EXIT-based Side Information Refinement in Wyner-Ziv Video Coding

Wen Ji, *Member, IEEE,* Pascal Frossard, *Senior Member, IEEE,* Yiqiang Chen, *Member, IEEE*

***Abstract*—The accuracy of the side information (SI) is critical in the performance of distributed video coding algorithms. The side information is typically built at decoder based on the reconstructed data and on channel coding parity bits transmitted by the encoder. The optimal encoding rate is generally difficult to compute precisely due to the dynamics of video content with varying correlation. Effective methods for the refinement of imprecise side information are therefore important for improved decoding quality. In this paper, we propose to exploit the intrinsic property of channel coding algorithms in Wyner-Ziv video coding. The side information is refined via both the information-plane and the parity-plane bits, which rapidly increases the accuracy of refined SI. We use extrinsic information transfer (EXIT) chart analysis in order to estimate the variations of the mutual information in the iterative decoding. In particular, we characterize mutual information variations for punctured regular and irregular rate-compatible Low-Density Parity-check (LDPC) codes. Tracking the mutual information changes permits to decrease the coding rate of the information and parity bitstreams, while preserving the decoding quality. Simulations results confirm that our method improves on the decoding quality of recent distributed video coding algorithms, especially for high-motion sequences or at high coding rate regime.**

***Index Terms*—Distributed video coding, Side information, EXIT chart, LDPC codes.**

I. Introduction

Distributed video coding (DVC) has proposed a new paradigm in image and video compression [1] where effective and low complexity compression can be achieved if source correlation is properly exploited at decoder. Constructive solutions for distributed video coding are based on Wyner-Ziv (WZ) coding [2] [3]. In most existing WZ video coding schemes, the video frames are grouped into key and WZ frames [4]. The key frame is similar to an Intra-frame and is encoded using conventional video coding methods. The WZ frame coding is based on the channel coding principles, e.g., Turbo [5] [6], Low-Density Parity-Check (LDPC) [7] [8], and Fountain [9] codes, and parity data only is sent to the decoder for WZ frames. In the decoder side, side information (SI) is created from the key frame and other decoded frames through motion-compensated interpolation/extrapolation methods. The WZ frames are then decoded by joint processing of the SI and parity bits from the encoder. While such a process looks pretty attractive in many applications, WZ video coding has still not reached the compression efficiency performance of conventional video coding solutions despite important research efforts.

One of the limitations of DVC resides in the suboptimal quality of the side information used for decoding the WZ frames [6]. This has three main causes. First, the WZ data is composed of parity bits, which are decoded with the SI. The SI is estimated at decoder by interpolation or extrapolation from previously decoded key and WZ frames. The more correlation is the side information to the source, the higher the Slepian-Wolf [10] [11] and Wyner-Ziv compression performance. Second, the statistical dependency between the original frame and the SI is often seen as a virtual "correlation channel" [2] and the parity bits are used to correct the correlation noise through channel decoding. The efficiency of WZ coding has thus a strong dependence on the correct estimation of the number of parity bits, or equivalently on the accuracy of the correlation noise model. In practice, video sources exhibit a spatial and temporal correlation structure with highly varying statistics, which renders variable correlation models very useful but difficult to construct. Third, as video coding typically results in variable length code design, the channel coding used on DVC has to rely on the rate-compatible capabilities of practical channel coding algorithms. This typically limits the performance of DVC compared to the theoretical performance bounds. Overall, the performance depends on the SI accuracy, on the effective estimation of the channel rate, and on the error correction capabilities of channel codes. In this paper, we address the issues by refinement of the side information in order to improve the rate-distortion performance of WZ video coding.

Traditionally, the mechanisms for SI refinement and channel decoding are designed separately. We propose here to study the improvement of the side information by exploiting the intrinsic properties of channel decoding. In particular, we analyze the iterative operations of the LDPC channel decoder and the convergence of the iterative WZ decoding process with an Extrinsic Information Transfer (EXIT) chart analysis [12]. The change in extrinsic mutual information reflects the evolution of the correlation in the SI refinement process. Based on this analysis, the convergence properties of the iterative decoding can thus be explicitly taken into account in the SI refinement process. The correlation between refined side information and original information data can further be used to adjust the bit rate of the parity and information data streams. The decoding performance is finally optimized by formulating a conditional optimization problem, which is

This work is supported by the National Natural Science Foundation of China (61001194), and by the Beijing Natural Science Foundation (4122078).

W. Ji, Y. Chen are with the Beijing Key Laboratory of Mobile Computing and Pervasive Device, Institute of Computing Technology, Chinese Academic of Sciences, P.R. China (e-mail: jiwen@ict.ac.cn; yqchen@ict.ac.cn).

P. Frossard is with Signal Processing Laboratory (LTS4), Ecole Polytechnique Fédérale de Lausanne (EPFL) (e-mail: pascal.frossard@epfl.ch).

solved through an interior-point method [13]. We show by experiments that the exploitation of the intrinsic properties of channel coding techniques leads to improved performance in WZ video coding. In particular, our novel algorithm is able to outperform recent DVC schemes at high coding rate or for high motion video sequences. We finally note that our new SI refinement method is generic, such that it can be extended to other side information types and other channel coding algorithms different than LDPC.

The rest of the paper is organized as follows. In Section II, we briefly review the design of DVC algorithms with a special attention to LDPC-based algorithms and side information improvement methods. In Section III, we describe our new SI refinement method. In Section IV, we solve the problem of the side information refinement in a WZ video coding structure. In Section V, we give the solution of rate minimization in a WZ video coding structure. The experimental results are provided in section VI. Finally, the conclusion and future work are presented in Section VII.

II. Distributed Video Coding Overview

A. Distributed Video Coding Systems

We first give a quick overview of DVC systems, which are generally implemented with a feedback channel. The overall structure of DVC video codec is shown in Fig.1. The frame structure typically includes key frames (i.e., I-frames) and WZ frames. The key frames are traditionally intra-frame coded (for example with H.264/AVC and JPEG). The coding of WZ frames is the core of DVC, which consists of a quantizer followed by a Slepian-Wolf encoder which encodes the raw data [14] or the DCT coefficients [2] of the WZ image. In more details, the most popular WZ coding algorithms operate as follows: the raw frame undergoes a transform (such as DCT), and the resulting DCT coefficients are quantized. The quantized coefficients are represented with bitplanes, which form binary sequences encoded through channel coding (such as Turbo and LDPC). The resulting parity bits output is the final WZ coding results. At the other end, the decoder reconstructs the key frames, and the WZ frames with help of side information. It generates the SI in two steps. First, it interpolates or extrapolates the previously decoded frames. Then, it employs the same process as the encoder to generate the coefficient bitplanes, which are approximate versions of the bitplanes generated at encoder. Channel decoding is then performed with help of the transmitted parity bits and the SI to decode the WZ frame, which is reconstructed after inverse quantization and inverse transform. The decoder can request additional parity bits from the encoder through a feedback channel in order to decode the WZ frame successfully. This is called a "request-and-decode" process [2]. Although the feedback channel may induce high latency, it leads to easier optimization of the encoding rate and to more accurate and flexible SI in DVC systems.

B. WZ Video Coding Using LDPC and RC-LDPC

The first Wyner-Ziv coding schemes adopt Turbo coding [5] or Trellis codes [15]. Turbo coding is built using a parallel concatenation of recursive systematic convolutional codes, whose

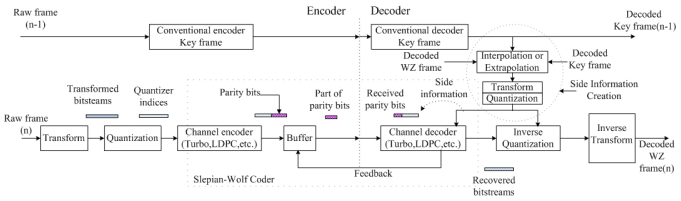


Fig. 1. Distributed video codec paradigm.

performance in term of BER is close to the Shannon limit [16]. It is proved that Turbo and LDPC codes, respectively, are used to approach arbitrary points on the SW bound [17]. The works in [18] [19] [20] show that LDPC codes might be a powerful alternative to turbo codes for DVC. The work in [21] proposes two classes of rate-adaptive distributed source codes design: rate-adaptive LDPC Accumulate (LDPCA) codes and Sum LDPC Accumulate (SLDPCA) codes, which perform within 10% and 5% of the Slepian-Wolf bound in the moderate and high rate regimes, respectively. And the work in [8] presents a rate-compatible LDPC-based SW coding scheme, which focuses on non-binary LDPC codes of moderate code length case. However, the main drawbacks of the above codes are the high decoding complexity and high latency, which may make them unsuitable in practice. More researches focus on the low complexity channel coding design while keeping the performance of DVC. In practice, it is also desirable to vary the rate of encoders. Then, puncturing the LDPC codes to provide different output bitrates and thus different error-correction ability is a common used method, which is called rate-compatible LDPC (RC-LDPC). It is proved that capacity-achieving codes of any rate can be constructed by puncturing some original LDPC code with a small enough rate [22]; this property is very suitable to variable video coding. Besides, it is worthy to note that punctured RC-LDPC shows advantages in low complexity rate-compatible coding provision. Thus, recently, the works in [23] and [24] provide a degree distribution design and puncturing method for SW coding. Those works show great potentials in using rate-compatible LDPC (RC-LDPC) codes to provide a flexible WZ video coding. However, much work remains to be done in order to construct an architecture that achieves good RD performance in RC-LDPC-based WZ video coding.

C. SI Refinement in Wyner-Ziv Coding

In general, one of the key steps in effective WZ coding consists in constructing accurate side information, which is very important for the decoding of the WZ frames. The important SI refinement step is achieved by updating and improving the SI during decoding. We can distinguish two main classes of methods.

