On Rate-adaptive LDPC-based Cross-layer SVC over Bursty Wireless Channels

Yongju Cho[1], Jihun Cha[1], Hayder Radha[2], and Kwang-deok Seo[3]
[1] Electronics and Telecommunications Research Institute (ETRI)
Daejeon 305-700 - Korea
[2] Michigan State University
East Lansing, MI 48824, - USA
[3] Yonsei University
Wonju, Gangwon 220-710, - Korea
[e-mail: yongjucho@etri.re.kr, radha@msu.edu, kdseo@yonsei.ac.kr]
*Corresponding author: Kwang-deok Seo

Received June 15, 2012; revised July 30, 2012; revised August 17, 2012; accepted September 3, 2012; published September 26, 2012

Abstract

Recent studies have indicated that a significant improvement in wireless video throughput can be achieved by Cross Layer Design with Side-information (CLDS) protocols. In this paper, we derive the operational rate of a CLDS protocol operating over a realistic wireless channel. Then, a Rate-Distortion (R-D) empirical model for above-capacity Scalable Video Coding (SVC) is deduced to estimate the loss of video quality incurred under inaccurate rate estimation scenarios. Finally, we develop a novel Unequal Error Protection (UEP) scheme which leverages the characteristics of LDPC codes to reduce the distortion of video quality in case of typically-observed burst wireless errors. The efficacy of the proposed rate adaptation architecture over conventional protocols is demonstrated by realistic video simulations using actual IEEE 802.11b wireless traces.

Keywords: Cross-layer design, SVC video adaptation, LDPC code, wireless video

This work was supported by the IT R&D program of MKE/ETRI [12ZI1140, HCI-based UHD Panorama Technology Development].
http://dx.doi.org/10.3837/tiis.2012.09.017

1. Introduction

In many wireless environments, deteriorated link conditions cause frequent bit-corruptions. These corrupted packets cause checksum failures and packet drops at wireless receivers. To reduce packet losses at the receivers, many recent efforts utilize cross-layer protocols that do not discard corrupted packets [1][2][3][4][5][6][7][8][9]. Consequently, two classes of wireless multimedia protocols have emerged [2]: (i) Cross-Layer-Design (CLD) protocols which relay corrupted packets to higher layer for further processing; (ii) conventional (CON) protocols which drop any packet that has one or more residue errors. Prior studies have shown that a significant improvement in wireless video throughput can be achieved by CLD [2][3][4][5]. Furthermore, it has also been exhibited that side information, which are already available from IEEE 802.11 compliant packets, is quite valuable for providing channel state information and modeling of the underlying (effective) video channel. This side information includes Signal to Silence Ratio (SSR) indicators and MAC-layer checksum, both of which can be used as parameters for channel estimation [8]. This form of CLD protocols that utilize side information have been referred to as Cross-Layer-Design with Side information (CLDS) protocols in prior literature [1][2], [8][9], [12].

Despite the demonstrated benefits of CLDS, full utilization of CLDS-based video streaming with appropriate rate adaptation remains unexplored. An important missing link in deploying a rate adaptive CLDS architecture is the ability to accurately estimate and predict the wireless channel capacity in real-time. Capacity estimation for wireless channels is a challenging problem because wireless link conditions fluctuate frequently and significantly due to interference, fading, multi-path effects, and mobility.

In this paper, we develop three novel schemes to realize an Optimal Rate Prediction Architecture under CLDS protocols ($ORPA_{CLDS}$) in a practical rate adaptation multimedia application. The developed three schemes include: 1) Operational rate that incorporates the performance drop of a practical channel code which is inferior to an ideal code due to finite length and other design limitations. 2) A Rate-Distortion (R-D) empirical model for above-capacity video. This model provides a PSNR value of video bitstream that is coded at over-estimated rates (or above capacity); and hence, we can complete $ORPA_{CLDS}$ for a realistic environment. 3) An Unequal Error Protection (UEP) scheme which leverages the characteristics of Low Density Parity Check (LDPC) codes to efficiently reduce the degradation of video quality in case of bursty error on a wireless channel. Our trace-driven wireless video simulation results show that $ORPA_{CLDS}$ provides outstanding rate prediction performance in a realistic environment.

The rest of this paper is organized as follows. Section 2 describes our wireless trace collection setup and then performs some preliminary analysis on the collected data. Section 3 completes $ORPA_{CLDS}$ by motivating and developing the proposed schemes. Section 4 and 5 respectively evaluate the performance of the proposed framework and summarize key conclusions of this work.

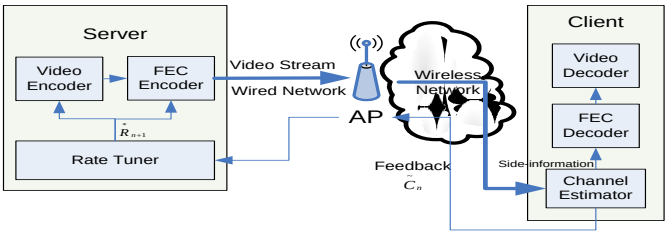


Fig. 1. The architecture of the proposed rate adaptation

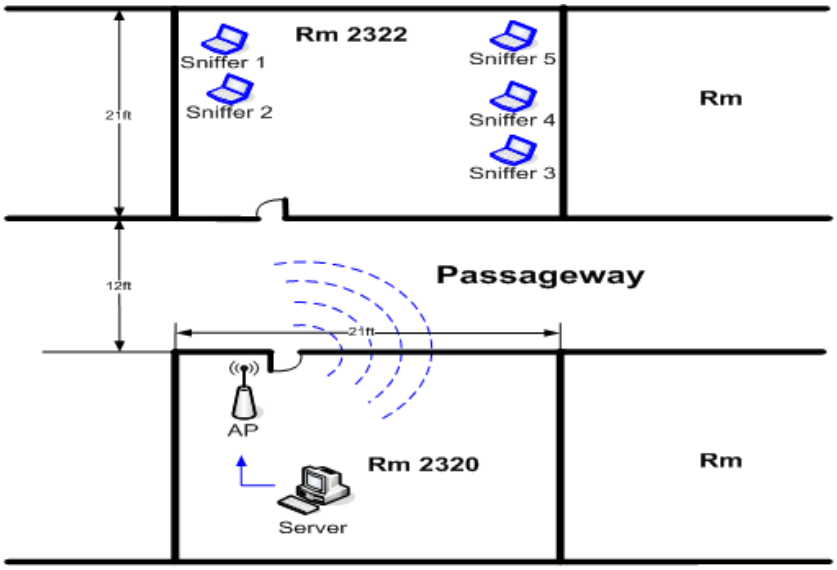


Fig. 2. Topologies used for wireless trace collection

2. Wireless Traces Collection and Analysis

2.1 Data Collection

For this study, we collected error traces simultaneously on five 802.11b wireless receivers. The receivers were located at different places in a research lab, while the access point (AP) was placed in a room across a hallway from the receivers to simulate a realistic classroom/office setting, as shown in **Fig. 1** and **Fig. 2**. In this setup, the receivers used the DLink DWL-650 WLAN PCMCIA card based on the Prism2 chipset together with the linux-wlan-ng-0.1.14-pre3 driver on a PC with Linux RedHat 7.1 OS. The receivers' MAC layer device drivers were modified to capture corrupted packets. Each experiment comprised of one million packets with a payload of 1,000 bytes each, i.e., each trace has approximately 1 GB of data, which corresponds to approximately 4 and half hours of error trace collection when packets are transmitted at 500 Kbps.

