Unified Distributed Source Coding Frames for Interactive Multiview Video Streaming

Zhi Liu, Gene Cheung, Yusheng Ji
The Graduate University for Advanced Studies, National Institute of Informatics
2-1-2, Hitotsubashi, Chiyoda-ku, Tokyo, Japan 101-8430
Email: {liuzhi, cheung, kei}@nii.ac.jp

***Abstract*—Because of differential coding used in standard video compression algorithms to exploit temporal correlation in adjacent frames for coding gain, a frame lost in network will cause error propagation in subsequent frames at the decoder. Previously proposed distributed source coding (DSC) frames can be periodically inserted to halt this error propagation by overcoming the uncertainty at encoder of which frames will be correctly received at decoder, without resorting to large intra-coded I-frames. In the case of interactive multiview video streaming (IMVS), where a user watches one of M available captured views at a time but can periodically select and switch to a neighboring view, the encoder must encode multiview video to enable this view-switching interactivity without knowing the exact view trajectories taken by viewers at stream time. In this paper, we propose a unified DSC frame construction for IMVS, so that the encoder can overcome both types of uncertainty in a coding-efficient manner; i.e., halt error propagation in differentially coded multiview video and facilitate periodic interactive view-switching at the same time. Having the additional unified DSC frames, we design a multiview frame structure to maximize the expected number of correctly decoded frames at decoder for a given bandwidth constraint. We develop a fast algorithm to find locally optimal structure parameters, and packetization and packet reordering strategies for transmission. Experimental results show that our optimized frame structures using unified DSC frames outperform naïve structures using I- and P-frames only by up to 49% in fraction of correctly decoded frames under typical network condition.**

I. Introduction

To exploit the inherent temporal correlation among successive video frames for coding gain, video compression standards like H.263 [1] and H.264 [2] employ *differential coding*, so that, instead of coding a target frame F_i independently, only the quantized differential d_i between the prediction (e.g., using previous frame F_{i-1} as predictor) and target F_i is encoded. While differential coding brings significant compression gain over independent coding, it also leads to error propagation when an irrecoverable frame loss occurs during network video streaming; a lost coded differential d_i for frame F_i will lead to incorrect decoding of subsequent frames F_j's, $j > i$, even if later differentials d_j's, $j > i$, are correctly delivered.

One naïve solution to halt error propagation in subsequent frames is to periodically insert independently coded I-frames. However, an intra-coded I-frame can be up to 10 times larger than an inter-coded P-frame, and hence frequent I-frame insertion is not a coding-efficient remedy. Instead, [3] proposed to periodically insert *distributed source coding* (DSC) frames. The key idea in DSC is to treat the *source coding* problem with uncertainty (the encoder does not know *a priori* which and how many transmitted frames will be lost over the network) as a *channel coding* problem instead: if the magnitude of propagation "noise" in the transform domain representation of subsequent frames (transform coefficients) can be bounded statistically, then the noise can be eliminated by deploying a proportional amount of channel code protecting the transform coefficients at the next DSC frame. [3] showed that using periodic DSC frames is significantly more coding-efficient than periodic I-frames when eliminating error propagation.

An orthogonal development recently is multiview video technologies. Because of continuing cost reduction in consumer-level cameras, a video sequence can now be recorded by a large array of cameras [4]; i.e., at each time instant, images of the same scene are simultaneously captured by multiple closely spaced cameras from different viewpoints. Given encoded multiview content at the server, in an *interactive multiview video streaming* (IMVS) scenario [5], a viewer can observe one of M available captured views at a time, but can periodically select and switch to a neighboring view, so that only frames of chosen viewpoint are received. While IMVS offers viewers a new media interaction (view-switching), it creates a new source coding difficulty: in an on-demand IMVS system, encoder must encode the multiview video *a priori* to facilitate periodic view-switching, *without* knowing the eventual view trajectory taken by each viewer at stream time. The encoder hence must resolve a different kind of uncertainty: given observer of view v, which view u, $v-1 \leq u \leq v+1$, will he select for observation at next view-switching instant during streaming.

In this paper, we construct a new *unified DSC frame* for IMVS, so that the encoder can overcome both kinds of uncertainty in a coding-efficient manner. In other words, a single unified DSC frame can halt error propagation in differentially coded video, *or* facilitate periodic interactive view-switching. The key to the unified DSC frame construction is to view the source coding problem with uncertainty again as a channel coding problem, so that the encoder only needs to add enough channel codes to handle the larger of the two kinds of noise due to error propagation and view-switching, *not* the aggregate of the two noise terms. Given the addition of unified DSC frames, we design a multiview frame structure to maximize the expected number of correctly decoded frames at decoder for a given bandwidth constraint. We develop a

fast algorithm to find locally optimal structure parameters, and packetization and packet reordering strategies for transmission. Experimental results show that our optimized frame structures using DSC frames outperform naïve structures using I- and P-frames only by up to 49% in fraction of correctly decoded frames under typical network condition.