The first class for SI generation and refinement uses motion compensated interpolation/extrapolation or hash-based motion estimation methods, which can be further classified into two sub-classes. One is using motion compensated interpolation/extrapolation at the decoder, and the other is using hash-based motion estimation method from the encoder.

Motion compensated interpolation/extrapolation SI refinement methods use the information available at decoder to

improve the side information. Typically, the generation and refinement of the SI mainly depend on the motion-compensated interpolation/extrapolation that is performed either in the temporal [25], spatial [26] or transform domain [6]. First, the temporal domain methods use motion compensation techniques in order to estimate the initial SI. Then they decode the SI and the parity data and keep on updating the SI during an iterative decoding process [25]. Second, refinement in the spatial domain is based on learning from previously decoding low-resolution data in order to refine the motion estimation at decoder, which in turn improves the quality of the SI for higher resolution data [26]. Third, in transform domain refinement methods, the decoder improves the SI accuracy in the DCT domain and corrects the data in different DCT bands through learning of the error patterns in the previously decoded bands [6]. Most the above methods are however based on the assumption that the motion in the video sequence is smooth [27]. For video sequences with high motion, the accuracy of the SI unfortunately drops quickly. Thus, some works have proposed to estimate motion trajectories with regressive methods [28] [7] and to use the motion trajectories for correcting the reconstructed pixels. This results however in high computational complexity at decoder and in low efficiency in sequences with quick or violent motions, long group of pictures (GoP) or noisy reference frames.

Hash-based motion estimation methods use extra information from the encoder in order to improve the side information. In particular, additional hash-like information is produced by the WZ encoder in order to improve the prediction accuracy. For example, the encoder can generate hash codewords corresponding to the coordinates of reference blocks [29]. It then sends the hash codewords and the parity bits together to the decoder. The purpose of hash codewords is to aid the decoder in accurately estimating the motion [30]. Common Hash-based generation methods include i) spatial domain hash codewords, where pixel subsampling is used to get the spatial block information, along with edge detection to extract moving objects and gradient direction estimation to extract block features; ii) frequency domain hash codewords that directly exploit information in the frequency domain (e.g., low frequency DCT coefficients) as additional information [29]. Most hash-based motion estimation methods are used to generate the SI reliably, such as adopting motion-compensated interpolation and quality enhancement [31], or using partial information to estimate the rate-distortion characteristic [32], or incorporating knowledge of the correlation channel statistics and extracting additional information from the hash [33], or using overlapped block motion estimation to create high quality SI [34]. For the SI refinement, the work in [35] proposes side-information-dependent (SID) model in the transform domain and gives an efficient online SID estimation. This method enables bit plane successive refinement, which leads to progressive improvement of the SI accuracy. Moreover, the work in [36] introduces hash-based overlapped block motion estimation and compensation (OBMEC) with hybrid subsampled matching (SSM) and block matching to further improve the quality of SI refinement.

The second class of methods refine the SI from the parity

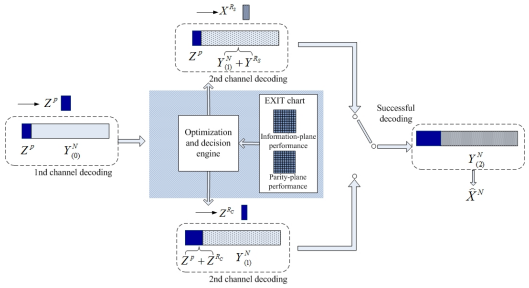


Fig. 2. Side information refinement process.

information. In their seminal paper [5], Aaron and Girod have proposed to lower the coding rates through puncturing the parity bits. This implies that the SI quality can be refined through additional parity bits subsets. When the channel decoding module uses the parity bits and the SI to decode the original frame, the decoder in fact corrects the errors due to the virtual "correlation channel" between the side information and the WZ frames. According to the principle of channel coding, additional parity bits therefore can help to improve the error correction ability of channel decoding. The methods that use additional parity bits are regarded as feedback channel-based DVC algorithms. If this feedback from the channel output is provided to the encoder during the refinement stage, it can significantly facilitate the implementation of optimum codes [37] [38]. Thus, most WZ video coding solutions use a feedback channel (FC) based decoder rate control strategy in order to adjust the bitrate and correct the errors in the SI. The feedback-based architectures are based either on pixel [2] [14] or transform domain WZ coding [39] [40]. An evaluation of the systems based on feedback channels is proposed in [41]. Finally, it should be noted that the use of a feedback channel between the encoder and the decoder defines an interactive procedure that is not always possible in some DVC applications. In these cases, effective encoder rate control [42] or residual frame estimation based on machine learning methods [43] might be used for improving side information.

Two main bottlenecks remain in the critical steps of improving the SI. First, the video signals are generally pretty complex and hard to estimate, so that accurate SI is hard to construct at decoder. Second, the parity bits of the WZ frame may not be sufficient for successful decoding when the error correction capabilities are exceeded due to the low quality of the SI. These two problems are generally addressed independently. However, this separation is suboptimal, as both problems are intimately related. For example, inaccurate SI might result in failure of the LDPC decoder when the channel coding rate is not sufficient. We propose in this paper a novel method to consider both problems jointly. We build on our recent work [44] and study jointly the LDPC decoding process and the SI refinement in WZ video coding. In particular, we exploit the intrinsic property of channel coding for improving the WZ decoding performance.

III. SI Refinement Model

A. SI Refinement Formulation

Let $\mathbf{X}^N = \{x_1, ..., x_N\}$ be a binary source sequence of length N. Let $\mathbf{Y}^N = \{y_1, ..., y_N\}$ and $\hat{\mathbf{X}}^N = \{\hat{x}_1, ..., \hat{x}_N\}$ be the SI and

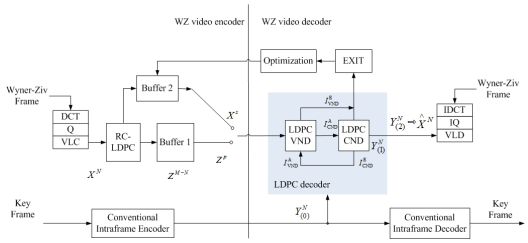


Fig. 3. WZ video coding with EXIT chart-based side information refinement.

reconstructed sequence at decoder, respectively. Define $\mathbf{Z}^P = \{z_1, ..., z_P\}$ as the subset parity data from the encoder, which is also in binary. In the WZ decoding process, the decoder uses the side information $\mathbf{Y}^N$ and the parity bits $\mathbf{Z}^P$ from the encoder to form an estimate $\hat{\mathbf{X}}^N$ of the source sequence $\mathbf{X}^N$. Let $\mathbf{Y}^N_{(0)}$ be the estimated SI. The initial SI estimate $\mathbf{Y}^N_{(0)}$ and the parity bits $\mathbf{Z}^P$ are both used as the inputs of channel decoder to form the first refined SI $\mathbf{Y}^N_{(1)}$, as follows,

$$\{y_1, ..., y_N\} \times \{z_1, ..., z_P\} \longrightarrow \mathbf{Y}^N_{(1)} \tag{1}$$

The SI is further refined along the decoding process, as illustrated in Fig.2. In particular, the SI can be refined via either the information or the parity plane, and the choice of the method is determined through an optimization engine. The choice is typically oriented towards the plane that results in the fastest channel decoding convergence through EXIT chart analysis. We describe each method below.

Definition 1 (Information-plane SI Refinement): A refined SI codeword for the source $\mathbf{X}$ is obtained by additional information bits from the encoder. In particular, the encoder sends the information-plane bits $\mathbf{X}^{R_s} = \{q_1, ..., q_{R_s}\}$, where R_s is the number of original WZ bits, which is used to update the SI. First, $\mathbf{X}^{R_s}$ is used for updating $\mathbf{Y}^N_{(1)}$ into $\hat{\mathbf{Y}}^N_{(1)}$; then, joint decoding of $\hat{\mathbf{Y}}^N_{(1)}$ and the original parity bits $\mathbf{Z}^P$ leads to the refined SI $\mathbf{Y}^N_{(2)}$, as follows,

$$\phi : \begin{cases} \mathbf{Y}^N_{(1)} + \mathbf{X}^{R_s} \longrightarrow \hat{\mathbf{Y}}^N_{(1)} \\ \hat{\mathbf{Y}}^N_{(1)} \times \{\mathbf{Z}^P\} \longrightarrow \mathbf{Y}^N_{(2)} \end{cases} \tag{2}$$

Definition 2 (Parity-plane SI Refinement): A refined SI codeword for the source $\mathbf{X}$ is obtained by refining the SI $\mathbf{Y}$ with additional parity bits. The encoder sends the parity plane $\mathbf{Z}^{R_c}$, where R_c is the number of additional parity bits. The original parity $\mathbf{Z}^P$, the new parity plane $\mathbf{Z}^{R_c}$ and the SI $\mathbf{Y}^N_{(1)}$ are then jointly used to provide refined SI $\mathbf{Y}^N_{(2)}$ as follows,

$$\psi : \mathbf{Y}^N_{(1)} \times \{\mathbf{Z}^{R_c} + \mathbf{Z}^P\} \longrightarrow \mathbf{Y}^N_{(2)} \tag{3}$$

The decoding performance can be improved either from information-plane or parity-plane. The problem becomes to determine which plane can better improve the mutual information between $\mathbf{X}$ and $\hat{\mathbf{X}}$, such that the channel decoding converges faster. Since each refinement scheme leads to different performance under different correlation characteristics, we propose to optimize both schemes in next paragraphs. We use an analysis of the correlation information to improve the SI refinement strategies of Definitions 1 and 2.