A wired sender was used to send multicast packets with a predetermined payload on the wireless LAN; multicasting disabled MAC layer retransmissions. In addition to a packet's header and payload information, we logged signal to silence ratio (SSR) for each packet. The sender used different transmission rates ranging from 500 Kbps to 1 Mbps for each experiment. At the physical layer, the auto rate selection feature of the AP was disabled and for each

experiment the AP was forced to transmit at a fixed data rate. Each trace collection experiment was repeated multiple times at 2, 5.5 and 11 Mbps physical layer data rates and at different times of day. Due to brevity, we skip the detailed study of BER behavior at different SSR values and refer the reader to Section 2 in [11].

2.2 Average Statistics of the Traces

Table 1 provides some statistics of the traces collected for this study. Since the physical layer robustness decreases with an increase in data rate, the average packet error rate increases with an increase in the physical layer data rate. In particular, the average packet error rate increases from approximately 10% at 5.5 Mbps to almost 40% at 11 Mbps. Since the wireless receivers were placed at different locations, the receivers experienced different packet error rates. The overall minimum and maximum error rates in **Table 1** outline that the receivers under consideration were experiencing both good and bad link conditions.

The average, minimum and maximum SSR values are also shown in **Table 1**. Note that the minimum SSR value is zero at all three data rates. From a prior analysis [8], [11], we know this SSR range is of interest for the protocols considered in this paper.

The relationship between SSR values and the channel error rate is also shown in **Table 1**. It is easily observed from the second column of **Table 2** that packet error rates increase drastically with a decrease in SSR values. In particular, the packet error rate increases by approximately 18% as the SSR decrease from 26 dB to 20 dB. Similarly, there is a packet error rate increase of about 41% between SSRs 13 and 20. Interested readers are referred to [11] for a more detailed discussion of BER and SSR behavior of the channel.

Table 1. Statistics of traces used in this study

Phy. data rate (Mbps)	Avg. PER	Min. PER	Max. PER	Avg. SSR (dB)	Min. SSR (dB)	Max. SSR (dB)
2	5.97%	0.75%	14.31%	14.75	0	34
5.5	9.79%	0.61%	22.74%	15.27	0	32
11	39.5%	10.99%	77.83%	16.51	0	35

Table 2. Error statistics for varying SSR values at 11 Mbps

SSR (dB)	Average Packet-Error Rate	BER of all packets (error-free & corrupted)	BER of corrupted packets
5	0.701	0.0253	0.0361
13	0.6248	0.0157	0.0251
20	0.2166	0.0048	0.0223
26	0.0384	0.0023	0.0591

3. Prediction Based Rate Adaptation

In this section, first $ORPA_{CLDS}$ is formulated to be deployed in a practical video streaming application by exploiting the three main contributions of this paper: operational rate, an R-D empirical model for above-capacity SVC and an LDPC-based UEP scheme.

3.1 A Rate Adaptation Architecture for Wireless SVC

The architecture of a multimedia streaming application depends heavily on a network on which it operates. Therefore, it is essential to define our proposed architecture for rate

adaptation. The proposed architecture consists of a server and a client that runs over a wireless network. In the given architecture, both a server and a client are designed to support source and channel rate adaptation. **Fig. 1** describes such architecture. The client supports CLDS protocols that leverage residue-error-process and side information, which can be relayed to an Forward Error Correction (FEC) decoder for soft-decoding [14], and to estimate the current channel capacity, $\tilde{C}_n$, for a block of packets (or time-window for m packets). The current channel capacity, which is estimated by the channel estimator with the entropy of the residue error process, $H_b(\varepsilon)$, is then transmitted to the server as feedback for rate adaptation. Using the feedback, $\tilde{C}_n$, the rate tuner at the server predicts operationally optimal source and channel rates, $\overset{*}{R}_n$, for the next block of multimedia packets to be transmitted. Note that for FEC encoding, different redundancy parameters such as α_i and α_P, are used for different types of video frames to protect more on more important information. The details are described in Section 3.5.

For this study, we consider the following configurations of video and channel codes: (i) a video test sequence (foreman) is encoded using Scalable Video Coding (SVC) [21] to easily adapt source bitrates by dropping packets rather than re-encoding for different rates.

SVC is the extension of ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC standard and its characteristics is to provide spatial, temporal and quality scalability with a video bitstream. A SVC bitstream can consist of base and one or more enhancement layer for spatial, temporal or quality aspect [21][22]. Higher quality video can be achieved by decoding both base and enhancement layers whereas lower quality of video can be experienced only with base layer by dropping of packets due to a bandwidth limitation. It is a strong point of SVC that one bitstream can be dynamically utilized to adjust to network bandwidth change compared to others that only coarse scalability can be provided. To that end, SVC can be efficiently employed in especially wireless networks where link conditions fluctuate frequently and significantly due to interference, fading, multi-path effects, and mobility.

The encoding configuration is as follows: two-layered spatial scalability and three-layered Fine Granularity Scalability (FGS); 30 frame-per-second (fps) only with I and P frames; and Group of Pictures (GOP) of 64. Note that the above encoding configurations were used to reduce the computational complexity in wireless mobile terminals, which are the main focus of our study. (ii) LDPC code [19] is used as a FEC code. Note that since the LDPC code, which is used in this study, does not achieve channel capacity, we have to consider the operational rate that is strictly less than channel capacity for reliable communication.

In addition, the time-window for the experiment is selected such that the variation in link-quality from one time window to the next is minimized. Allan Variance is a measure which allows us to methodically determine such an optimal size of the window [17], and it is computed as

$$AVAR^2(\tau) = \frac{1}{2(k-1)} \sum_{n=1}^{k} (C(\tau)_{n+1} - C(\tau)_n)^2 \tag{1}$$

where $AVAR^2(\tau)$ is the Allan Variance as a function of averaging time, τ; C_n $\left(=1-\frac{1}{m}\sum_{i=1}^{m} H_b(\varepsilon_i)\right)$ is the average channel capacity of the measurement in time window n; and k is the total number of time windows.

For all traces that we have considered, the Allan Variance of BER is minimal for a time-window size of 5 seconds as shown in **Fig. 3**. Hence, in this study we use the time window with a size of 5 seconds.

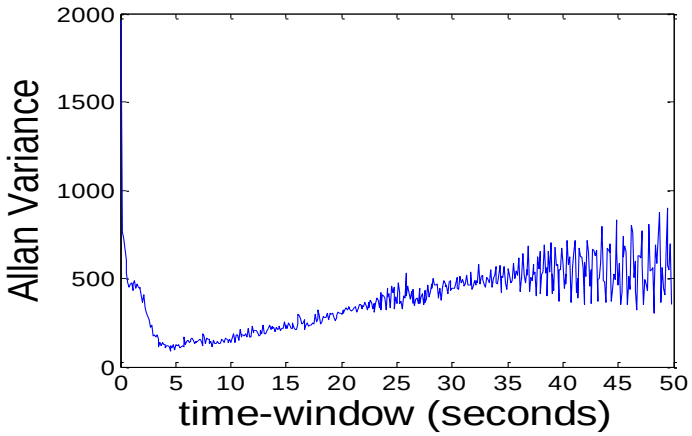


Fig. 3. Allan Variance as function of time window

3.2 Optimal Rate Prediction

- Channel Estimation and Prediction

As shown in **Table 2**, the BER of the channel changes with respect to the channel SSR. In general, we observed a non-linear relationship between SSR and BER. Here, we employ a BSC model, which utilizes each SSR of a packet to estimate the BER over a given block of packets. While we acknowledge that the BSC model is somewhat simplistic for the present problem, we observed that this simple model can provide quite accurate BER prediction; and more importantly it provides a conservative estimate for the channel capacity due to the lack of a memory model in the channel. (The impact of channel memory on the proposed scheme is the subject of an ongoing work and is beyond the scope of this paper.). We partitioned the collected traces into training and test data. Next, with training data we define bins over the entire SSR range and determine the average BER for each bin. Note that we regard the average BER as the crossover probability in Binary Symmetric Channel (BSC) model [8], [9], [10]. Hence, a packet renders a BER estimate according to its SSR. During the testing phase we use these estimates of BER to determine channel capacity over the time window (or the number of packets, *m* for a given transmission rate) that is defined in Eq. (1). The channel capacities are calculated as follows:

$$\tilde{C}_n^{CLDS} = 1 - \frac{1}{m}\sum_{i=1}^{m} H_b(\tilde{\varepsilon}_i) \tag{2}$$

$$\tilde{C}_n^{CON} = 1 - \frac{1}{m}\sum_{i=1}^{m} Z_i = 1 - PER \tag{3}$$

where $\tilde{\varepsilon}_i$ represents the channel BER estimate for packet i, Z_i is a binary variable representing the status of the checksum of packet i ($Z_i = 1$ if checksum fails). Eq. (2) is the capacity estimate $CLDS$ computed based on the average of the instantaneous per-packet error-process entropy in a block of m packets, and Eq. (3) is the estimate of channel capacity under CON protocols.

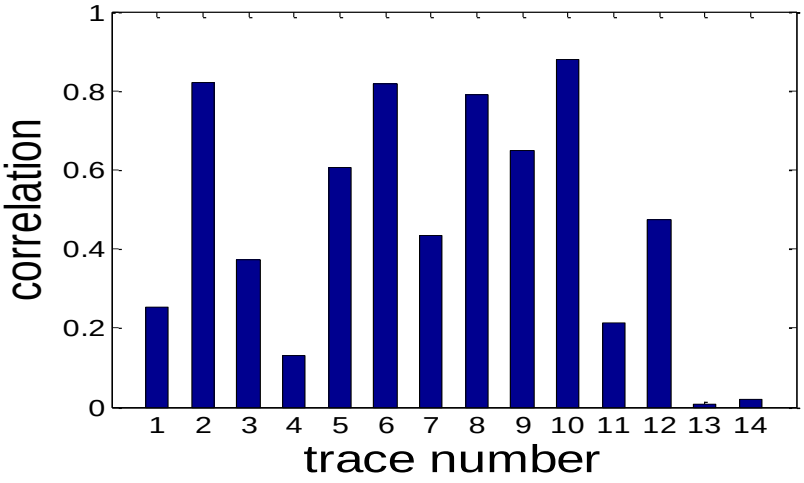


Fig. 4. Temporal correlation in channel capacity for 11 Mbps traces (traces from 1 to 5 collected with transmission rate at 500, 6 to 7 at 750, 7 to 10 at 900, and 11 to 14 at 1024 Kbps).

Prior studies have indicated that bit-errors have high temporal correlations, especially in 802.11b wireless networks [10][11][12][13]. In **Fig. 4**, the correlation coefficients of a number of traces are shown and calculated on the basis of the channel capacity process. The correlation coefficient is computed as

$$\rho = \frac{E[C_n C_{n+1}] - E[C_n]E[C_{n+1}]}{\sqrt{\text{var}[C_n] \cdot \text{var}[C_{n+1}]}}, \tag{4}$$

where $E[\cdot]$ and $\text{var}[\cdot]$ are the sample mean and the sample variance functions. **Fig. 4** clearly exhibits the existence of temporal correlation that is non-negligible in all traces, and it is quite significant in most traces. This correlation can be taken advantage of to predict the channel capacity for the next time-window. We exploit this correlation by using the channel capacity estimate of the current packet as an estimate for the next packet's channel capacity:

$$\hat{C}_{n+1} = \tilde{C}_n, \tag{5}$$

which deduces the channel prediction error[1], *e*.

Although we have considered other optimum predictors (e.g., Yule-Walker [16]), our simulation results demonstrate that the above simplistic prediction in conjunction with the optimal rate tuning (described below) provides significant improvement over conventional protocols. Furthermore, the prediction performance of Eq. (5) is very similar to the optimum Yule-Walker predictor [1]. Consequently, we focus the remainder of this paper (in the context of the proposed rate prediction architecture) on the predictor determined by Eq. (5) due to its minimal complexity.

- Optimal Rate Tuning

[1] $e_{n+1} = C_n - \hat{C}_n = \tilde{C}_n - \hat{C}_n = \hat{C}_{n+1} - \hat{C}_n$

It is essential that we need to adhere to a rate that is strictly below channel capacity to avoid excessive packet drops. Therefore, if the predicted channel capacity is directly employed, there is a good likelihood that the predicted capacity (as a random variable) may exceed the actual channel capacity. A natural workaround to this problem is to make the predicted capacity more conservative by subtracting a small offset Δ from the channel capacity prediction, $\hat{C}_n$. Such a strategy can, however, result in considerable long-run bandwidth wastage, and therefore judicious selection of the Δ parameter is extremely important. We propose the following formula to find the "optimal" value of Δ (leading in turn to the optimal video rate) such that the average video peak signal-to-noise ratio (PSNR) over some period of time (or over a set of blocks of packets) is maximized:

$$\overset{*}{\Delta}_n = \arg\max_{\Delta}\left[\frac{1}{N}\sum_{n=1}^{N}(I(C_n > \hat{C}_n - \Delta)) \cdot Q((\hat{C}_n - \Delta) \cdot T)\right] \quad \text{and} \quad \overset{*}{R}_n = \hat{C}_n - \overset{*}{\Delta}_n \tag{6}$$

where C_n and $\hat{C}_n$ are the actual and predicted channel capacities, and $I(\cdot)$, $Q(\cdot)$ and T respectively represent an Indicator function[2], an R-D (video quality) function and a transmit rate[3]. Thus the above objective function assumes a rather simple binary quality-indicator that forces the estimated PSNR value to zero when the rate exceeds the capacity. Note that based on the present objective function defined in Eq. (6), the "optimal" rate can be computed only when the overall statistics and the actual channel capacity values are available. Since the optimal rate determined using Eq. (6) is based on the entire trace, it cannot be utilized in real-world applications. We resolve this issue by observing that the probability distribution of the channel prediction error process, *e*, is very close to a Gaussian distribution, $N(0,\sigma_e^2)$ as shown in **Fig. 5**.

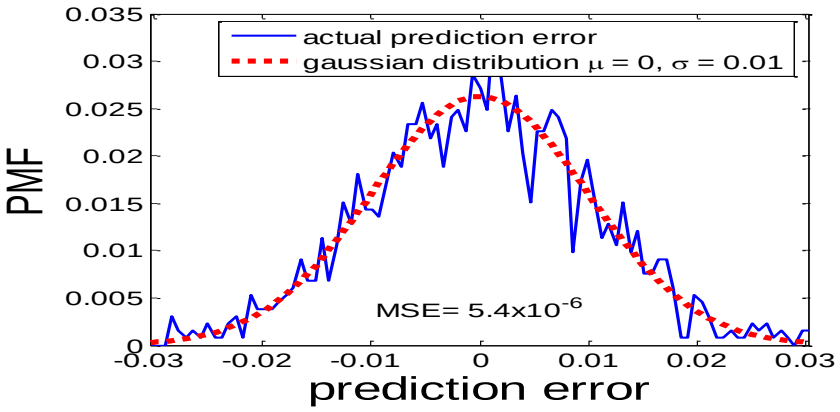


Fig. 5. The channel prediction error process (*e*) resembles the probability distribution of Gaussian

---

[2] $I(\zeta) = \begin{cases} 0 & \text{if } \zeta \text{ is not satisfied} \\ 1 & \text{otherwise} \end{cases}$

[3] $T = \frac{pkts\ in\ \tau \cdot pkt\ size}{\tau}$ *where* τ *is time – window* (bits per second), and hence $\overset{*}{R} \cdot T$ is the effective transmission rate which is actually transmitted for the underlying channel.