The outline of the paper is as follows. We first discuss related works in Section II. We then outline the multiview video streaming system and present our proposed coding structure using DSC in Section III. The problem of finding optimized parameters for our proposed DSC-based coding structure is formalized in Section V, and the corresponding algorithm is presented in Section VI. Results and conclusion are presented in Section VII and VIII, respectively.

II. Related Work

Conventional transport layer strategies to combat network packet losses include *forward error correction* (FEC) and *automatic retransmission requests* (ARQ). ARQ is known to be inapplicable in many video streaming scenarios—e.g., video streaming with a low-delay requirement (a retransmitted packet is late and useless), video multicast to a large group (due to the well-known NAK implosion problem [6]), etc. Deploying block FEC alone to the extent that lossless transmission is guaranteed under varying network conditions translates to a large consumption of precise network bandwidth. Given streaming video is in general more tolerable to packet losses, in our approach we use a judicious amount of FEC in combination with error-resilient video coding via DSC for optimal streaming performance.

Though [3] proposed a DSC-based tool to halt error propagation in single-view video at the DSC-frame boundary, the authors did not discuss how the proposed tool can be optimally deployed in a real network streaming scenario. [5] proposed to use DSC for view-switching in an IMVS application, but did not consider network packet losses and their impact on visual quality. In this paper, we combine the advantages of both previous proposals via the construction of a single unified DSC frame that can halt error propagation *or* facilitate view-switching. Further, we design a frame structure using our proposed unified DSC frames, and optimize its network transmission via packetization and packet ordering, assuming a Gilbert-Elliot packet loss model.

III. Multiview Video Multicast System

Our goal is to maximize video quality using DSC frames to evade error propagation in general IMVS systems. For concreteness, however, we focus on the scenario where a Wireless Wide Area Network (WWAN) multicasts multiview video to a group of viewers. We first overview components in a WWAN multiview video multicast system. We then discuss a loss model for the WWAN transmission link.

A. System Overview

The multiview video multicast system is illustrated in Fig. 1. A scene of interest is captured by a 1D array of M closely spaced cameras from different viewing angles. Different views of the same video content are synchronously multicasted on the WWAN network, one view per multicast channel. At any given instant, a user can obtain and observe one of M captured views by subscribing to the corresponding multicast channel. User can also periodically switch to an adjacent view by re-subscribing to a new multicast channel every T video frames.

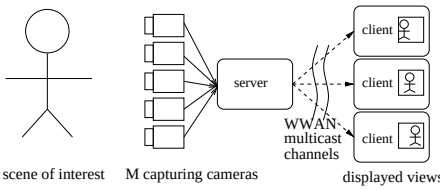


Fig. 1. Overview of Multiview Video Multicast System

B. Packet Loss Model

To model WWAN packet losses, we use the Gilbert-Elliot (GE) model (a commonly used model for wireless losses [7]) with independent and identically distributed (iid) packet loss probabilities g and b for each of *good* and *bad* state, and state transition probabilities p and q to move between states. In other words, when a packet arrives, a weighted coin (with weight p or q depending on current state) is first tossed to determined whether it stays in the current state or transition to the other state. Then a second weighted coin (with weight g or b depending on current state) is tossed to determine if the packet is lost or not. See Fig. 2 for an illustration.

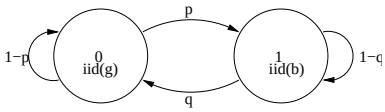


Fig. 2. Gilbert-Elliot loss model. g and b are the packet loss probabilities in 'good' and 'bad' states, respectively, and p and q are transition probabilities between states. 1 (0) indicates a bad (good) state.

IV. Coding Mechanism

We now design a frame structure for an IMVS system to encode multiview video content as I-, P-frames and two different kinds of Distributed Source Coding (DSC) frames. We then discuss how source bits of encoded frames are packed into IP packets (packetization), and how the created packets are ordered for transmission over WWAN.

A. Overview of Video Frame Types

We first explain the four types of video frames used in our structure as follows.