B. Correlation Noise Model in SI Refinement

The channel decoding process starts by converting the SI to virtual channel input information. Through tracking the mutual information change, the convergence performance of channel decoder (the successful decoding) can be predicted [45] [12]. We introduce an EXIT chart to analyze the mutual information changes due to information-plane and parity-plane bitstreams, respectively. The mutual information is a function of correlation noise threshold. Since the correlation between $\mathbf{X}$ and $\mathbf{Y}$ is modeled as the input and output of a noise channel with variance σ_n^2, better side information quality denotes a lower noise σ_n^2, which in turn leads to a lower rate for achieving a given decoding quality. In our system, each SI refinement step leads to the update of the channel noise variance σ_n^2. In order to obtain the noise estimate, we design a pilot-assisted correlation channel estimation algorithm, where the pilot symbol sequence $\{WZ_1, ..., WZ_\eta\}$ is a binary sequence of η symbols (a small subset of the original frame). This pilot symbol sequence is transmitted together with the parity data. The decoder correlates the received sequence with the pilot symbol sequence and estimates the initial σ_n^2 as

$$\sigma_0^2 = E\Big[(WZ_k - S_k^{'})^2\Big] - \Big[E(WZ_k - S_k^{'})\Big]^2 \tag{4}$$

where the $S_k^{'}, k \in [1, \eta]$ represents the corresponding value in the current SI. Next, we deduce the noise variances during SI refinement.

We first consider the information-plane SI refinement. When the encoder sends the information-plane bits $\mathbf{X}^{R_s} = \{q_1, ..., q_{R_s}\}$, the information bits can directly be used in SI refinement. The correlation between $\mathbf{X}^N$ and $\mathbf{Y}^N$ is improved; accordingly, the noise variance decreases. We denote this decrement by $\Delta\sigma_1^2$, which leads to

$$\Delta\sigma_1^2 = \frac{1}{N}[(q_1 \oplus y_1)^2 + (q_2 \oplus y_2)^2 + ... + (q_{R_s} \oplus y_{R_s})^2] \tag{5}$$

In this case, the noise variance directly decreases without channel decoding process.

Now, let $p_{e,s} = F_s(\mathring{r}_c, \sigma_n^2)$, where $p_{e,s}$ is the bit-error-rate (BER) of the channel decoding, and $\mathring{r}_c = N/(N+P)$ is the channel coding rate. It describes the iterative decoder's error probability under fixed information bitrate, which is a function of both coding rate and noise variance. The function $F_s(\cdot, \cdot)$ has no closed-form expression, but can be estimated empirically. Once the correlation noise has been updated according to Eq (5), the BER is modified as $p_{e,s} = F_s(\mathring{r}_c, \sigma_n^2 - \Delta\sigma_1^2)$.

The decoder then uses the updated SI and the parity bits in order to form the refined SI. Since both $\mathbf{Y}$ and $\hat{\mathbf{X}}$ in WZ video coding are computed in the Galois field $\mathbb{GF}(2)$ for channel decoding, the correlation noise variance after decoding is

$$\breve{\sigma}_n^2 = \frac{1}{N}(p_{e,s} \times N) = p_{e,s} = F_s(\mathring{r}_c, \sigma_n^2 - \Delta\sigma_1^2) \tag{6}$$

The evaluation of the correlation noise permits to analyze the performance of the decoder. If $\breve{\sigma}_n^2 < \sigma_{th}^2$, no additional bits are needed because the current information plane bits are sufficient for the decoding, where σ_{th}^2 be the noise threshold of successful channel decoding.

We now consider the SI refinement from parity-plane bits, as given in Definition 2. When the encoder sends the parity data $\mathbf{Z}^{R_c}$, the BER decreases after channel decoding, accordingly, the correlation between $\hat{\mathbf{X}}$ and $\mathbf{Y}$ is improved and the noise variance decreases. Similarly to the above analysis, we define $p_{e,c} = F_c(\mathring{r}_c, \sigma_n^2)$ as the empirical formula that describes the iterative decoder's error probability under fixed parity bitrate $\mathring{r}_c$. Then, the noise variance decrement is

$$\Delta\sigma^2 = \sigma_n^2 - \frac{1}{N}(p_{e,c} \times N) = \sigma_n^2 - p_{e,c} \tag{7}$$

Accordingly, the current correlation noise variance after this decoding is

$$\breve{\sigma}_n^2 = \sigma_n^2 - \Delta\sigma^2 = p_{e,c} = F_c(\mathring{r}_c, \sigma_n^2) \quad (8)$$

Finally, if $\breve{\sigma}_n^2 < \sigma_{th}^2$, no additional bits are needed because the parity bits are sufficient for decoding.

Obviously, the SI refinement algorithm includes two alternative refinement paths: one from information-plane and the other from parity-plane. Both schemes are intimately linked with the performance of channel decoding. In the next section, we analyze this performance in WZ coding with LDPC codes.

IV. WZ Video Coding with Refined SI

In this section, we present the WZ video coding architecture with the proposed SI refinement process. We present the use of rate-compatible LDPC (RC-LDPC) to provide flexible WZ video coding. We propose to exploit the intrinsic property of channel coding algorithm in WZ video coding for refining the SI. In particular, we estimate the mutual information variations due to the SI refinement in LDPC decoding through an EXIT chart analysis approach. Through this analysis, we can define the rates of information and parity bits that are needed for proper decoding. This results in better rate-distortion performance in WZ video coding.

A. WZ Frame Codec Structure Embedded with Proposed SI Refinement

We consider the WZ video coding architecture illustrated in Fig.3. In the WZ frame coding stage, the DCT, quantization and variable-length coding (VLC) follow the key (intra) frame coding process. Then, the binary source sequence is encoded as the parity output of the RC-LDPC encoder. Let $\mathbf{X}^N = \{x_1, ..., x_N\}$ be the binary source sequence. The LDPC encoder, with codeword length M, is characterized by a $N \times (M-N)$ generator matrix $\mathbb{G}$. The parity bits are generated from $\mathbf{Z}^{M-N} = \mathbf{X}^N\mathbb{G}$. With RC-LDPC, the variable code rate $\mathring{r}_c = \frac{N}{M}$ can be achieved by puncturing techniques. It generates the compatible output $\mathbf{Z}^P = \{z_1, ..., z_P\}$, where $P \leq M-N$. The output of RC-LDPC encoder is cached into two buffers, one is for parity plane bits $\mathbf{Z}^P$ and the other is for information plane bits $\mathbf{X}^s$ (this part can also use a Hash function to further improve compressibility).

At the decoder, the parity data $\mathbf{Z}^P$ from the encoder and the initial SI $\mathbf{Y}_{(0)}^N$ is used as the input of the LDPC decoder. The initial SI is obtained from the key frame. After the first LDPC decoding, the SI is updated to $\mathbf{Y}_{(1)}^N$. Then, the refined SI $\mathbf{Y}_{(2)}^N$ can be obtained from the information part by 1) decoding the current parity plane $\{\mathbf{Z}^{R_c} + \mathbf{Z}^P\}$ and the previous SI version $\mathbf{Y}_{(1)}^N$; or 2) decoding the previous and current information plane $\{\mathbf{Y}_{(1)}^N + \mathbf{X}^{R_s}\}$, and the previous parity plane bits $\mathbf{Z}^P$ from the encoder. The decoder is allowed to run a maximum of L iterations, and if convergence has not been reached, decoding is considered to have failed. In order to predict the convergence performance of the LDPC decoding independently, an EXIT module is introduced through keeping track of the change in mutual information. This helps to properly adapt the bitrates of information or parity planes sent by the encoder.

It is worth to mention that we do not use motion estimation module in the video decoder. Instead, the initial SI $\mathbf{Y}_{(0)}^N$ is a simple copy from the results of key frame. Those SI generation algorithms such as motion-compensated interpolation/extrapolation can work together with proposed structure to further improve the accuracy of $\mathbf{Y}_{(0)}^N$.

Finally, we recall the major changes between the framework (EXIT-SI-WZ) in Fig.3 and the conventional DVC architecture (C-WZ) in Fig.1: (1) the binary source is entropy coded data in EXIT-SI-WZ, and is DCT coded in C-WZ; (2) EXIT-SI-WZ employs the I frame data as the initial SI such that it does not need the motion estimation module in the decoder. These differences have two main consequences. First, if the binary data is DCT data, the decoder should also use the DCT data and may need to support lower error correction ability, and the WZ decoder needs entropy decoding or other modules. Second, if the binary data is entropy coded, the decoder can directly use the Intra data; the channel coding module needs to support stronger error correction ability, however the WZ decoder is simpler. From the viewpoint of channel coding, the difference between the DCT and entropy coded data resides in the correlation process, which determines the bit-length of coding results.

