for those lost video frames (which repeat the last successful frame) are not calculated because penalizing those frames does not correlate well with the perceptual video quality preference. In contrast to the previous method that yields severe PSNR degradation in the presence of frame losses and error propagations, our proposed method provides more consistent PSNR performance with little fluctuation throughout the tests. Although removing sequential prediction dependencies in our proposed method incur a moderate PSNR drop on average (specifically 0.52 dB for *Akiyo* and 0.41 dB for *Paris*), our subjective quality evaluation shows that the playbacked videos obtained by our proposed method are clearly preferred over those by the previous method, which have disturbing error-propagation artifacts. To

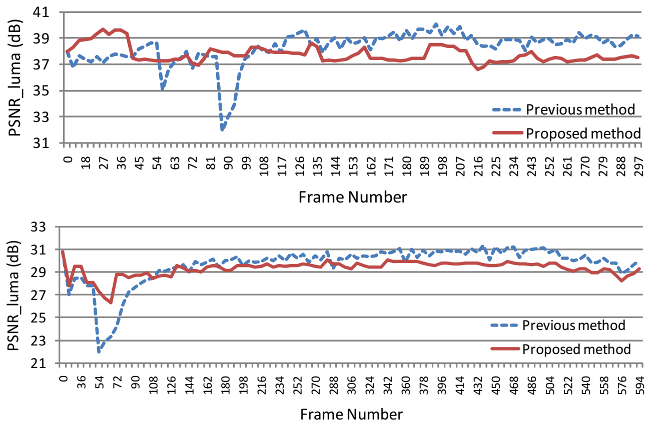

Fig. 6. PSNRs of the *Akiyo* sequence (top panel) and the *Paris* sequence (bottom panel) at the receiver obtained by the previous and our proposed methods.

(a) (b) (c)

Fig. 7. Sample frame (the 88th frame) of *Akiyo* by the previous method under a) a loss-free environment (PSNR = 37.5 dB) and b) a lossy environment (PSNR = 31.9 dB), and c) by our proposed method under a lossy environment (PSNR = 37.9 dB).

(a) (b) (c)

Fig. 8. Sample frame (the 55th frame) of *Paris* by the previous method under a) a loss-free environment (PSNR = 27.9 dB) and b) a lossy environment (PSNR = 22.0 dB), and c) by our proposed method under a lossy environment (PSNR = 27.3 dB).

demonstrate this visual effect, Figures 7 and 8 compare a sample frame reconstructed by the previous method and by our proposed method.

As stated in Section 3, we have empirically set the I-frame period T in our proposed method to 1 sec. To validate this choice, Figure 9 shows that video coding efficiency generally drops as T increases, although the changes in PSNR across T are minor. This result suggests that long-term temporal references perform equally well on video-conferencing sequences. However, more frequent insertion of I frames (without exceeding $R_{\max}$) can provide smoother videos in case of I frame losses, although the probability of I frame losses is considerably lower in our proposed design. For instance, the combination of $(N_I, N_I^S) = (10, 3)$ can lead to a target unconcealed full-frame rate of 0.16%, even in the presence of a high packet loss rate of

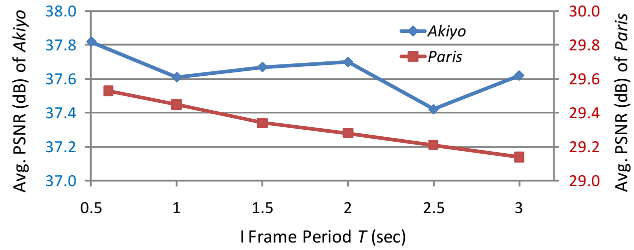

Fig. 9. PSNR of the proposed method at the encoder under different I frame insertion periods T. Tests were performed using the same parameter settings in Table 1, whereas T was varied.