1) Conventional I- and P-frames: I-frame, denoted as $I_{i,v}$ for frame at instant i and view v, is an *intra*-coded frame and can be decoded independently from other frames. *P-frame*, denoted as $P_{i,v}$, is *inter*-coded via motion compensation; i.e., using another encoded frame $F_{i-1,v}$ as predictor, only the frame difference—block-by-block motion vectors and quantized motion prediction residuals—are encoded [1], resulting in a frame size much smaller than an I-frame. However, correct decoding of a P-frame $P_{i,v}$ requires first the correct decoding of predictor $F_{i-1,v}$ at the decoder.

2) Drift-Elimination DSC Frames: Drift-Elimination DSC (DE-DSC) frame [3], denoted as $W^1_{i,v}$, is designed to halt error propagation (coding drift) due to *prediction mismatch* between encoder and decoder at the DE-DSC frame boundary. Mismatch happens when there are irrecoverable packet losses in the transmission network, resulting first in a reconstructed frame $\hat{F}_{i,v}$ of instant i at decoder that is different from encoded $F_{i,v}$ at encoder. Due to differential coding using $F_{i,v}$ as predictor, subsequent reconstructed frames $\hat{F}_{j,v}$'s, $j > i$, will also be incorrect, even if differentials $d_{j,v}$'s are correctly delivered, resulting in error propagation. DE-DSC frame $W^1_{l,v}$ halts this error propagation—i.e., restore $\hat{F}_{l,v}$ at decoder back to encoded $F_{l,v}$ at encoder at a later instant l.

For implementation, we first assume prediction residuals of a given frame are block-by-block transformed using Discrete Cosine Transform (DCT), with the resulting DCT coefficients quantized as done in [1]. If the magnitude of reconstruction noise due to error propagation in different bit-planes of the quantized coefficients can be bounded statistically, then DE-DSC frame can deploy just the right amount of channel codes (*low-density parity check* (LDPC) codes are used in [3] and [8]) for each bit-plane to remove the noise, given the noise statistics of that bit-plane. We retain the assumption in [3] that the motion vectors of predicted frames between two DE-DSC frames are correctly delivered, and only prediction residuals can be lost during transmission; this assumption helps bound the noise level in transform coefficients to a manageable amount. Henceforth, a DE-DSC frame $W^1_{i,v}$ capable of halting propagated error given prediction residuals of *any* k or fewer preceding frames have been lost will be denoted as $W^1_{i,v}(k)$. We will discuss how the noise statistics can be derived to compute channel codes used in $W^1_{i,v}(k)$ in Section VII.

3) Multi-Predictor DSC Frames: Multi-Predictor DSC (MP-DSC) frame [8], denoted as $W^2_{i,v}$, generalizes the single-predictor motion compensation paradigm in P-frame by employing *multiple* predictors at encoder. At decoder, only *one* in the encoder set of predictors needs to be available for the MP-DSC frame to be correctly decoded. For IMVS, we use MP-DSC frames for view-switching: MP-DSC frame $W^2_{i,v}$ will be encoded using predictor frames $F_{i-1,u}$'s of previous instant, where $u \in \{\max(1, v-1), \ldots, \min(M, v+1)\}$. A client of view u can thus switch to view v and decode frame $W^2_{i,v}$ correctly, using $F_{i-1,u}$ in his buffer as predictor.

For implementation, MP-DSC frames can be encoded similarly to DE-DSC frames. To overcome the uncertainty of which predictor will be available at decoder, a MP-DSC frame first encodes multiple sets of motion vectors, one for each predictor frame. Then, the resulting quantized DCT coefficients of the prediction residual for each predictor are compared against the coefficients of the target frame to compute the noise statistics in each bit-plane. Appropriate amount of LDPC codes are then deployed in each bit-plane to overcome the *largest* noise of all prediction residuals for that plane [8].

We now encode MP-DSC so that it can also halt error propagation in the same view, as done in DE-DSC (new frame will be called DE/MP-DSC): in addition to the noise statistics

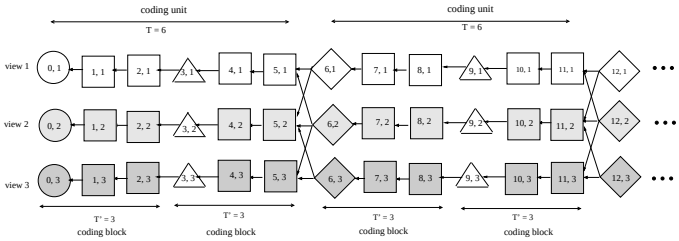


Fig. 3. Example of proposed coding structure for $M = 3$ views and coding block size $T' = 3$, coding unit size $T = 6$. Circles, squares, triangles and diamonds are I-, P-, DE-DSC and MP-DSC frames. Each frame $F_{i,v}$ is labeled by its time index i and view v.

of different prediction residuals from multiple predictors, we consider the computed noise statistics for a DE-DSC frame $W^1_{i,v}(k)$ of the same view also when deciding the amount of LDPC code used for each bit-plane. Note that doing so means that *the overhead of a DE/MP-DSC frame is not the sum of overheads from both a DE-DSC and a MP-DSC frame, but only the larger of the two*. This is the key in creating a coding-efficient DE/MP-DSC frame.