Furthermore, we take into account the normal distribution, which is well known to have the highest entropy, and hence provides the most conservative estimate. Based on the Gaussian assumption for the prediction error, this leads to an expression that replaces the indicator function with the normal distribution function. The predicted channel capacity can then be optimally tuned by finding a rate $\overset{*}{R}_n$ to maximize $PSNR_n$ as

$$\begin{aligned}\overset{*}{R}_n &= \arg\max_{R_n\,(0\le R_n\le 1)} \; Q(R_n\cdot T)\cdot \Pr\{C_n \ge R_n\} \\ &\quad + Q'\left(\left|R_n - C_n\right|\cdot T\right)\cdot \Pr\{C_n < R_n\} \\ &= \arg\max_{R_n\,(0\le R_n\le 1)} \; Q(R_n\cdot T)\cdot \Pr\{e_n \ge -\Delta_n\} \\ &\quad + Q'\left(\left|R_n - C_n\right|\cdot T\right)\cdot \Pr\{e_n < -\Delta_n\} \\ &= \arg\max_{R_n\,(0\le R_n\le 1)} Q(R_n T)\cdot \frac{\int_{R_n-\hat{C}_n}^{1-\hat{C}_n} \frac{1}{\sqrt{2\pi}\sigma_e}\exp\left(\frac{-e_n^{\,2}}{2\sigma_e^2}\right)de}{\int_{-\hat{C}_n}^{1-\hat{C}_n} \frac{1}{\sqrt{2\pi}\sigma_e}\exp\left(\frac{-e_n^{\,2}}{2\sigma_e^2}\right)de} + Q'(\left|R_n - C_n\right|)\cdot \frac{\int_{-\hat{C}_n}^{R_n-\hat{C}_n} \frac{1}{\sqrt{2\pi}\sigma_e}\exp\left(\frac{-e_n^{\,2}}{2\sigma_e^2}\right)de}{\int_{-\hat{C}_n}^{1-\hat{C}_n} \frac{1}{\sqrt{2\pi}\sigma_e}\exp\left(\frac{-e_n^{\,2}}{2\sigma_e^2}\right)de}\end{aligned} \tag{7}$$

where $Q(\cdot)$ is the R-D (quality) function of the video sequence (as a function of total number of bits used to code the sequence); $Q'(\cdot)$ is the quality function of the video sequence for rates above capacity (or video quality distortion function); and $\int(\cdot)$ represents the probabilities of rate below capacity (the first term in Eq. (7)) and of rate exceeding capacity (the second term in Eq. (7)) based on the distribution of channel prediction error process. Thus, the predicted channel capacity can be fully utilized and the video quality can be optimized when the product of the R-D function and the probability distribution of the channel prediction error process is maximized.

3.3 Operational Rate

It is important to notice that a rate in Eq. (7) is based on the ideal channel code that achieves channel capacity. When a realistic channel code is employed we have to consider the operational rate[4], R^{op} which is strictly less than channel capacity for a reliable communication. Therefore, the rate in Eq. (7) should be adjusted in conjunction with a specific channel code. We formulate the operational rate as:

$$R^{op} = 1-\alpha\cdot H(\varepsilon)\ ,\ 1\le\alpha\le\frac{1}{H(\varepsilon)} \tag{8}$$

where ε is the actual channel BER and α is the redundancy parameter. Note that the channel capacity for the considered BSC channel can be estimated as $C = 1-H_b(\varepsilon)$ [20]. For reliable communication (i.e. distortion free communication) the operational rate has to satisfy $R^{op}\le C$. While it is theoretically possible to satisfy this constraint, in practice the performance of a code

[4] The operational rate in this study is equivalent to the achievable channel coding rate, $\frac{\log(M)}{n}$ for $(n,\log(M))$ channel code, which embodies inferior performance of a practical code to an ideal code.

is inferior to the theoretical predictions because of finite length and other design limitations. We capture this performance drop by introducing the parameter α. Thus, we use a stricter constraint $R^{op}+(1-\alpha)H(\varepsilon)\le C$, where a hypothetically optimal code can be represented by $\alpha=1$ [12].

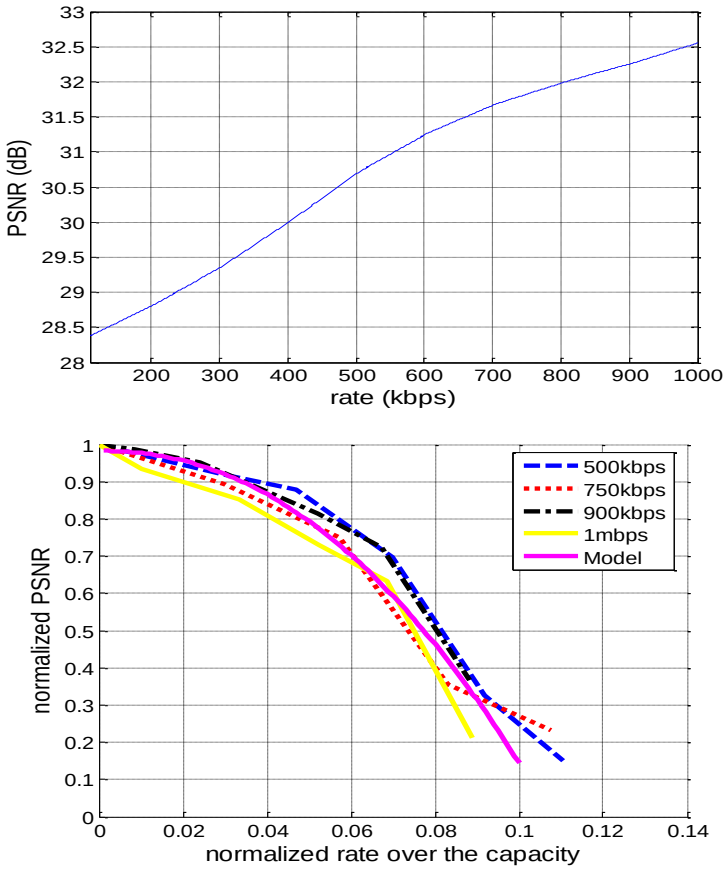


Fig. 6. (a) The R-D (quality) function of the SVC test (foreman) video sequence, and (b) the empirical model for the video quality distortion.