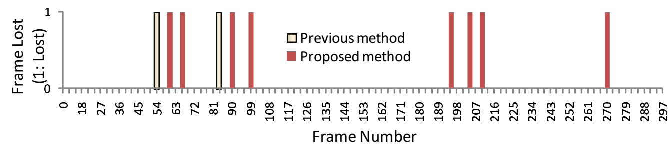

Fig. 10. Comparison of P frame losses tested on *Akiyo*.

30%.

Due to a different protection degree, the redundancy level of P frames in our proposed method is not as high as that in the previous method (for instance, (N_P, N_P^S) = (2,1) versus (3,1) for *Akiyo*, and (4,2) versus (6,2) for *Paris*). As a result, more P frame losses will occur in a video conferencing session (see Figure 10). However, the loss of non-reference P frames in our proposed method does not cause error-propagation artifacts observed in the previous method. Further, no jerky motion is perceptible in our decoded videos at receivers, as the loss of P frames occurs randomly, not in a bursty manner.

Fast packet transmissions. Next, we present our experimental results on evaluating our proposed fast packet transmission scheme in Section 4. We show how the long established intra-MB updating techniques can be further improved, when coupled with our proposed method. For this purpose, we have used the random MB updating scheme discussed in the last experiments as our baseline solution (termed "previous method" hereinafter). Note that other advanced intra-MB updating methods can also benefit from our proposed method.

In the experiment on *Akiyo*, the previous method was set with the following parameters: $F = 7.5$ fps, $N_P^S = 1$, $N_P = 4$, and an intra-MB updating rate of 5. The end-to-end delay constraint Γ is empirically set to 300 ms. A packet is, therefore, considered lost if it arrives later than the maximum expected time at the receiver. Besides this, an average random packet loss rate of 30% is still simulated in the experiments.

Based on common test conditions, our simulation has found that the proposed fast packet transmission scheme effectively reduces the late-arriving packet rate from 33.5% to 23.2%. This means that a video frame is more likely to be correctly concealed with more redundant packets received before the deadline. Figure 11 shows the simulation results in PSNR that compares the previous method with and without our proposed UPTR scheme included. The results convincingly illustrate the bene-

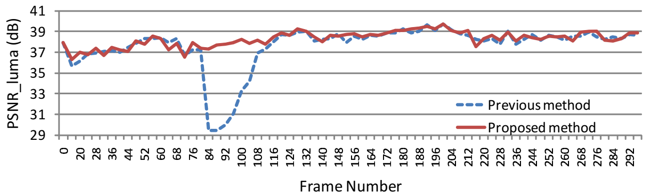

Fig. 11. PSNR of *Akiyo* at the receiver obtained by the previous method alone and the previous method with UPTR.

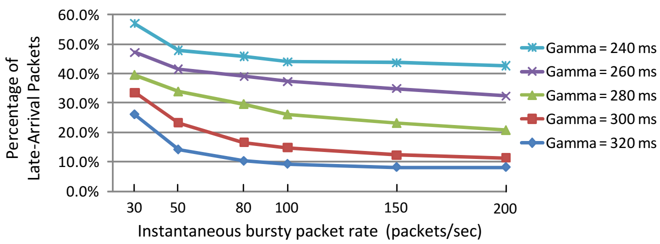

Fig. 12. Percentage of packets exceeding the end-to-end delay constraint Γ (Gamma in the figure), under different instantaneous bursty packet transmission rates $R_{\max}$.

fit of our proposed fast transmission scheme under high packet losses and delays.

Up to this point, we have limited $R_{\max}$ to 50 packets/sec. This is relatively conservative, considering that intermediate routers in the Internet can normally tolerate a much higher bursty rate. By increasing $R_{\max}$ from 30 to 200 packets/sec, Figure 12 shows that the fraction of late-arriving packets consistently drops, when given the same end-to-end delay constraint Γ. On the other hand, by exploiting UPTR, Γ can be decreased further, while a similar (or even reduced) fraction of late packets can still be achieved.