B. IMVS Frame Structure

We assume IMVS application requires a view-switching period of T frames, and we consider coding of a Group of Pictures (GOP) of ΘT frames, $\Theta \in \mathcal{Z}^+$ (i.e., user can perform up to $\Theta - 1$ view-switches in a GOP). A segment of T consecutive frames in a single view v is called a *coding unit* $U_{\theta,v}$. A coding unit $U_{\theta,v}$ is coded as a sequence of *coding blocks* $L_{\theta,v}(j)$'s of T' frames each, $T' < T$, as follows. We first encode a starting DE/MP-DSC frame $W^2_{\theta T,v}$ (if it is the first coding unit, i.e., $\theta = 0$, then use I-frame $I_{0,v}$ instead) with $T' - 1$ trailing P-frames $P_{\theta T+i,v}$, $1 \le i < T'$, each motion-compensated using previous frame as predictor, into the first coding block $L_{\theta,v}(1)$. We then encode a DE-DSC frame $W^1_{\theta T+T',v}(k)$, $k < T'$, followed again by $T'-1$ trailing P-frames as the second coding unit $L_{\theta,v}(2)$. See Fig. 3 for an illustration.

If a DE-DSC $W^1_{i,v}(k)$ or DE/MP-DSC $W^2_{i,v}(k)$ frame can be correctly reconstructed, it can mitigate error propagation due to earlier irrecoverable packet losses by serving as the good predictor for the following frames. Larger recoverability k results in a larger DE-DSC or DE/MP-DSC frame, however.

C. Packetization of Encoded Bits in Coding Block

We now discuss how we packetize encoded bits from the frame structure into packets. Since correct decoding of a DE-DSC or DE/MP-DSC frame requires all motion vectors of preceding P-frames to be transmitted losslessly, we design a packetization scheme so that motion vectors are protected more heavily against packet losses than prediction residuals.

As illustrated in Fig. 4, encoded bits in a P-frame are divided into *header*, *motion vectors* and *prediction residuals*. We group encoded bits of I-, DE-DSC and DE/MP-DSC frames plus header and motion vectors of P-frames in coding unit $U_{\theta,v}$ together for packetization into $M_{\theta,v}$ *motion packets*, each of maximum size MTU bytes (Maximum Transmission

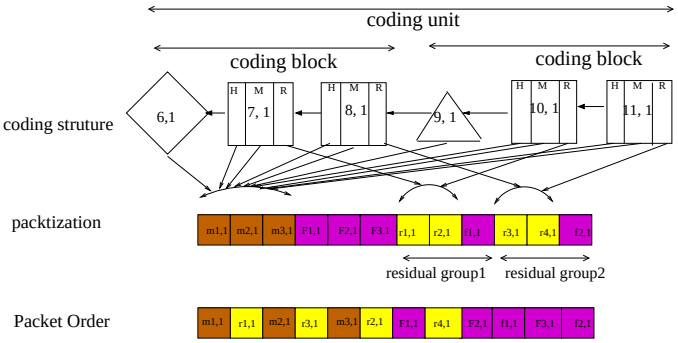


Fig. 4. The three stages of the transmission scheme: i) encoding of captured images into frames in coding structure, ii) packetization of encoded bits into IP packets, and iii) ordering of generated packets for transmission. Motion, residual and FEC packets are indicated by red, yellow and blue, respectively.

Unit). These are the important packets that require more loss protection.

We next packetize encoded bits of prediction residuals in P-frames: we gather residual bits of the lth frame of each coding block $L_{\theta,v}(j)$ into the lth *residual group*, which are then divided into packets of maximum size MTU bytes. One can generalize the above scheme so that ρ frames of each coding block goes into one residual group. Let the number of *residual packets* in each residual group be $r_{\theta,v}$. The number of residual groups is $G = \lceil F'/\rho \rceil$.