B. SI Refinement During RC-LDPC Decoding

The RC-LDPC decoder is characterized by a $P \times (N+P)$ parity check matrix $\mathbb{H}$. The output $\hat{\mathbf{X}}^N$ is decoded from the available parity data $\mathbf{Z}^P$ and the refined SI. The SI refinement process is actually embedded in the LDPC decoding algorithm. Generally, the decoding of LDPC codes can be represented by a message passing algorithm on the bipartite graph. The message between variable and parity check nodes represents the Log Likelihood Ratio (LLR) of a symbol 0 or 1. The bipartite graph has $N+P$ variable nodes corresponding to the codewords $\hat{\mathbf{Y}}^N$ and $\mathbf{Z}^P$ along with P check nodes corresponding to the codeword $\mathbf{Z}^P$. Let $\mathcal{P}(j)$ denote the set of check nodes connected to the variable node j; $\mathcal{P}(j)\backslash i$ represents the set $\mathcal{P}(j)$ excluding the i-th variable node. Accordingly, let $\mathcal{Q}(i)$ denote the set of variable nodes that participate in the i-th parity-check node; $\mathcal{Q}(i)\backslash j$ represents the set $\mathcal{Q}(i)$ excluding the j-th check node. Since the statistical dependency between the source binary sequence $\mathbf{X}$ and the SI $\hat{\mathbf{Y}}$ can be regarded as a virtual correlation noise channel, the channel model between $\mathbf{X}$ and $\hat{\mathbf{Y}}$ complies with the Laplacian distribution [35] [46] [47], $f_{x|y}(x) = \frac{\alpha}{2}e^{-\alpha|x-y|}$, where α is the model parameter, given by correlation noise variance $\sigma_n^2 = \frac{2}{\alpha^2}$. Recall that the LLR of a binary valued random variable can be written as

$$f_j = \ln\frac{p(x_j=0|Y)}{p(x_j=1|Y)} = \ln\frac{\frac{\alpha}{2}e^{-\alpha|0-y_j|}}{\frac{\alpha}{2}e^{-\alpha|1-y_j|}} = \alpha(1-2y_j) = \sqrt{\frac{2}{\sigma_n^2}}(1-2y_j) \quad (9)$$

The LDPC decoding can typically be performed with efficient implementations such as [48]. It follows an iterative process, after initialization where each variable node j is assigned a posteriori LLR: $L(x_j) = \ln\{p(x_j=0|y_j)/p(x_j=1|y_j)\} = f_j$. Then, after iteratively check-node update, variable-node update, and decision processes, we can get the final output of the decoder, $\hat{\mathbf{X}}$.

Besides, we control both information-plane and parity-plane data request through the feedback channel. In order to reduce the load, we design a 2-bits feedback scheme. The first bit

represents whether the feedback data is needed. For example, '1' denotes that the decoder needs the feedback data; while '0' expresses that current LDPC decoding is successful and needs no additional data. The second bit demonstrates that the decoder needs information-plane or parity-plane data, which is also distinguished by '1' and '0'. In the WZ encoder, assume there are W bits entered into the RC-LDPC encoder in a frame. That is, the RC-LDPC encoder runs $\frac{W}{N}$ times for a WZ frame. Then, the total length of feedback is $\frac{W}{N} \times Num \times 2$ bits, where Num is the request times.

Next, we provide below details about the EXIT chart analysis that permits to characterize the performance of the SI refinement, and to compute the necessary rates for information and parity bits.

C. Mutual Information Variations in RC-LDPC Decoding

In this work, we introduce an EXIT chart analysis [12] [49] to deduce the mutual information variations in the LDPC decoding process. It helps to estimate the match between the information and parity planes and to predict the decoding performance. The sets of variable and check nodes in the LDPC decoder are referred as the variable-node decoder (VND) and check-node decoder (CND), respectively. Decoding is performed by iteratively exchanging extrinsic information between the VND and CND [50] [12]. During each iteration, we compute the mutual information I^A_{VND} and I^E_{VND} of the input and output of VND, which represent the mutual information changes of the variable nodes. We also compute the mutual information I^A_{CND} and I^E_{CND} of the input and output of CND, which represent the mutual information changes of the check nodes. During iterative decoding, the extrinsic output of the CND is fed back to the input of the VND as a priori knowledge, that is $I^A_{VND} = I^E_{CND}$. The iterative decoder for a LDPC code of length M can be viewed as a Tanner graph that has M variable nodes and $M - N$ check nodes. A variable node is involved in d_v check nodes and a check node checks d_c variable nodes. Generally, the ensemble of regular LDPC codes is denoted by $\mathcal{G}(M, N, d_v, d_c, r)$, where the design rate is $r = N/M$. For regular LDPC codes, the EXIT function of a degree-d_v variable node is given by

$$I^E_{\text{VND}}(I^A_{\text{VND}}, d_v, \sigma_n^2, r) = J\left(\sqrt{(d_v - 1)[J^{-1}(I^A_{\text{VND}})]^2 + \sigma_n^2}\right) \tag{10}$$

where σ_n^2 is the noise variance. The functions $J(\cdot)$ and $J^{-1}(\cdot)$ are given in [12], as $J(\sigma_{ch}) = 1 - \int_{-\infty}^{\infty} \frac{e^{-(\xi-\sigma_{ch}^2/2)^2/2\sigma_{ch}^2}}{\sqrt{2\pi\sigma_{ch}^2}} \cdot \log_2[1 + e^{-\xi}]d\xi$. Further details on the interpretation of the EXIT function are given in [49] and approximate expressions are given in the appendix in [12].

We now extend the EXIT chart analysis and the analysis of the mutual information to RC-LDPC decoding with respectively punctured regular and irregular ensembles used in WZ coding.

Theorem 1: An ensemble of punctured regular RC-LDPC codes is denoted by $\mathcal{G}(M, N, d_v, d_c, r_p)$, where $\frac{N}{M} = \frac{d_c}{d_v}$. Then the mutual information function of inner VND I^E_{VND} is

$$\begin{aligned} I^E_{\text{VND}}(I^A_{\text{VND}}, d_v, \sigma_n^2, r_p) =& (1 - \frac{r}{r_p}) \cdot J\left(\sqrt{(d_v - 1)[J^{-1}(I^A_{\text{VND}})]^2}\right) \\ &+ \frac{r}{r_p} \cdot J\left(\sqrt{(d_v - 1)[J^{-1}(I^A_{\text{VND}})]^2 + \sigma_n^2}\right) \end{aligned} \tag{11}$$

Proof: The proof is given in Appendix A. □

We now propose two corollaries that define the mutual information in the output and input CND functions I^E_{CND} and I^A_{CND}.

Corollary 1.1: An ensemble of punctured regular RC-LDPC codes is denoted by $\mathcal{G}(M, N, d_v, d_c, r_p)$. Then the mutual information function of output CND function is

$$I^E_{\text{CND}}(I^A_{\text{CND}}, d_c, \sigma_n^2, r_p) = 1 - J\left[\sqrt{d_c - 1} \cdot J^{-1}(1 - I^A_{\text{CND}})\right] \tag{12}$$

Corollary 1.2: An ensemble of punctured regular RC-LDPC codes is denoted by $\mathcal{G}(M, N, d_v, d_c, r_p)$. Then the mutual information function of input CND function is

$$I^A_{\text{CND}}(I^E_{\text{CND}}, d_c, \sigma_n^2, r_p) = 1 - J\left[\frac{J^{-1}(1 - I^E_{\text{CND}})}{\sqrt{d_c - 1}}\right] \tag{13}$$

The proofs of the corollaries are similar to the proof of Theorem 1 given in Appendix A.

We illustrate the above results in an example with a regular LDPC mother code of block length 2048 and rate 0.5. The degree pair is $d_v = 3$ and $d_c = 6$. This mother code is used to construct the RC-LDPC coder in WZ video coding. We show both the EXIT analysis curves as well as the simulation results for this RC-LDPC code. First, we implement half rate mother LDPC codes with a regular structure. Then, we can accurately predict the convergence threshold under each code rate through tracking the mutual information changes. We calculate the EXIT analysis curves using the results of Theorem 1 and Corollary 1.2. Since $I^A_{VND} = I^E_{CND}$, we show the EXIT analysis curves of Eq.(13) and Eq.(11) visually in one figure, as illustrated in Fig.4. The blue line is the CND curve of Eq.(13) and the red line is the VND curve Eq.(11) under rates $\mathring{r}_c$=0.5, 0.65, 0.8, respectively. We can observe that with the regular LDPC mother code of length 2048 and rate 0.5, when E_b/N_0 = 1.2dB, the two curves match closely. This shows that the decoder can keep an open convergence tunnel when the SNR is greater than 1.2. Thus, the noise threshold is 1.2dB for rate $\mathring{r}_c$=0.5. This means that, when the noise is below the thresholds, the LDPC decoding can be successful in the WZ video decoding process. Furthermore, we see that, when $\mathring{r}_c$=0.65 and 0.8, the noise thresholds rise to 2.5dB and 4dB, respectively.

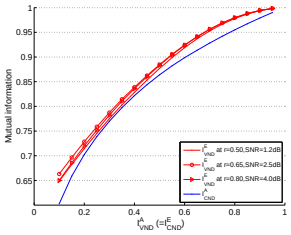


Fig. 4. EXIT chart analysis for different rate $\mathring{r}_c$ under regular LDPC.

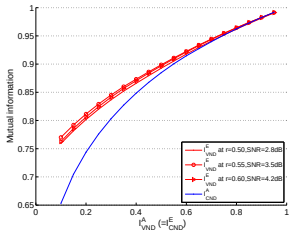


Fig. 5. EXIT chart analysis for different rate $\mathring{r}_c$ under irregular LDPC.

Although the puncturing scheme is tailored to regular codes, it is also directly applicable to irregular parent ensembles. In the case of an irregular LDPC code, its Tanner graph $\mathcal{G}$ has varying variable node degrees and varying check node degrees. The degree distributions of these two types of nodes can be represented as two polynomials, $\lambda(x) = \sum_i \lambda_i x^{i-1}$, where λ_i indicates the fraction of edges that are connected to variable nodes of degree i; and polynomial $\rho(x) = \sum_i \rho_i x^{i-1}$, where ρ_i is the fraction of edges incident to check nodes of degree i.

The evaluation of the mutual information in irregular LDPC codes is given by the following theorem.

Theorem 2: An ensemble of punctured irregular RC-LDPC codes is denoted by $\mathcal{G}(M, N, \lambda(x), \rho(x), r_p)$. The mutual information function of inner VND is

$$I^E_{\mathrm{VND}}(I^A_{\mathrm{VND}}, \lambda, \sigma_n^2, r_p) = \sum_{i=1}^{d_v} \left\{ \lambda_i(\delta_i) \cdot J\left(\sqrt{(i-1)[J^{-1}(I^A_{VND})]^2} \right) + \lambda_i(1-\delta_i) \cdot J\left(\sqrt{(i-1)[J^{-1}(I^A_{VND})]^2 + \sigma_n^2} \right) \right\} \quad (14)$$

where $\delta(x)$ is the puncturing distribution of the irregular LDPC code, and $\delta(x) = \delta_1 + \delta_2 x + ... + \delta_{d_v} x^{d_v - 1}$.