As explained above, the value of α plays a critical role in rate selection. Therefore, in this study, we first deduce a suitable value for this parameter. Analytical deduction of α requires us to consider finite length analysis of LDPC codes. This is a challenging and reasonably open area of research. In this study, we provide a practical solution by empirically evaluating the value of α. For this purpose, we conducted a comprehensive set of experiments with LDPC channel codes and SVC. Ten experiments in randomly selected time windows for each trace were conducted with a SVC bitstream of foreman consisting of 1200 packets. We observed that for $\alpha=2.3$ most of the video packets with size of more than 1000 bits are decoded successfully as shown in **Fig. 7**. Henceforth, we use this empirically deduced value for all experimental evaluation. Note that in this study the operational rate is the maximum achievable rate for the given channel code and for the underlying channel. Thus, when the operational rate is employed in Eq. (7), the predicted channel capacity can be operationally and optimally tuned as

$$R_n^{op*} = \arg\max_{R_n^{op}\,(0\le R_n^{op}\le 1)} Q(R_n^{op}T)\cdot \frac{\int_{R_n^{op}-\hat{C}_n^{op}}^{1-\hat{C}_n^{op}} \frac{1}{\sqrt{2\pi}\sigma_e}\exp\left(\frac{-e_n^{\,2}}{2\sigma_e^2}\right)de}{\int_{-\hat{C}_n^{op}}^{1-\hat{C}_n^{op}} \frac{1}{\sqrt{2\pi}\sigma_e}\exp\left(\frac{-e_n^{\,2}}{2\sigma_e^2}\right)de} + Q'\left(\frac{R_n^{op}-C_n^{op}}{R_n^{op}}\right)\cdot \frac{\int_{-\hat{C}_n^{op}}^{R_n^{op}-\hat{C}_n^{op}} \frac{1}{\sqrt{2\pi}\sigma_e}\exp\left(\frac{-e_n^{\,2}}{2\sigma_e^2}\right)de}{\int_{-\hat{C}_n^{op}}^{1-\hat{C}_n^{op}} \frac{1}{\sqrt{2\pi}\sigma_e}\exp\left(\frac{-e_n^{\,2}}{2\sigma_e^2}\right)de} \quad (9)$$

3.4 An Empirical Model of Video Quality Distortion

It is well-known that a channel coding rate that exceeds the capacity leads to unreliable communication and increased distortion. The precise increase in distortion depends on a number of factors, such as: *i*) the specific video content, *ii*) the error concealment/resilience and robustness of a particular source codec, and most importantly *iii*) the difference between the overestimated rate and the actual channel capacity. Commercially available video systems are complex and consist of a number of sub-parts. Hence, it is difficult (if not impossible) to deduce a closed form analytical expression that can precisely capture the impact of factor *iii*) on the video quality. Consequently, we resort to an empirical model that can be used in Eq. (9) to express the video quality for normalized rates[5]

To deduce the empirical model, we conduct the following measurement procedure:

1. Compute the operational channel capacity[6] of the underlying channel.
2. Extract the test SVC bit-stream at the operational channel capacity and compute the PSNR of the stream.
3. Encode the extracted bit-stream with the LDPC encoder and transmit it over the underlying channel.
4. Decode the bit-stream packets using the LDPC decoder and the SVC decoder (with error concealment[7]).
5. Compute the normalized PSNR[8] from the successfully decoded packets.
6. Repeat 3-5 with an increased rate and for four different transmission rates

Note that we conducted this experiment in the same environment setting (LDPC channel codes and foreman SVC bitstream) that we used for derivation of the redundancy parameter in section 3.3. From a comprehensive set of measurements, it can be observed that the normalized PSNR decreases as a function of normalized rates [**Fig. 6(b)**]. We derive the empirical model of distortion of video quality for rates over the channel capacity as:

$$f(x) = ax^b + c, \quad 0 \le x \le 0.12 \quad (10)$$

where x is a normalized rate over the capacity.

---

[5] Normalized rate is the rate difference between the over-estimated rate and the operational channel capacity, which normalized by the over-estimated rate, $\frac{R-C^{op}}{R}$.

[6] The *operational channel capacity* is the maximum achievable rate for the given channel code and for the underlying channel

[7] A lost frame was replaced with the previous frame.

[8] $Q'\left(\frac{R-C^{op}}{R}\right) / Q'(0)$

Through curve fitting, we found $a = -1.18 \times 10^{2}$, $b = 2.148$ and $c = 0.9898$ [**Fig.6 (b)**]. Note that these values are specially designated for the foreman video sequence. At this point, we have developed a simple and generic model which takes only a rate as input. Due to its simplicity, in some cases the model's predicted distortion is not very accurate. However, the magnitude of the prediction error is not significant and (as desired) the model provides somewhat conservative estimates. Consequently, we leverage this empirical video quality distortion model, $Q'(\cdot)$ in Eq. (9), to optimally select the source and channel rate, i.e., the channel coding rate [20].

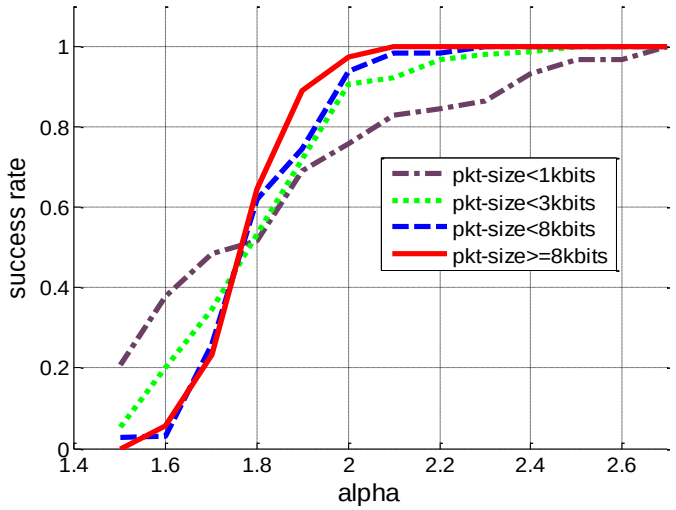


Fig. 7. For the LDPC, the probability of successful decoding depends on the parameter α and the size of packet.

3.5 Unequal Error Protection

As described in the previous section, for a rate-adaptive video, it is essential to accurately estimate/predict the channel capacity for reliable delivery of video content. However, it is always a challenging task to accurately estimate/predict the channel capacity, especially in wireless environments which incur significant channel impairments. More specifically, bursty channel error on a wireless medium leads to severe corruption of packets, in turn introducing considerable distortion in video quality. We note that in the case of bursty errors it is possible to reduce the distortion in video quality by employing an Unequal Error Protection (UEP) scheme. To this end, in this section we develop an UEP scheme which utilizes the characteristics of scalable video content and LDPC code.

It is well-known that a video content can be encoded with Intra (I), Predicted (P) or Bidirectional (B) frames. Each of those frames renders different amount of source information (per frame) and hence the impact on the video quality introduced by each of frame type differs dramatically. Therefore, the overall distortion in video content can be reduced by offering higher error protection to I frames (by providing more redundant bits on packets containing I frames) which are considered the most important frames. In addition, it is well-known that the error correction capability of an LDPC code depends on the packet size; a large-sized packet has a better chance of successful decoding than a smaller packet for a given underlying channel as shown in **Fig. 7**. We can observe from **Fig. 7** that the probability of successful decoding is a function of both the size of packet and redundancy parameter α (described in

Section 3.3).