Packet allocation N_I. Lastly, we investigate the optimal operating point of N_I, where a constrained amount of IP packets need to be optimally allocated to I and P frames. Under a high packet-loss rate of 30%, Figure 13 plots the average unconcealed frame rate (*UCFR*) of the I and P frames as a function of N_I for two test cases. It is found that setting N_I to 9 or 10 leads to a good trade-off in these cases. In the future, we plan to study methods to dynamically adapt N_I using feedbacks from receivers.

6. CONCLUSIONS

In this paper, we have studied the scheduling of transmissions of video packets in real-time video conferencing using two packet sending rates. We have shown that our proposed UPTR approach can augment traditional designs in quest of robust video delivery and better video quality. Based on UPTR, we have proposed two techniques: the first overlapping the transmission of quality-critical I frames with prediction-dependency-modified P frames, and the second sending temporally flexible source and redundant packets earlier in order to improve the chance of concealing losses at receivers. Our results illustrate the effectiveness of our methods when compared to traditional approaches

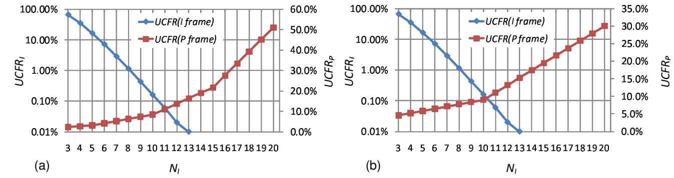

Fig. 13. $UCFR_{I/P}$ versus N_I. a) $F=5$, $N_I^S=3$, $N_P^S=2$. b) $F=10$, $N_I^S=3$, $N_P^S=1$. $UCFR_I$ is plotted on a logarithmic scale.

without UPTR under heavy loss conditions.

Our future work will focus on further improving the video coding efficiency of our proposed overlapped-I coding and transmission scheme, as well as adapting the parameters to various network conditions. In particular, we are currently implementing and evaluating the proposed UPTR approach on the H.264 video codec. We also plan to integrate UPTR with advanced error resilience and concealment methods.

7. REFERENCES

[1] D. Wu, Y. T. Hou, and Y.-Q. Zhang, "Transporting real-time video over the Internet: Challenges and approaches," *Proc. IEEE*, vol. 88, no. 12, pp. 1855–1877, Dec. 2000.

[2] A. K. Katsaggelos, Y. Eisenberg, F. Zhai, R. Berry, and T. N. Pappas, "Advances in efficient resource allocation for packet-based real-time video transmission," *Proc. IEEE*, vol. 93, no. 1, pp. 135–147, Jan. 2005.

[3] G. Cote and F. Kossentini, "Optimal intra coding of blocks for robust video communication over the Internet," *EUROSIP Signal Process. Image Commun.*, vol. 15, pp. 25–34, Sep. 1999.

[4] P. Baccichet, S. Rane, A. Chimienti, and B. Girod, "Robust low-delay video transmission using H.264/AVC redundant slices and flexible macroblock ordering," in *Proc. IEEE Int. Conf. Image Processing*, 2007, pp. 93–96.

[5] J. Rosenberg and H. Schulzrinne, "An RTP payload format for generic forward error correction," RFC 2733, 1999.

[6] S. Wenger, "H.264/AVC over IP," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 645–656, Jul. 2003.

[7] P. Salama, N. B. Shroff, and E. J. Delp, "Error concealment in encoded video," in *Image Recovery Techniques for Image Compression Applications*. 1998, Norwell, MA: Kluwer.

[8] D. Wu, Y. T. Hou, B. Li, W. Zhu, Y.-Q. Zhang, and H. J. Chao, "An end-to-end approach for optimal mode selection in Internet video communication: Theory and application," *IEEE J. Sel. Areas Commun.*, vol. 18, pp. 977–995, 2000.

[9] B. W. Wah and B. Sat, "The design of VoIP systems with high perceptual conversational quality," *Journal of Multimedia*, vol. 4, no. 2, pp. 49–62, Apr. 2009.

[10] M. Gallant, G. Cote, B. Erol, and F. Kossentini, *UBCs H.263+ public domain software*, in Official ITU-T Study Group 16 Video Experts Group Reference Codec, 1998.