Proof: The proof of Theorem 2 is given in Appendix B. □

We further have the following corollaries that define the mutual information in the CND functions for irregular LDPC codes.

Corollary 2.1: An ensemble of punctured irregular RC-LDPC codes is denoted by $\mathcal{G}(M, N, \lambda(x), \rho(x), r_p)$. Then the mutual information function of the output CND function is

$$I^E_{\mathrm{CND}}(I^A_{\mathrm{CND}}, \rho, \sigma_n^2, r_p) = 1 - \sum_{j=1}^{d_c} \rho_j \cdot J\left[\sqrt{j-1} \cdot J^{-1}(1 - I^A_{CND}) \right] \quad (15)$$

Corollary 2.2: An ensemble of punctured irregular RC-LDPC codes is denoted by $\mathcal{G}(M, N, \lambda(x), \rho(x), r_p)$. Then the mutual information function of the input CND function is

$$I^A_{\mathrm{CND}}(I^E_{\mathrm{CND}}, \rho, \sigma_n^2, r_p) = 1 - J\left[\frac{J^{-1}(1 - I^E_{\mathrm{CND}})}{\sum_{j=1}^{d_c} \rho_j \cdot \sqrt{j-1}} \right] \quad (16)$$

The proof of the corollaries 2.1 and 2.2 are similar to the proof of Theorem 2 given in Appendix B.

We now illustrate the above results with an irregular LDPC mother code of block length 8100 and rate 0.5. The degree distribution pair is $\lambda(x) = 0.5x + 0.3x^2 + 0.2x^7$ and $\rho(x) = x^6$. We will use this mother code to construct RC-LDPC in performance analysis of proposed WZ video coding. We show both the EXIT curves as well as simulation results for RC-LDPC codes in Fig.5. First, we implement half rate mother LDPC codes with an irregular structure. Then, we can accurately predict the convergence threshold under each code rate through tracking the mutual information changes. We calculate the EXIT curves using the results of Theorem 2 and Corollary 2.2. Fig.5 shows pairs of EXIT curves for RC-LDPC codes at rates $\mathring{r}_c$=0.5, 0.55, 0.6, respectively. The blue line is the CND curve in Eq.(16) and the red line is the VND curve in Eq.(14) under different rates. We can observe that with the irregular LDPC mother code of length 8100 and rate 0.5, when E_b/N_0 = 2.8dB, the two curves match closely. This demonstrates that the decoder can keep an open convergence tunnel when the SNR is greater than 2.8dB. Thus, the noise threshold is 2.8dB for a rate $\mathring{r}_c$=0.5. When $\mathring{r}_c$=0.55 and 0.6, the noise thresholds increase to 3.5dB and 4.2dB, respectively.

The above results are used to predict the performance of the LDPC decoder, and thus to choose proper channel rates. The performance of LDPC codes typically exhibits a threshold phenomenon under iterative decoding. The LDPC decoding over various channels is successful only when the noise level is below some threshold. Conventionally, the convergence behavior of the decoder is analyzed through using density evolution (DE) techniques. The EXIT chart analysis that we propose to use in WZ video coding is a simpler method than the DE algorithm; it also provides better accuracy in prediction of the convergence threshold [51]. In this work, the objective of the EXIT functions is to determine the noise thresholds of the RC-LDPC codes under different rates. The noise threshold for successful decoding is finally computed by

$$\sigma^2_{th} = \inf_{\sigma_n} \left\{ \sigma_n^2 \middle| I^E_{\mathrm{VND}} > I^A_{\mathrm{CND}} \right\} \quad (17)$$

where the condition $I^E_{\mathrm{VND}} > I^A_{\mathrm{CND}}$ is determined with the EXIT chart analysis.

In order to lower the complexity, the EXIT chart analysis results are computed off-line and are stored in a lookup table, noted as EXIT-lookup-table. This lookup table stores the relations between noise threshold σ^2_{th} for successful decoding and rate $\mathring{r}_c$ (such as the points in Fig.4 and Fig.5). For example, for irregular LDPC codes, the σ^2_{th} at $\mathring{r}_c$ = $\{0.9, 0.8, 0.72, 0.66, 0.56, 0.55, 0.53, 0.5\}$.

V. Rate Minimization in WZ Video Coding

In WZ video coding architectures with SI refinement through feedback channel, the decoder helps the encoder to obtain lower coding rate. We thus address here the typical problem of determining the minimal number of bits, as measured by the rate R, in order to reach a certain video quality. We cast this problem in our DVC system as minimizing the rate while achieving successful LDPC decoding. Based on the EXIT chart analysis of Section IV, this problem can be further interpreted as minimizing the number of bits while maintaining the opening of the convergence tunnel. This is equivalent to setting the condition that the output mutual information I^E_{VND} of the VND is always larger than the input mutual information I^A_{CND} of the CND. Thus, we can express the condition for an open convergence tunnel as $I^E_{\mathrm{VND}} \geq I^A_{\mathrm{CND}} + \varepsilon$, where ε is a small positive value, which controls the gap between the two EXIT function curves. Through Definition 1 and 2, the rate R is composed of two parts, the information-plane and parity-plane bitrates, noted as R_s and R_c. Then, the optimization problem for minimizing the number of coding bits in the WZ frame is expressed as

$$\min \quad R_s + R_c \quad (18)$$

$$\text{s.t.} \quad I^E_{\mathrm{VND}} \geq I^A_{\mathrm{CND}} + \varepsilon \quad (19)$$

We now solve the optimization problem for irregular LDPC codes. The regular codes can be regarded as a special case. By substituting Eq. (14) and Eq. (16) in Eq. (19), we obtain

$$\sum_{i=1}^{d_v} \lambda_i \cdot J\left[\sqrt{(i-1)(J^{-1}(I^A_{\mathrm{VND}}))^2 + \frac{4}{\sigma_n^2}} \right] \geq 1 - J\left[\frac{J^{-1}(1 - I^E_{\mathrm{CND}})}{\sum_{j=1}^{d_c} \rho_j \cdot \sqrt{j-1}} \right] + \varepsilon \quad (20)$$

As shown in Fig.3, the mutual information at the input of the VND is equal to the mutual information at the output of CND, that is, $I^A_{\mathrm{VND}} = I^E_{\mathrm{CND}}$. Let $(I^{A,\mathrm{target}}_{\mathrm{VND}}, I^{E,\mathrm{target}}_{\mathrm{CND}})$ be the target convergence point in EXIT curves, which corresponds to the noise threshold σ^2_{th} of the LDPC decoder. Then, using an interior-point method[1], the EXIT functions can be sampled at a set of discrete points $I^E_{\mathrm{CND},k} \in [0, I^{E,\mathrm{target}}_{\mathrm{CND}}]$, where $k \in [1, K]$ indexes the discrete points and $I^E_{\mathrm{CND},K} = I^{E,\mathrm{target}}_{\mathrm{CND}}$. Then, the continuous convergence condition in Eq.(20) can be replaced by

$$\sum_{i=1}^{d_v} \lambda_i \cdot J\left[\sqrt{(i-1)(J^{-1}(I^E_{\mathrm{CND},k}))^2 + \frac{4}{\sigma_n^2}} \right] \geq 1 - J\left[\frac{J^{-1}(1 - I^E_{\mathrm{CND},k})}{\sum_{j=1}^{d_c} \rho_j \cdot \sqrt{j-1}} \right] + \varepsilon_k, \quad k \in [1, K] \quad (21)$$

where $\varepsilon_k = \varepsilon, \forall k$. These samples are separated by small steps τ (e.g., τ = 0.001), such that they can reflect the shape and

[1]It shows good performance in iterative equalization [52].

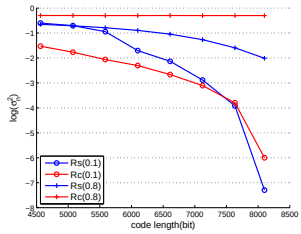


Fig. 6. The relations between rates and correlation noise variances.

trajectory of the original EXIT function curve. This condition relates the mutual information to the noise variance on the virtual correlation channel.

Next, the rates R_s and R_c in Eq.(18) are functions of the correlation noise variance σ_n^2. The accurate expressions of $R_s(\sigma_n^2)$ and $R_c(\sigma_n^2)$, which are also the inverse functions of $F_s(\cdot,\cdot)$ and $F_c(\cdot,\cdot)$ in Section III, depend on the performance of a given LDPC codec. Their estimation is dependent on the prediction of the convergence threshold of a code with given check and variable node degree profiles. In this work, we use a curve fitting algorithm in order to rapidly deduce the performance of a given LDPC decoder:

$$\begin{cases} R_s(\sigma_n^2) = \arg_{\langle \hat{R}_s, \hat{\sigma}_n^2 \rangle} \inf\left\{ \left| \hat{R}_s(\hat{\sigma}_n^2) - \sum_{i=0}^{m} \binom{m}{i} \left(\frac{\sigma_{n,0}^2 - \hat{\sigma}_n^2}{\sigma_{n,0}^2}\right)^{m-i} \left(\frac{\hat{\sigma}_n^2}{\sigma_{n,0}^2}\right)^{i} R_{s,i} \right| \right\} \\ R_c(\sigma_n^2) = \arg_{\langle \hat{R}_c, \hat{\sigma}_n^2 \rangle} \inf\left\{ \left| \hat{R}_c(\hat{\sigma}_n^2) - \sum_{i=0}^{m} \binom{m}{i} \left(\frac{\sigma_{n,0}^2 - \hat{\sigma}_n^2}{\sigma_{n,0}^2}\right)^{m-i} \left(\frac{\hat{\sigma}_n^2}{\sigma_{n,0}^2}\right)^{i} R_{c,i} \right| \right\} \end{cases} \tag{22}$$

where $\hat{\sigma}_n^2, R_{s,i}$ and $R_{c,i}$ are the values of the discrete points on noise variance, information bitrate, and channel coding rate by simulation, respectively.