Consequently, when a packet containing I frame is encoded with α that provides the probability of success decoding equal to 1 (i.e., $\alpha = 2.1$ for a packet size of 8 Kbits or more), it is highly likely that the packet will be decoded successfully at the receiver. Therefore, we channel encode I frame packets with a redundancy parameters that leads to success rate equal to 1. Note that without a UEP scheme the same redundancy parameter ($\alpha = 1.8$) is used for every packet and the total number of redundant bits can be defined (or specified) as

$$\alpha \cdot H(\varepsilon) \cdot \sum_{i=1}^{n} L_i = H(\varepsilon) \cdot \sum_{i=1}^{k} \alpha_i^I L_i^I + \alpha_P \cdot H(\varepsilon) \cdot \sum_{i=k+1}^{n} L_i^P \tag{11}$$

where $\alpha \cdot H(\varepsilon) = 1 - \overset{*\,op}{R}$; α_i represents the redundancy parameter for I frame packets; α_P represents the redundancy parameter for P frame packets; L_i represents the length of the i th packet; L_i^I represents the length of the i th packet containing I frames; and L_i^P denotes the length of the i th packet containing P frames. It should be noted that in Section 3.3, we define $\alpha = 1.8$ where almost every packet can be successfully decoded. However, when bursty errors are introduced, redundant bits with $\alpha = 1.8$ are insufficient for successful decoding.

The redundancy parameter for P frame packets can be simply computed as

$$\alpha_P = \frac{\alpha \sum_{i=1}^{N} L_i - \sum_{i=1}^{k} \alpha_i^I L_i^I}{\sum_{i=k+1}^{N} L_i^P}. \tag{12}$$

4. Experiment Results And Analysis

We now compare the performance of the proposed rate prediction architecture, $ORPA_{CLDS}$, with $ORPA_{CON}$. For $ORPA_{CON}$, we first use checksums for a time-window to find the PER, which is in turn used to estimate the channel capacity for the time-window. As explained earlier, this channel capacity estimate is then used as the predicted capacity for the next time-window. For $ORPA_{CON}$, the operationally optimal rate tuning scheme is also employed in the same way as the $ORPA_{CLDS}$ schemes.

To compare the performance of each architecture, experiments in this study were conducted as follows:

1. Estimate the operational channel capacities of k time windows in a trace by using the BER estimate of each packet and Eqs. (2) and (3).
2. Predict the operationally optimal rates of k time windows using Eq. (9). Note that $Q'(\cdot)$ is incorporated to find the rates
3. Calculate the average PSNR values over all time windows of a trace as follows

$$avg\ PSNR = \frac{1}{k}\sum_{n=1}^{k}\left[\begin{array}{l} Q(\overset{*\ op}{R_n}\cdot T)\cdot I(\overset{*\ op}{R_n} \le C_n^{op}) \\ +Q'\left(\left|\frac{\overset{*\ op}{R_n} - C_n^{op}}{\overset{*\ op}{R_n}}\right|\right)\cdot I(\overset{*\ op}{R_n} > C_n^{op}) \end{array}\right] \quad (13)$$

where $I(\cdot)$ is an Indication function; $\overset{*\ op}{R_n}$ is the predicted rate computed by using Eq. (9); C_n^{op} is the operational channel capacity; and k is the number of time windows in a trace.

4. Repeat 1-3 for different error traces

Note that for the predicted rate below the operational channel capacity, the video quality is calculated by using the first and second terms of Eq. () for the predicted rates exceeding the operational channel capacity.

As the experiment results of our previous study [1] indicated, $ORPA_{CLDS}$ performs better in capacity prediction than $ORPA_{CON}$ with practical SVC and LDPC code [**Fig. 8, Fig. 9**].

Table 3. Operational rate prediction performance before rate tuning

Phy (Mbps)	Xmit Rate (Kbps)	Operational Channel (PSNR-dB)	$ORPA_{CLDS}$ (dB)	$ORPA_{CON}$ (dB)
2	500	28.96	26.76	24.11
	750	31.02	29.94	27.99
	900	31.93	30.51	27.98
	1024	32.52	31.43	30.07
	Avg.	**31.11**	**24.41**	**27.54**
5.5	500	29.00	21.07	25.32
	750	30.88	23.39	27.10
	900	31.90	24.78	27.23
	1024	32.47	25.38	27.88
	Avg.	**31.06**	**23.66**	**26.88**
11	500	29.00	26.98	25.45
	750	30.88	29.01	30.19
	900	31.78	27.01	24.07
	1024	31.99	27.51	16.55
	Avg.	**30.91**	**27.63**	**24.06**

However, it can be also observed that accurate prediction alone does not necessarily imply better overall rate adaptation as shown in **Table 3**. This is due to over-predicted channel capacities which cause significant packet drops and considerable degradation of video quality. We resolve this problem by employing rate tuning scheme which provides significant performance improvement in terms of average PSNR values over all time windows of a trace as shown in **Table 4**.

Table 4 shows that rate tuning scheme significantly improves the overall performances of $ORPA_{CLDS}$ as well as $ORPA_{CON}$. For $ORPA_{CON}$, it should be noted that the optimal rate tuning degrades the overall performance at 11 Mbps. This is because $ORPA_{CON}$ predicts the channel capacity rather conservatively at 11 Mbps due to the fact that large number of packet drops are often introduced. Hence, the predictions of $ORPA_{CON}$ are consistently and considerably lower than the operational channel capacity. As a result, the performance of $ORPA_{CON}$ at 11 Mbps is slightly degraded.

Table 4. Operational rate prediction performance after rate tuning

Phy (Mbps)	Xmit Rate (Kbps)	Operational Channel (PSNR-dB)	$ORPA_{CLDS}$ (dB)	$ORPA_{CON}$ (dB)
2	500	28.96	27.67	27.81
	750	31.02	30.74	30.78
	900	31.93	31.51	31.25
	1024	32.52	32.43	32.31
	Avg.	**31.11**	**30.59**	**30.53**
5.5	500	29.00	27.92	28.23
	750	30.88	29.39	29.84
	900	31.90	30.78	29.98
	1024	32.47	32.38	30.95
	Avg.	**31.06**	**30.11**	**29.75**
11	500	29.00	27.59	25.22
	750	30.88	29.53	30.18
	900	31.78	30.67	22.73
	1024	31.99	30.12	15.01
	Avg.	**30.91**	**29.47**	**23.28**

Now, we extend our analysis on the performance of $ORPA_{CLDS}$ in more realistic environment by embodying practical SVC, LDPC code, and an actual IEEE 802.11b channel (using the traces that we collected). The simulation procedure with the video quality functions of the pre-encoded SVC bit-stream shown in **Fig. 6-(a)** is as follows:

1. Predict the optimal rates, $\overset{*}{R}_n^{op}$ and $\overset{*}{R}_{n+1}^{op}$, for two randomly chosen consecutive time-windows of a randomly chosen trace using Eq. (9).
2. Extract a SVC bitstream based on the rates from 1.
3. Encode the extracted bitstream packets based on the rates from 1 with the LDPC encoder and transmit the FEC-encoded packets over the underlying channel.
4. Decode the transmitted packets with LDPC and SVC decoder (with error concealment).
5. Repeat 1-4 for different traces with different transmission rates.

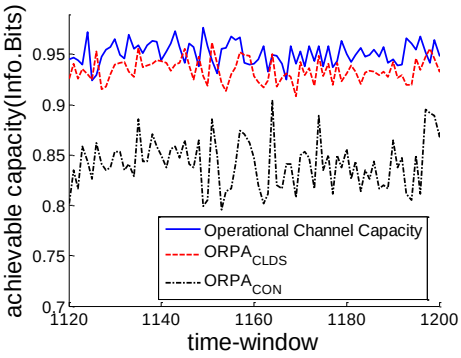
This simulation results are shown in **Table 5**.