After packetization of the encoded source bits in all coding blocks $L_{\theta,v}(j)$'s in a coding unit $U_{\theta,v}$, we next generate Forward Error Correction (FEC) packets to protect the motion and residual packets unequally. We first generate $F_{\theta,v}$ level-1 FEC packets to protect $M_{\theta,v}$ motion packets in coding unit $U_{\theta,v}$. We then generate $f_{\theta,v}$ level-2 FEC packets to protect $r_{\theta,v}$ residual packets in *each* residual group. Assuming a perfect code (e.g., Reed-Solomon, network codes) is used for FEC, $M_{\theta,v}$ motion packets can be correctly recovered if at least $M_{\theta,v}$ of $M_{\theta,v} + F_{\theta,v}$ transmitted motion plus level-1 FEC packets are delivered. Similarly, each residual group can be correctly recovered if at least $r_{\theta,v}$ of $r_{\theta,v} + f_{\theta,v}$ residual plus level-2 FEC packets are delivered. We discuss how $F_{\theta,v}$ and $f_{\theta,v}$ for each unit $U_{\theta,v}$ are selected in the next section.

D. Packet Ordering

After packetizing encoded source bits of the frames in a coding unit into packets and generating two levels of FEC packets, we now sort the generated packets into a transmission order. Given the WWAN loss model is a GE model, the guiding principle we use is *interleaving*: space the motion information and prediction residuals apart so that the adverse effect of one trip into the bad state in the GE model will be spread evenly across the coding unit.

Let ratio of the number of packets for motion plus level-1 FEC packet to residual plus level-2 FEC packets be $\lambda_M : \lambda_R$, where λ's are integers. We alternatively select λ_M motion & level-1 FEC packets and λ_R residual & level-2 FEC packets into a transmission order. When selecting residual packets, we select packets from different residual groups in a round-robin fashion. Doing so means the spacings among motion packets and among residual packets are maximized.

V. Problem Formulation

We now formalize an optimization problem to find the optimal structure parameters: period of insertion T' and error recoverability k for DE-DSC $W^1_{i,v}(k)$ and DE/MP-DSC $W^2_{i,v}(k)$, and the number of FEC packets for each level, $F_{\theta,v}$ and $f_{\theta,v}$. We first discuss the WWAN transmission constraint for each coding unit. We then derive the probabilities that: i) an entire coding unit $U_{\theta,v}$ is correctly decoded, and ii) the DE-DSCs in a coding unit are correctly decoded. We then write the appropriate objective function for our optimization.

A. WWAN Transmission Constraint

We assume a WWAN transmission constraint in number of packets B for each coding unit $U_{\theta,v}$. We can write the WWAN transmission constraint as follows:

$$M_{\theta,v} + Gr_{\theta,v} + F_{\theta,v} + Gf_{\theta,v} \leq B \tag{1}$$

In words, (1) states that the total packets used for motion and residual packets and FEC packets for both levels cannot exceed the WWAN bandwidth of B packets for unit $U_{\theta,v}$.

B. Preliminaries

For ease of later derivation, we first formally define mathematical quantities that are useful when dealing with a GE packet loss model. Let $P(i)$ be the probability of having *at least* i consecutive transmissions in the good state in the GE model, given transmission starts in bad state. Further, let $p(i)$ be the probability of having *exactly* i good state transmissions between two bad state transmissions, given transmission starts in bad state. We write $P(i)$ and $p(i)$ as follows:

$$\begin{aligned} P(i) &= \begin{cases} 1 & \text{if } i = 0 \\ q(1-p)^{i-1} & \text{otherwise} \end{cases} \\ p(i) &= \begin{cases} 1-q & \text{if } i = 0 \\ q(1-p)^{i-1}p & \text{otherwise} \end{cases} \end{aligned} \tag{2}$$

Similarly, we define $Q(i)$ and $q(i)$ as the probability of *at least* i consecutive bad state transmissions, and the probability of *exactly* i bad state transmissions, given transmission starts in good state. Equations for $Q(i)$ and $q(i)$ will be the same as those for $P(i)$ and $p(i)$, with the parameters p and q interchanged.

We can now recursively define the probability $R(m,n)$ of *exactly* m bad state transmissions in n total transmissions, given transmission starts in bad state:

$$R(m,n) = \begin{cases} P(n) & \text{for } m = 0 \text{ and } n \geq 0 \\ \sum_{i=0}^{n-m} p(i)R(m-1, n-i-1) & \text{for } 1 \leq m \leq n \end{cases} \tag{3}$$

Similarly, the probability $S(m,n)$ of *exactly* m good state transmissions in n total transmissions, given transmission starts in good state, is written in the same form as (3), with $Q(i)$ and $q(i)$ replacing $P(i)$ and $p(i)$ in (3), respectively.