It simplifies the estimations of $R_s(\sigma_n^2)$ and $R_c(\sigma_n^2)$. It still has no closed-form expression, and the performance of RC-LDPC codes depends on the degree distribution of the mother code for given rates R_c and R_s. We illustrate the curve fitting process with the RC-LDPC of Section IV. We test two cases with $\sigma_0^2 = \{0.1, 0.8\}$ and $\mathring{r}_c = \{0.9, 0.8, 0.72, 0.66, 0.56, 0.55, 0.53, 0.5\}$. Fig.6 plots the variance σ_n^2 as the functions of two coding bitrates R_s and R_c, respectively. The blue lines correspond to the results of the information-based refinement (Definition 1, with different R_s, and corresponding to Eq.(6)). The red lines correspond to the parity-based refinement (Definition 2, with different R_c, and corresponding to Eq.(8)). Based on the fitting algorithm for Eq. (22) , the simulation results suggest that it is reasonable to model $\log(\sigma_n^2)$ with the simple relation $\log(\sigma_n^2) = \sum_{i=0}^{6} a_{s,i}(\sigma_0^2) R_s^i$ and $\log(\sigma_n^2) = \sum_{j=0}^{6} a_{c,j}(\sigma_0^2) R_c^j$, where $a_{s,i}$ and $a_{c,i}$ are fitting coefficients estimated with a Monte Carlo method. As a result, both $R_s(\sigma_n^2)$ and $R_c(\sigma_n^2)$ are convex.

Fig.6 also demonstrates that, when the correlation between the SI and the WZ is weak, the information-plane SI can provide better performance alternatively, when the correlation is strong, the parity-plane SI provides better performance. For example, when $\sigma_0^2 = 0.1$, the correlation is strong, a few parity bits are sufficient to correct this correlation noise. When $\sigma_0^2 = 0.8$, the correlation is weak, the information bits are more efficient because they are able to directly decrease the noise. Thus, we can adjust the error correction results of LDPC codes through suitable bit-plane selection.

We build a lookup table with more accurate rate relations to speed up our rate administration processing, noted as performance-lookup-table. In more detail, this lookup table stores the relations between rates and correlation noise variances (such as the points in Fig.6). For example, when the initial correlation noise variance is in a set of discrete points $\sigma_0^2 \in [0.1, 0.8]$ in step-size 0.1, the lookup table records the (σ_n^2, R_c) and (σ_n^2, R_s) pairs for all $\mathring{r}_c = \{0.9, 0.8, 0.72, 0.66, 0.56, 0.55, 0.53, 0.5\}$ in irregular LDPC codes.

We can now rewrite the optimization of Eqs (18) and (19), by substituting Eqs (22) and (21), as

$$\min \quad R_s(\sigma_n^2) + R_c(\sigma_n^2) \tag{23}$$

$$\text{s.t.} \quad \sum_{i=1}^{d_v} \lambda_i \cdot J\left[\sqrt{(i-1)(J^{-1}(I_{\text{CND},k}^E))^2 + \frac{4}{\sigma_n^2}} \right] \geq 1 - J\left[\frac{J^{-1}(1 - I_{\text{CND},k}^E)}{\sum_{j=1}^{d_c} \rho_j \cdot \sqrt{j-1}} \right] + \varepsilon_k, \quad k \in [1, K] \tag{24}$$

Since the objective function (23) is a convex function of the noise variance, it can be solved efficiently by using optimization tools such as interior-point methods. Since it is difficult to compare the two continuous functions I_{VND}^E and I_{CND}^A in Eq. (19), we can use a simpler condition through the discrete sampling operation [52] in Eq. (24). Furthermore, under the same bitrate budget, the above lookup table is used to compare the noise variance decrements due to the information-plane and parity-plane SI refinements.

Let us denote the estimate refined noise variance in Eq. (6) by $\breve{\sigma}_s^2$, and denote the noise variance in Eq. (8) by $\breve{\sigma}_c^2$. If $\breve{\sigma}_s^2 < \breve{\sigma}_c^2$, we select the information plane bits send to the decoder; otherwise, we select parity plane bits, until the noise threshold σ_{th}^2 is reached.

Algorithm 1: Rate minimization

1: **Off-line EXIT analysis:**
   Compute the EXIT functions of RC-LDPC codes.
   For regular codes: follow Theorem 1 and Corollary 1.2.
   For irregular codes: follow Theorem 2 and Corollary 2.2.
2: Determine the noise threshold σ_{th}^2 with Eq. (17).
3: **Input:**
   The first refined SI $\mathbf{Y}_{(1)}^N$.
4: Estimate the current correlation noise variance $\sigma_{n,0}^2$ with Eq. (4) through a pilot-assisted correlation channel estimation algorithm.
5: Solve the optimization problem in Eqs (18) and (19) to get the optimal information plane rate R_s and parity plane rate R_c.
6: if $\breve{\sigma}_s^2 < \breve{\sigma}_c^2$, then
      Send R_s information plane bits from encoder.
      Updating the SI with Definition 1.
   else
      Send R_c parity plane bits from encoder.
      Updating the SI with Definition 2.
   end if
7: LDPC decoding. Get the refined SI $\mathbf{Y}_{(2)}^N$.
   if ${\sigma_n}^2 \leq \sigma_{th}^2$, then
      Decoding successfully, stop.
   else
      Go to step (4).
   end if
8: **Output:**
   The reconstructed binary sequence $\hat{\mathbf{X}}^N$.

Finally, the proposed rate optimization algorithms are embedded in the WZ video decoder, as shown in Fig.3. We briefly summarize in Algorithm 1, the minimization of the WZ video coding bitrate through EXIT chart-based SI refinement.

VI. Experimental results

In this section, we present the experimental results that illustrate the various features of our novel EXIT-chart based SI refinement algorithm. The experiments include three parts. First, we present coding performance comparisons with most relevant DVC coding schemes, which is the key performance in WZ video coding structures. Second, since the major performance increase of this work is to lower the coding rate, we give the rate analysis of the rate components. Third, we present the computational complexity analysis of the proposed method.

A. Experiment Setup

The test sequences include *foreman*, *hall*, *coastguard*, and *soccer*, in QCIF resolution at 15 fps. The RD performance results are given for different Group of Pictures (GOP) sizes, respectively when GOP=2, 4, 8. We implement a simple WZ video coding architecture according to the scheme in Fig.3. The key (I) frame uses H.264/AVC intra coding (JM18.3) [53]. The intra coding process follows the main profile properties, with intra prediction and rate-distortion-optimization switched on, and use CABAC entropy coding. The quantization parameters (QP) for each RD point are given in Table I.

TABLE I
Quantization parameters

	QP							
foreman	25	29	32	34	36	38	39	40
hall	24	29	31	33	35	36	37	39
coastguard	26	30	31	33	34	36	37	38
soccer	25	31	34	36	38	41	43	44

The WZ frame uses a RC-LDPC encoder to generate parity bits. We also consider using the RC-LDPC codes with regular and irregular LDPC structure cases. We use the EXIT chart analysis and the algorithm of Section V for adapting the rate in the WZ frame. It is worthy to mention that we do not use any motion estimation module in the video decoder, instead, the initial SI $\mathbf{Y}_{(0)}^{N}$ is simple copy of the key frame. Other SI construction algorithms such as motion-compensated interpolation/extrapolation [7], can also work together with our system and further improve the accuracy of $\mathbf{Y}_{(0)}^{N}$. Besides, the rate control algorithms with feedback channel-based WZ coding structure [2] can also work together with our novel method to further reduce the coding rate.

We investigate the compression performance in the following codecs:

(i) *H.264/AVC Intra codec*: This is a widely used Intra coding solution, which has been listed as the basic comparison for most DVC solutions. The results have been produced by JM18.3 [53].

(ii) *H.264/AVC no motion codec*: The video sequences are encoded in H.264/AVC codec when no motion vector component prediction is used in encoder. This mode is also employed as one of comparison algorithms such as in [6] [31] [54]. Herein, we refer the results presented in [54].

(iii) *DISCOVER WZ video codec* [54]: This codec is designed by the DISCOVER EU-project [40], which is a baseline codec used for benchmarking for most DVC codecs.

(iv) *IST-TDWZ with SIR video codec* [6]: A typical side information refinement (SIR) algorithm for a transform domain WZ video codec. This work shows better performance than the transform domain WZ video codec without any refinement case.

(v) *SID-CC hash-based WZ video codec* [35]: A latest hash-based WZ video coding framework. This work proposes a SID model to improve the WZ coding performance. It generates high quality SI at the decoder through hash-based motion estimation technique, and then enables bit plane successive refinement through the correlation channel (CC) estimation. This work shows good performance compared with the other revelent hash-based solutions in [30] [55] [56] [31].

(vi) *Proposed EXIT-based WZ video codec*: The proposed WZ video coding in Fig.3, where the initial SI is a simple copy of the key frame (H.264/AVC Intra frame). Then, the coding process follows the previous sections.