Table 5. Performance of $ORPA_{CLDS}$ in conjunction with practical SVC and LDPC code

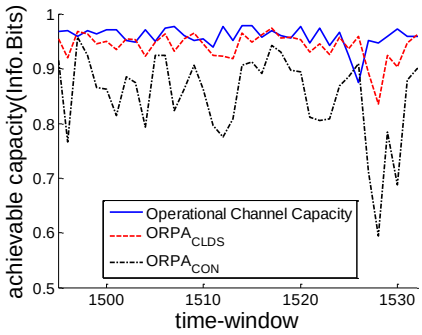
Phy	Xmit Rate (Kbps)	Time Window Index (Optimal Rates) $(\overset{*}{R}_n^{op}, \overset{*}{R}_{n+1}^{op})$	Extracted Bitstream PSNR(dB) Bitrate(Kbps)	Decoded Bitstream PSNR(dB)
11	500	437, 438 (0.774,0.744)	29.87 (379.7)	29.20
		1703, 1704 (0.877,0.886)	30.28 (440.3)	29.56
	750	227, 228 (0.904,0.901)	31.57 (677)	30.67
		1837, 1838 (0.890,0.926)	31.59 (681)	31.21
	900	384, 385 (0.638,0.558)	30.96 (538.9)	30.96
		1014, 1015 (0.849,0.779)	31.80 (732)	31.70
	1024	422, 423 (0.843, 0.818)	32.10 (849.1)	32.05
		739, 740 (0.841,0.826)	32.12 (853.5)	32.02

In **Table 5**, the first and the second columns represent underlying channels (or traces collected at the given physical data rates and transmit rates), the third column represents time window indices, which were randomly chosen from a randomly chosen trace, and the predicted (operational optimal) rates by $ORPA_{CLDS}$; the forth column represents bitrates and PSNRs of the video source, which was extracted based on the predicted rates; and PSNRs of video source which was preserved at a client are shown in the fifth column.

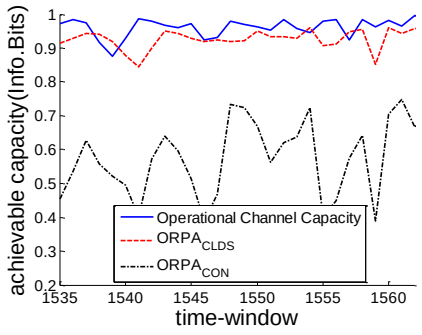
We can observe from **Table 5**, $ORPA_{CLDS}$, which employs the operational rate and an R-D empirical model for above-capacity provides reasonably accurate rate prediction performance



(a) Phy. rate– 2 Mbps & Xmit rate– 750Kbps

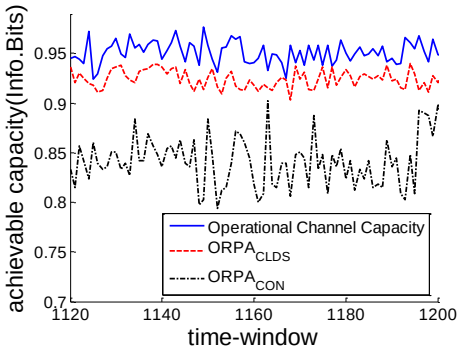


(b) Phy. rate– 5.5 Mbps & Xmit rate– 1 Mbps

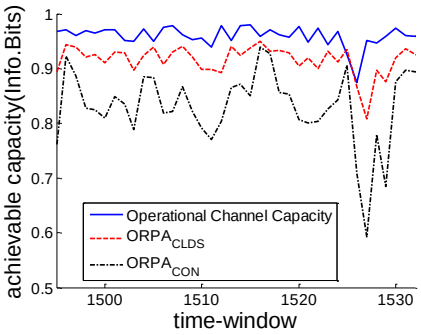


(c) Phy. rate– 11 Mbps & Xmit rate– 900 Kbps

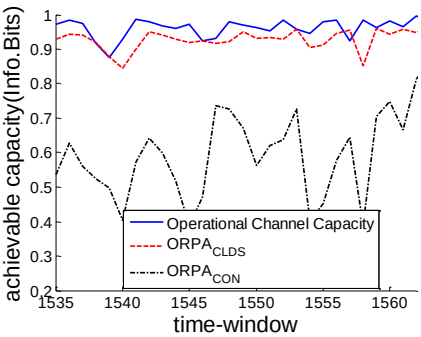
Fig. 8. The operational channel capacity prediction (zoomed in) results by $ORPA_{CLDS}$ and $ORPA_{CON}$ before rate tuning.



(a) Phys. rate– 2 Mbps & Xmit rate– 750 Kbps



(b) Phys. rate– 5.5 Mbps & Xmit rate– 1 Mbps



(c) Phys. rate– 11 Mbps & Xmit rate– 900 Kbps

Fig. 9. The operational channel capacity prediction (zoomed in) results by $ORPA_{CLDS}$, and $ORPA_{CON}$ after rate tuning.

(accurate average PSNR to the maximum PSNR for any given channel). Note that there is a small amount of PSNR reduction relative to an error-free decoded bitstream. This distortion is due to burst errors in wireless channel which cause severe corruption on a few packets. However, the distortion is very small, and the bitstream at the clients preserves most of the original video quality (PSNR) for randomly selected rate adaptation intervals (or time windows) and for different transmission rates [**Table 5**]. Note that since the test bitstream is 10 second long (300 frames), two consecutive time windows were used for these simulations. Thus, the first 150 frames and the rest were extracted according to two different rates. On the contrary, the video quality was computed from the entire bitstream after SVC decoding.

In **Fig. 10**, we compare $ORPA_{CLDS}$ with the proposed LDPC-based UEP scheme and without the proposed LDPC-based UEP scheme (for the case of I frame packets loss). We can see that when the LDPC-based UEP scheme is used, $ORPA_{CLDS}$ protects perceptually important information (i.e., I frame packets in this study), and hence the degradation of video quality is significantly reduced [**Fig. 10**]. It should be noted that in a worst case where an Instantaneous Decoder Refresh (IDR) frame packet, which is the first I frame packet in our SVC bit-stream, is lost, the entire bit-stream cannot be decoded, i.e., PSNR value becomes zero. **Fig. 10** excludes this worst case scenario.

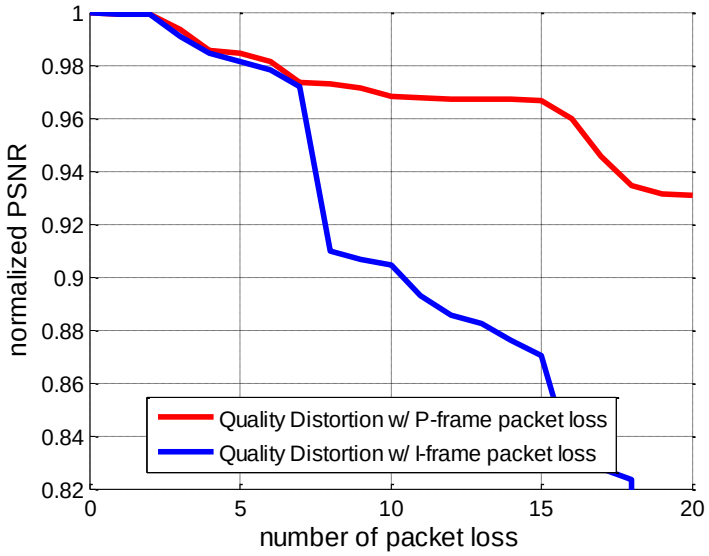


Fig. 10. The performance comparison in video qualities with the proposed UEP scheme and without UEP (when only I-frame packets are lost, i.e., the worst case).