C. Correctly Received Probability of a Coding Unit

Given previous definitions, we now derive the probability $\alpha_{\theta,v}$ that all motion and residual packets of unit $U_{\theta,v}$ are correctly delivered. As previously discussed, besides the source packets, two levels of FEC packets are employed to protect against WWAN losses. Hence, a necessary condition for recovery is to require the number of lost packets do not exceed the total number of FEC packets used: $F_{\theta,v} + Gf_{\theta,v}$.

We first write $\alpha_{\theta,v}$ as a weighted sum of $\alpha^0_{\theta,v}$ and $\alpha^1_{\theta,v}$, the decoding success probability of unit $U_{\theta,v}$ given transmission starts in good and bad state respectively:

$$\alpha_{\theta,v} = \left(\frac{q}{p+q}\right)\alpha^0_{\theta,v} + \left(\frac{p}{p+q}\right)\alpha^1_{\theta,v} \tag{4}$$

Assuming transmission starts in the good state, m of B total packets can be transmitted in good state with probability $S(m,B)$. Unit $U_{\theta,v}$ can be successfully received if at least $r \geq M_{\theta,v} + Gr_{\theta,v}$ packets are correctly delivered, and these r packets can be a sum of r_G and $r - r_G$ delivered packets in good and bad states respectively. Hence we can write $\alpha^0_{\theta,v}$ as:

$$\alpha^0_{\theta,v} \approx \sum_{m=0}^{B} S(m,B) \sum_{r=M_{\theta,v}+Gr_{\theta,v}}^{B} \sum_{r_G=0}^{r} P_G(r_G,m) P_B(r-r_G, B-m) \tag{5}$$

where $P_G(x,y)$ and $P_B(x,y)$ are the probabilities of exactly x delivered packets in y tries in good and bad state respectively:

$$\begin{aligned} P_G(x,y) &= \begin{cases} C^y_x (1-g)^x g^{y-x} & \text{if } x \leq y \\ 0 & \text{o.w.} \end{cases} \\ P_B(x,y) &= \begin{cases} C^y_x (1-b)^x b^{y-x} & \text{if } x \leq y \\ 0 & \text{o.w.} \end{cases} \end{aligned} \tag{6}$$

C^y_x denotes the number of combinations of y chooses x. $\alpha^1_{\theta,v}$ can be written similarly to $\alpha^0_{\theta,v}$ in (5) and is hence omitted.

D. Correctly Decode Probability for DSC

We next derive the correctly decode probability for all DSC frames (DE-DSC and DE/MP-DSC) in unit $U_{\theta,v}$, *given* not all motion and residual packets in the unit are correctly delivered. A DE-DSC frame is correctly decoded if: i) the motion packets between two DSC frames are correctly recovered, and ii) residual packets of at least $T' - k$ of T' preceeding frames are correctly recovered.

We first consider the probability $\delta_{\theta,v}$ that the motion information of all frames in unit $U_{\theta,v}$ are correctly recovered. As done in (4) for $\alpha_{\theta,v}$, $\delta_{\theta,v}$ can also be written as a weighted sum of $\delta^0_{\theta,v}$ and $\delta^1_{\theta,v}$, depending on whether transmission starts in good or bad state. Let $\gamma_M = (M_{\theta,v} + F_{\theta,v})/B$ be the fraction of bandwidth for transmission of motion and level-1 FEC packets. $M_{\theta,v}$ motion packets are correctly recovered if at least $r \geq M_{\theta,v}$ packets are correctly delivered, where again r can be a sum of delivered packets r_G and $r - r_G$ transmitted in good and bad state. The difference from (5) is that for given m and $B-m$ transmissions in good and bad states, only portions $\gamma_M m$ and $\gamma_M(B-m)$ are used for transmission of motion and level-1 FEC packets. We can now write $\delta^0_{\theta,v}$ as follows:

$$\delta^0_{\theta,v} \approx \sum_{m=0}^{B} S(m,B) \sum_{r=M_i}^{M_{\theta,v}+F_{\theta,v}} \sum_{r_G=0}^{r} P_G(r_G, \gamma_M m) P_B(r-r_G, \gamma_M(B-m)) \tag{7}$$

Next, we derive the probability $\eta_{\theta,v}$ that residual packets of at least $T' - k$ of T' frames are recovered for each DSC frame to be correctly decoded. Given our packetization scheme, that means at least $\lceil \frac{T'-k}{\rho} \rceil$ residual groups are correctly recovered. Because interleaving was performed to space packets in one residual group to be as far apart as possible, we can treat packet losses within a residual group as iid losses, with probability $l = \left(\frac{p}{p+q}\right) g + \left(\frac{q}{p+q}\right) b$. The probability $\phi_{\theta,v}$ that a residual group is correctly recovered is hence:

$$\phi_{\theta,v} = \sum_{r=r_{\theta,v}}^{r_{\theta,v}+f_{\theta,v}} C^{r_{\theta,v}+f_{\theta,v}}_r (1-l)^r l^{r_{\theta,v}+f_{\theta,v}-r} \tag{8}$$

In words, (8) states that a residual group must receive at least $r_{\theta,v}$ packets for the group to be correctly recovered.