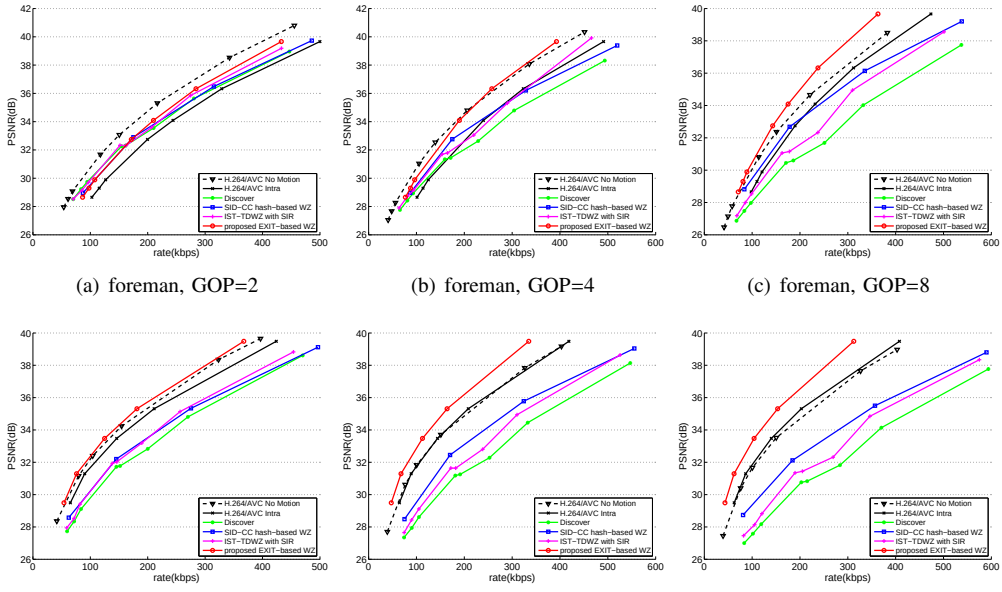
B. Compression Performance Experiments

The objective of our first experiment is to assess the rate-distortion (RD) performance of the different DVC algorithms. The proposed EXIT-chart based WZ video codec employs the regular rate compatible LDPC structure, where $d_v = 3$ and $d_c = 6$ with a rate of 0.5 and a block length of 2048. Fig. 7 shows the RD performance results for *Foreman* and *Soccer* sequences under different GoP sizes 2, 4, and 8, respectively. Fig. 8 respectively shows the RD performance results for *Coastguard* and *Hall* sequences.

The experimental results show that the exploitation of the intrinsic property of channel coding in DVC bring an effective contribution in compression performance improvement. In particular, we see in Fig.7 that for *Foreman* and *Soccer* sequences with high motions, the proposed EXIT-chart based WZ video codec in red line provides better performance than the other five benchmarks encoders in most cases. The proposed EXIT-based WZ video codec presents better performance in larger GOP size, which is reflected in Fig.7 and Fig.8. Only for the *Hall* sequence with small motion, the proposed codec is worse than the conventional motion-based WZ video coding schemes. For low coding rates in GoP=2, the proposed codec is slightly less than the other DVC encoders. And we should keep in mind that the side information does not use any motion compensated in our scheme.

As shown in Fig.7 and Fig.8, the results show six conclusions:

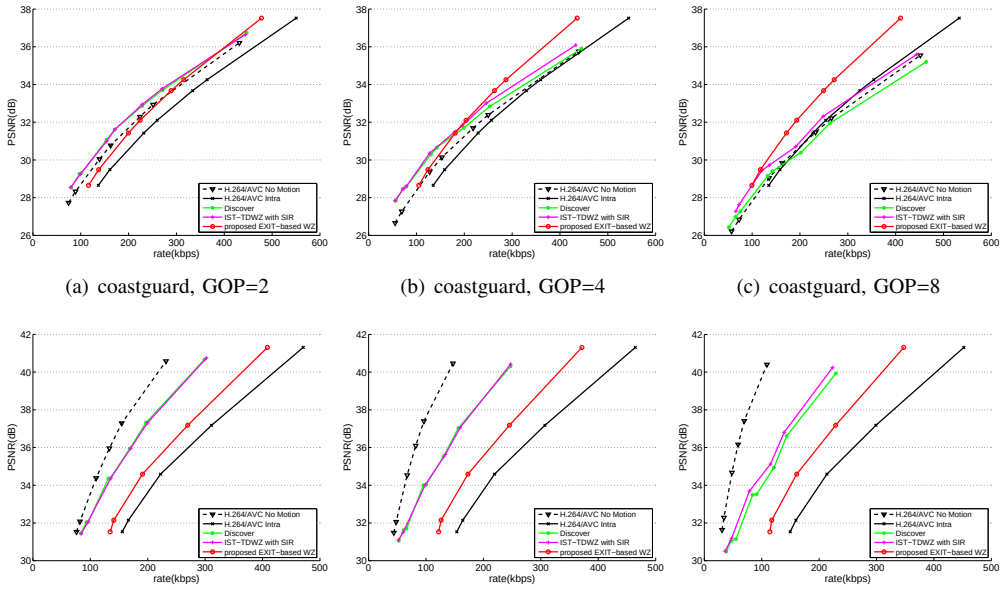
① In the complex motion scenario, the proposed method is more effective. Most conventional SI refinement methods are based on an assumption that the motion is smooth and flat. For the video sequences with high motion, the accuracy of SI drops down quickly, which results in the poor coding performance (see the results in sequence *soccer* in Fig. 7).



(a) foreman, GOP=2 (b) foreman, GOP=4 (c) foreman, GOP=8

(d) soccer, GOP=2 (e) soccer, GOP=4 (f) soccer, GOP=8

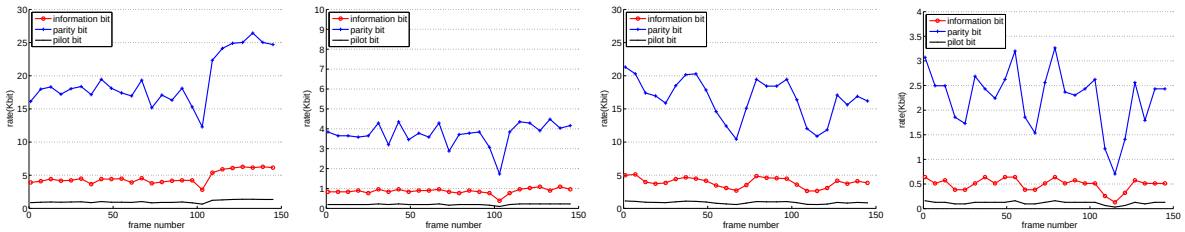
Fig. 7. Compression performance of the proposed and comparison algorithms for the *Foreman* (top) and *Soccer* (bottom) sequences (QCIF, 15Hz): (a) and (d) GOP=2; (b) and (e) GOP=4; (c) and (f) GOP=8.



(a) coastguard, GOP=2 (b) coastguard, GOP=4 (c) coastguard, GOP=8

(d) hall, GOP=2 (e) hall, GOP=4 (f) hall, GOP=8

Fig. 8. Compression performance of the proposed and comparison algorithms for the *Costguard* (top) and *Hall* (bottom) sequences (QCIF, 15Hz): (a) and (d) GOP=2; (b) and (e) GOP=4; (c) and (f) GOP=8.



(a) foreman (QP=25) (b) foreman (QP=40) (c) soccer (QP=25) (d) soccer (QP=41)

Fig. 9. The rates of information-plane, parity-plane and pilot bits.

Motion-based SI refinement shows poor performance in this case because it is very difficult to estimate the scene motion. The results of proposed EXIT-chart based WZ video codec still show good performance. This result reflects the robustness of the proposed method to high motion sequences.

② In case of moderate and high coding rates, the proposed method shows better performance. In Fig.7 and Fig.8, for the sequences *foreman*, *coastguard* and *soccer*, the results of proposed method are better in a large bitrate range. And the advantage is broadened when the bitrate increases. The reason is that the performance of DVC has close relation of the error correction capability of LDPC coding. Improving the error correction performance of LDPC decoder is an efficient method.

③ In the small objective motion scenario, the original SI refinement methods based motion estimation and compensation are more effective. We observe that for the sequence *hall*, shown in Fig.8(d)(e)(f), the results of (iii)(iv) video codecs are better; the reason is that the *hall* sequence has small objective motion with moderate contrast and noise and in a uniform background. It is easier to extract the motion information. Thus, conventional motion compensated interpolation/extrapolation and hash-based motion estimation methods provide good performance. However, for the proposed method, the WZ coded data of this sequence is mainly composed by parity-plane data. To adaptively adjust the information-plane and parity-plane data plays little role in this case.

④ The proposed method does not generate error propagation. We can observe the following phenomena. First, the PSNR values of the proposed WZ method (vi) and H.264 Intra (i) are in same level. The differences of these two algorithms are in the bitrates. That is in the same PSNR level, the bitrates of the two algorithms are different. The reason is that the proposed method focus on exploiting the intrinsic property of channel coding module. A successful channel encoder/decoder design should show better error correction capability. Thus, a principle of our framework is the successful decoding. The LDPC module will not lower the video quality under successful decoding condition. Consequently, the proposed method will not result in any error propagation, which is reflected in Fig.7 and 8.

⑤ In case of low coding rates in smooth motion scenario, motion-based method shows effective. We observe that for the sequence *coastguard*, shown in Fig.8(a), when QP is above 31 and GOP=2, the results of (iii)(iv) video codecs are better than the proposed method. The reasons are (1) the quantization level still affects the error correction capability of LDPC. And (2) the motions of *coastguard* sequence are somewhat smooth, in this case, motion-compensated based method shows advantage. The ①, ② and ④ show that in high-motion sequences or for moderate and high coding rates, the proposed method has better performance. And ③ shows that the motion-based methods show advantage in low coding rates or low-motion sequences. However, the error propagation of conventional methods will affect the compression efficiency for larger GOP size. The proposed method shows more effective in this case for *coastguard* sequence when GOP=8, which is reflected in Fig.8(c).