5. Conclusion

In this paper, we derive and develop practical solutions such as an operational rate, an R-D empirical model for above-capacity SVC, and LDPC-based UEP scheme, to employ $ORPA_{CLDS}$ in practice. It is observed from trace-driven experiments that $ORPA_{CLDS}$ which incorporates the proposed schemes, provides an excellent rate prediction performance, and therefore robust rate adaptation can be achieved. Moreover, our experimental results showed

that the proposed architecture under CLDS protocols provides considerably better rate prediction performance and multimedia quality than that under conventional protocols.

References

[1] Y. Cho, S. Karande, K Misra, H. Radha, J. Yoo, and J. Hong, "On channel capacity estimation and prediction for rate adaptive wireless video," *IEEE Trans. on Multimedia*, vol.10, no7, pp.1419-1426, Nov.2008. Article (CrossRef Link).

[2] S. Karande, and H. Radha, "Hybrid erasure-error protocols for wireless video," *IEEE Trans. on Multimedia*, vol.9, no.2, pp.307-319, Feb.2007. Article (CrossRef Link)

[3] L. Larzon, M. Degermark, and S. Pink, "UDP lite for real time multimedia applications," *IEEE International Conference of Communication*, Jun.1999. Article (CrossRef Link).

[4] A. Singh, A. Konrad, and A. Joseph, "Performance evaluation of UDP lite for cellular video," *NOSSDAV*, 2001. Article (CrossRef Link).

[5] S. Khayam, S. Karande, H. Radha, and D. Loguinov, "Performance analysis and modeling of errors and losses over 802.11b LANS for high-bitrate real-time multimedia," *Signal Processing: Image Commun.*, vol.18, no.7, pp.575-595, Aug.2003. Article (CrossRef Link).

[6] F. Pauchet, C. Guillemot, M. Kerdranvat, S. Pateux, and P. Siohan, "Conditions for SVC bit error resilience testing," Joint Video Team (JVT) 17th, Oct.2005. Article (CrossRef Link).

[7] R. Riemann and K. Winstein, "Improving 802.11 range with forward error correction", MIT-CSAIL-TR-2005-011. Article (CrossRef Link).

[8] S. Karande, U. Parrikar, K. Misra, and H. Radha, "Utilizing SSR indications for improved video communication in presence of 802.11b residue errors," ICME, Jul.2006. Article (CrossRef Link).

[9] S. Karande, K. Misra, and H. Radha, "Survival Of The Fittest: An active queue management technique for noisy packet flows," *Journal of Advances in Multimedia* - Special Issue on Multimedia Networking, Hindawi Publication. Article (CrossRef Link).

[10] A. Khayam, "Wireless channel modeling and malware detection using statistical and information theoretic measures," Ph.D. Dissertation, Michigan State University, Dec.2006. Article (CrossRef Link).

[11] Y. Cho, S. Khayam, S. Karande, H. Radha, J. Kim, and J. Hong, "A multi-tier model for BER prediction over wireless residual channels," *CISS 2007*, Mar.2007. Article (CrossRef Link).

[12] Y. Cho, H. Radha, J. Yoo, and J. Hong, "A rate-distortion empirical model for rate adaptive wireless scalable video," *CISS 2008*, Mar.2008. Article (CrossRef Link).

[13] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, "Link-level measurements from an 802.11b mesh network", *SIGCOMM 2004*, Aug.2004. Article (CrossRef Link).

[14] S. Lin, and D. Costello, "Error control coding: fundamentals and applications," Englewood Cliffs, NJ: Prentice Hall, 1983. Article (CrossRef Link).

[15] P. Lambert, W. D. Neve, P. E. Neve, and I. Moerman, "Rate-distortion performance of H.264/AVC compared to state-of-the-art video codecs," *IEEE Trans. on Circuits and Systems for Video Technology*, vol.16, no.1, pp.134-140, Jan.2006. Article (CrossRef Link).

[16] A. Leon-Garcia, "Probability and random processes for electrical engineers," Addison-Wesley, 2nd Ed., 1994. . Article (CrossRef Link)

[17] W. Allan, "Time and frequency (Time-domain) characterization, estimation, and prediction of precision clocks and oscillators," *IEEE Trans. on Ultrasonics, Ferroelectrics, and Frequency Control*, vol.34, no.6, pp.647-654, Nov.1987. Article (CrossRef Link).

[18] S. Boyd, and L. Vandenberghe, "Convex optimization," Cambridge, 2006. Article (CrossRef Link).

[19] http://www.cs.toronto.edu/~radford/ftp/LDPC-2006-02-08/index.html

[20] R. Yeung, "A first course in information theory," Kluwer Academic/Plenum Publishers. Article (CrossRef Link).

[21] ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC, "Advanced video coding for generic audiovisual services," version 8.9, 2007. Article (CrossRef Link)

[22] H. Zhang, Y. Zheng, M. A. Khojastepour, and S. Rangarajan, "Cross-layer optimization for streaming scalable video over fading wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 3, pp. 344–353, Apr. 2010. Article (CrossRef Link).

Yongju Cho received the BS and MS degrees from Iowa State University, Ames, in 1997 and 1999, respectively, and the PhD degree from Michigan State University in 2009, all in electrical and computer engineering. He joined ETRI, Daejeon, Korea, in 2001, and has been engaged in the development of a data broadcasting system based upon MPEG-2/4 systems, IPMP and in the MPEG-21 area. His research interests include digital signal processing, computer vision and adaptive wireless video communications.

Jihun Cha received his BS degree in Computer Science in 1993 from Myongji University, Korea and MS and Ph.D. in Computer Science from Florida Institute of Technology, USA in 1996 and 2002 respectively. He joined Electronics and Telecommunications Research Institute (ETRI) of Korea in 2003. He has been participated in projects on the developments of various interactive multimedia technologies. His research interests include advanced user interaction, interactive rich media system, feature extraction and object detection/tracking in motion pictures.

Hayder Radha received the BS degree (with honors) from Michigan State University (MSU) in 1984, the MS degree from Purdue University, West Lafayette, IN, in 1986, and the PhM and PhD degrees from Columbia University, New York, in 1991 and 1993, respectively, all in electrical engineering. Currently, he is a professor of electrical and computer engineering (ECE) at MSU, the associate chair for Research and Graduate Studies of the ECE Department, and the director of the Wireless and Video Communications Laboratory at MSU. He was with Philips Research (1996–2000), where he worked as a principal member of the research staff and then as a consulting scientist in the Video Communications Research Department. He became a Philips Research Fellow in 2000. He was a distinguished member of the technical staff at Bell Laboratories, where he worked between 1986 and 1996 in the areas of digital communications, image processing, and broadband multimedia. His current research areas include wireless communications and networking, sensor networks, network coding, video coding, stochastic modeling of communication networks, and image and video processing. He has more than 100 peer-reviewed papers and 30 U.S. patents in these areas.

Kwang-deok Seo received the B.S., M.S., and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 1996, 1998, and 2002, respectively. From Aug. 2002 to Feb. 2005, he was with LG Electronics. Since March 2005, he has been a Faculty Member in the Computer and Telecommunications Engineering Division, Yonsei University, Gangwon, Korea, where he is an associate professor. Since September 2012, he has been a Courtesy Professor in the School of Electrical and Computer Engineering at the University of Florida, Gainesville, USA. His current research interests include digital video broadcasting, mobile IPTV, scalable video coding, and protocol design for scalable video transport. He is a member of KICS, KSBE, IEEE and IEICE.