Having derived $\phi_{\theta,v}$, we can now write $\eta_{\theta,v}$ as follows:

$$\eta_{\theta,v} \approx \sum_{j=\lceil \frac{T'-k}{\rho} \rceil}^{G-1} \binom{G-1}{k} \phi^j_{\theta,v} (1-\phi_{\theta,v})^{G-j} \tag{9}$$

where the upper limit in (9) is $G-1$, since by assumption not all the motion and residual packets in the coding unit are correctly recovered.

E. Objective Function

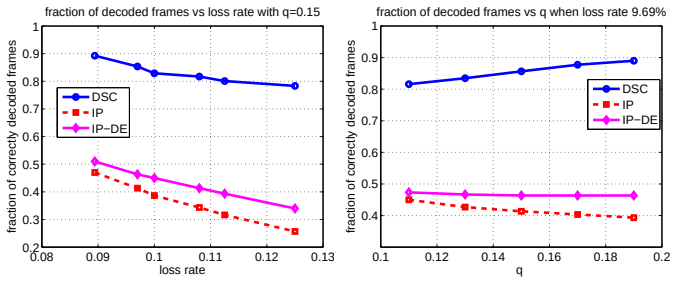
We can now write our objective function as the expected number Z_v of correctly decoded frames in the entire GOP for view v. For a coding unit $U_{\theta,v}$ to be correctly decoded, each previous unit $U_{j,v}$, $j < \theta$, must be either fully correctly received with probability $\alpha_{j,v}$, or have all its DSC frames correctly decoded with probability $(1-\alpha_{j,v})\delta_{j,v}\eta_{j,v}$. If the entire unit $U_{\theta,v}$ is correctly delivered as well (with probability $\alpha_{\theta,v}$), then all T frames are correctly decoded. Otherwise, at least the T/T' DSC frames are correctly decoded if the motion packets and enough residual packets are correctly received. Z_v can now be written as follows:

$$\begin{aligned} Z_v &= \sum_{\theta=1}^{\Theta} \left(\alpha_{\theta,v} T + (1-\alpha_{\theta,v})\delta_{\theta,v}\eta_{\theta,v}(\frac{T}{T'})\right) Y_{i,v} \\ Y_{\theta,v} &= \prod_{j=1}^{\theta-1} \alpha_{j,v} + (1-\alpha_{j,v})\delta_{j,v}\eta_{j,v} \end{aligned} \tag{10}$$

The goal is to find parameters that maximize Z_v in (10) for the entire GOP subject to transmission constraint (1) for each coding unit $U_{\theta,v}$.

VI. Coding Structure Optimization

In this section, we describe a simple heuristic to find good structure parameters for the optimization formulated previously. We optimize one coding unit at a time, starting from the last unit $U_{\Theta-1,v}$ and work backwards. We first insert one DE-DSC frame in a coding unit $U_{\theta,v}$. We then locally search for error recoverability k in the lone DE-DSC



(a) varying loss rate (b) varying duration in bad state

Fig. 5. Comparison of fraction of decoded frames for different coding structures: (a) buffer time, (b) varying average duration in bad state in DE model while WWAN loss rate is fixed at 0.0969.

frame $W^1_{i,v}(k)$ and the number of FEC packets $F_{\theta,v}$ and $f_{\theta,v}$ in each level that maximize objective function (10), while observing the coding unit bandwidth constraint (1). Then we incrementally increase the DE-DSC insertion frequency, each time locally searching for optimal error recoverability k and FEC packets $F_{\theta,v}$ and $f_{\theta,v}$, until the objective function cannot be further increased.

VII. Experimentation

To test the performance of our optimized coding structure in typical WWAN loss environment, we set up the following experiment. For source coding, we use the DSC codec in [9]—a H.263-based codec[1] with modifications to encode bit-planes of DCT coefficients given noise statistics using LDPC codes—to encode a 300-frame MPEG multiview video test sequence `akko` at 640×480 resolution. We assume there are $M = 3$ captured views, and a user can switch view every $T = 30$ frames. Given video playback speed of $FPS = 30$fps, that means maximum view-switching delay is 1 second.

We fixed the quantization parameters for I-, P- and DSC frames so that the resulting visual quality in Peak Signal-to-noise Ratio (PSNR) after compression is roughly 32.5dB. The size of each DE-DSC frame $W^1_{i,v}(k)$ varies with k: increasing k by 2 will lead to a 8% increase in size. To generate noise statistics for $W^1_{i,v}(k)$, we set the prediction residuals of the last k P-frames preceding the DE-DSC frame to zero; [5] has shown that this induces the largest possible propagation error given k frame losses in T' frames. MTU is assumed to be 1500 bytes. Typical sizes for I-, P-, DE-DSC and DE/MP-DSC frames are 5, 1, 2 and 2 packets, respectively. For WWAN network, bandwidth for each multicast channel is assumed to be 450kbps, and packet losses were simulated according to fixed GE model parameters: $p = 0.15$, $g = 0.05$, $b = 0.8$ (q may vary to effect loss rates from 0.09 to 0.125).

We compare the resulting fraction of correctly decoded frames for three different coding structures. `DSC` is our optimized structure using DE-DSC and DE/MP-DSC for a GOP. `IP` is a structure using I- and P-frames only for the entire GOP. `IP-DE` is a non-optimized structure with periodic DE-DSC inserted into the GOP besides I- and P- frames.

[1]The tradeoff between DSC and FEC for evasion of error propagation and loss protection would be similar if a H.264-based codec is used instead.

In Fig. 5(a), we see the resulting fraction of correctly decoded frames for all three structures against the average WWAN loss rate with GE parameters q, g and b fixed. We see that `DSC` is the best performing structure; it outperformed `IP` and `IP-DE` by up to 53% and 44%, respectively. The reason is because `IP` could not properly evade error propagation when irrecoverable packet losses were encountered. In contrast, `DSC` could rely on DE-DSC frames to halt error propagation. Further, since we optimized the use of DE-DSC and DE/MP-DSC for the entire GOP, early in the GOP tends to have fewer DE-DSC inserted but larger error recoverability k, so that later frames in the GOP can be decoded with higher probability. This optimization leads to a better performance of `DSC` over non-optimized `IP-DE`.

In Fig. 5(b), we see the performance of the three structures when WWAN loss rate was fixed at 0.0969 but the average duration in bad state was varied. We see that `DSC` also outperformed `IP` and `IP-DE` by up to 49% and 42%, respectively.

VIII. Conclusion

Evading error propagation due to packet losses in differentially coded video without using independently coded I-frame is difficult, since at encoder there exists an uncertainty of which frames will be correctly received at decoder. Similarly, in interactive multiview video streaming, encoder must encode multiview video to facilitate periodic view-switching with the uncertainty of which view trajectory a user will choose at stream time. In this paper, we propose a unified distributed source coding (DSC) frame, so that the encoder can overcome both types of uncertainty in a coding-efficient manner: halt error propagation in differentially coded multiview video or facilitate periodic view-switching using a single frame. We show that optimal use of our proposed DSC frame in a coding structure can improve performance significantly in fraction of correctly decoded frames over previous structures.

References

[1] *Video Coding for Low Bitrate Communication*, ITU-T Recommendation H.263, February 1998.

[2] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no.7, July 2003, pp. 560–576.

[3] A. Sehgal, A. Jagmohan, and N. Ahuja, "Wyner-Ziv coding of video: an error-resilient compression framework," in *IEEE Transactions on Multimedia*, vol. 6, no.2, April 2004, pp. 249–258.

[4] T. Fujii, K. Mori, K. Takeda, K. Mase, M. Tanimoto, and Y. Suenaga, "Multipoint measuring system for video and sound—100 camera and microphone system," in *IEEE International Conference on Multimedia and Expo*, Toronto, Canada, July 2006.

[5] G. Cheung, A. Ortega, and N.-M. Cheung, "Interactive streaming of stored multiview video using redundant frame structures," in *IEEE Transactions on Image Processing*, vol. 20, no.3, March 2011, pp. 744–761.

[6] J. Crowcroft and K. Paliwoda, "A multicast transport protocol," in *ACM SIGCOMM*, New York, NY, August 1988.

[7] I.-H. Hou and P. R. Kumar, "Scheduling heterogeneous real-time traffic over fading wireless channels," in *IEEE INFOCOM*, San Diego, CA, March 2010.

[8] N.-M. Cheung, A. Ortega, and G. Cheung, "Distributed source coding techniques for interactive multiview video streaming," in *27th Picture Coding Symposium*, Chicago, IL, May 2009.

[9] N. Cheung and A. Ortega, "Distributed source coding application to low-delay free viewpoint switching in multiview video compression," in *Proc. of Picture Coding Symposium, PCS'07*, Lisbon, Portugal, Nov. 2007.