⑥ In case of large GOP size, the proposed method shows good performance. As shown in Fig.7 and Fig.8, the proposed method is efficient in overall R-D performance under different GOP sizes. In most conventional WZ coding methods, the R-D efficiency decreases with long GOP intervals because the accuracy of SI drops progressively. Fig.7 and Fig.8 show that the proposed WZ video codec keeps steady performance in long GOP intervals. One reason is that the correction ability is better estimated with the EXIT chart analysis. The second reason is that the proposed method does not generate error propagation to the following frames in the same GOP. The third reason is that the encoder sends information-plane bits together with parity-plane bits, which will compensate the correlation decay in the SI for larger GOP sizes.

C. The rates analysis of information-plane, parity-plane and pilot bits

As mentioned above, WZ frame includes both information-plane and parity-plane bits. Besides, we use pilot-assisted algorithm to make the decoder estimate current correlation noise. The final WZ frame consists of information-plane, parity-plane and pilot bits. In this experiment, we give how the WZ frame rate is distributed among these three parts. We implement half rate mother LDPC codes with a regular structure at block lengths of 2048. The degree pair is $d_v = 3$ and $d_c = 6$. The pilot symbol sequence is in $\eta = 32$ bits for a LDPC block. We respective test two cases, *foreman* sequence in QP=25, 40; and *soccer* sequence in QP=25, 41. All the two sequences are in QCIF format, 15fps and GOP=2. As shown in Fig. 9, the blue line and the red line are the parity-plane bitrate and information-plane bitrate of each frame. The final WZ frame is composed by about 78% parity-plane, 19% information-plane, and 3% pilot bits[1]. The proportion of pilot bits is relative fixed. We can observe two phenomena. First, the proportion of parity-plane decreases when the original video frames with high motion, which is shown in the inflection points. The reason is that the correlation between adjacent frames decrease for the high motion sequence. Information-plane bits show more effective in this case. Second, the quantization level affects the distribution of information-plane and parity-plane bits. The shapes of the information-plane and parity-plane curves change with the QP. The reason is that the correlation noise variation between the initial SI and WZ frame increases in larger QP. Accordingly, the constitution of information and parity planes changes with the correlation noise. When the correlation between the SI and the WZ is weak, the information-plane SI improves the performance alternatively, when the correlation is strong, the parity-plane SI provides better performance.

D. Computational complexity analysis

This work is implemented in Matlab. The core is to add the EXIT chart based SI refinement and rate estimate method. Thus, we analyze the computational complexity from the view of algorithm. To reduce the prohibitive computational

[1]The length of feedback channel is about 100-110 bits per frame in irregular LDPC codes, and is about 400-440 bits in regular LDPC codes in the experiments.

complexity of the WZ video decoding, hierarchical lookup tables are used in two stages. First, we employ off-line EXIT functions computation scheme, and store the noise thresholds in the EXIT-lookup-table, which is off-line computed through the mutual information variable functions. In the WZ video decoding process, the complex and time consuming calculation of the EXIT-chart analysis is thus replaced by a table lookup based on the proposed model[2]. Second, we also build the performance-lookup-table for RC-LDPC codes. This table stores two relations, one is between the information-plane rate and correlation noise variance, the other is between the parity-plane rate and correlation noise variance. This table provides the basic data to the rate optimization module. The overall complexity of the proposed algorithm is composed of three main parts.

First, the initial correlation noise has to be computed. Let $C_{f(\sigma_{n,0}^2)}$ be the complexity of the function $f(\sigma_{n,0}^2)$, which is implemented by a pilot algorithm. According to Eq.(4), $C_{f(\sigma_{n,0}^2)} = O(K_0^2)$ where K_0 is the length of pilot symbol sequence.

Second, the noise thresholds are determined under different coding rates from the off-line EXIT-lookup-table. Let $C_{f(\sigma_{th}^2)}$ be the complexity cost of the function $f(\sigma_{th}^2)$. From the lookup table operation, $C_{f(\sigma_{th}^2)} = O(\log K_1)$, where K_1 is the size of table. According to the choice of τ in Eq. (21), $K_1 \approx 1/\tau$.

Third, the optimization module of (23) determines the rates. Let $C_{f_{\text{opt}}(R_s,R_c)}$ be the complexity of optimization operation $f_{\text{opt}}(R_s, R_c)$. Since in each SI refinement, the encoder can either send P information-plane bits or parity-plane bits, the key point of the optimization is that the decoder can determine which plane bits can make the quick gradient descent of correlation noise variation through query the performance-lookup-table. Thus, the complexity concentrates in two lookup table operations. One is for the information plane, the other is for the parity plane. We use a bisection method. Assuming that the complexity of each lookup table is $O(\log K_2 \cdot \log K_3)$, where K_2 is the number of correlation noise values and K_3 is the number of bitrates. Then, $C_{f_{\text{opt}}(R_s,R_c)} = 2 \cdot O(\log K_2 \cdot \log K_3)$.

Overall, the computational complexity of proposed method can be approximated as

$$\begin{aligned} C_{\text{ProWZ}} &= C_{f(\sigma_{n,0}^2)} + N_{R_{\text{SI}}}(C_{f(\sigma_{th}^2)} + C_{f_{\text{opt}}(R_s,R_c)}) \\ &= O(K_0^2) + N_{R_{\text{SI}}}(O(\log K_1) + 2 \cdot O(\log K_2 \cdot \log K_3)) \end{aligned} \tag{25}$$

where $N_{R_{\text{SI}}}$ represents the number of SI refinement iterations. Comparing with the key (Intra) frame coding, the proposed WZ fame coding corresponds to adding two major modules. The first one is the proposed EXIT-based SI refinement process, in which the computational complexity is estimated by Eq.(25). The second one is LDPC codec, where the computational complexities are roughly $O(N^2)$ at encoder and linear at decoder in terms of code length N. In conclusion, the additional complexities are roughly $O(N^2)$ at encoder and $C_{\text{ProWZ}} + O(N)$ at decoder comparing with the key frame codec.

[2]This off-line computation is independent with the WZ coding process because it can be determined for a given RC-LDPC codec.

VII. Conclusion and Future Work

In this paper, we propose a framework for SI refinement and rate minimization in WZ video coding. We use an EXIT chart analysis to deduce the mutual information variations in LDPC decoding with SI refinement. In the case of general and complex motion video, the proposed method provides significant benefits over previous solutions from the literatures, notably lower coding rate and, at the same time, a lower decoding complexity. These advantages have been analyzed through bitrate deductions, EXIT charts and complexity analysis, and finally corroborated by simulation results. The proposed refinement solution can be extended to other SI refinement methods and other DVC algorithms. Finally, it would be necessary to take into account the basic LDPC structure in the analysis. Extension to LDPC codes design, as used in WZ video coding, would be interesting. In the future, we will focus on deploying this method in other video codecs and exploit more channel coding structures to video compression.

Appendix A
Proof of Theorem 1: the mutual information function of inner VND in punctured regular LDPC codes

A variable node with d_v degree, has $d_v + 1$ incoming messages. The variable node decodes by computing

$$L_{i,out} = L_{ch} + \sum_{j \neq i}^{d_v} L_{j,in}$$

where $L_{j,in}$ is the jth LLR value going into the variable node, $L_{i,out}$ is the ith extrinsic LLR value coming out of the variable node, L_{ch} is the channel LLR value, and $i = 1, ..., d_v$ indexes the degree.

Then, from Eq. (10), for the unpunctured variable node with degree d_v, the mutual information function of inner VND I_{VND}^E is

$$I_{\text{VND},C}^E(I_{\text{VND}}^A, d_v, \sigma_n^2, r_p) = J\left(\sqrt{(d_v - 1)[J^{-1}(I_{\text{VND}}^A)]^2 + \sigma_n^2}\right) \tag{26}$$

When a variable node is punctured, the channel LLR value $L_{ch} = 0$. The LLR value going into the punctured variable node, is

$$L_{i,out} = \sum_{j \neq i}^{d_v} L_{j,in}$$

Then, for the punctured variable node, the I_{VND}^E function is

$$I_{\text{VND},P}^E(I_{\text{VND}}^A, d_v, \sigma_n^2, r_p) = J\left(\sqrt{(d_v - 1)[J^{-1}(I_{\text{VND}}^A)]^2}\right) \tag{27}$$

By combining Eqs.(26) and (27), the average mutual information I_{VND}^E of the output of VND becomes

$$\begin{aligned} &I_{\text{VND}}^E(I_{\text{VND}}^A, d_v, \sigma_n^2, r_p) \\ &= \alpha \cdot I_{\text{VND},P}^E(I_A, d_v, \sigma_n^2, r_p) + \beta \cdot I_{\text{VND},C}^E(I_{\text{VND}}^A, d_v, \sigma_n^2, r_p) \end{aligned} \tag{28}$$

When the punctured rate is r_p, there is $\alpha = 1 - \frac{r}{r_p}$ and $\beta = \frac{r}{r_p}$. This completes the proof. □

Appendix B
Proof of Theorem 2: the mutual information function of inner VND in punctured irregular LDPC codes

For the irregular LDPC codes with design rate r, there is

$$r = 1 - \frac{\sum_{i=1}^{d_c} \rho_i / i}{\sum_{i=1}^{d_v} \lambda_i / i}$$

In a rate-compatible fashion, higher rate r_p is obtained by some puncturing patterns, indexed by puncturing distribution $\delta(x)$. After the puncturing operation, the fraction of edges incident to variable nodes of degree j is changed to $\lambda_j^{'}$. Under given puncturing distribution $\delta(x)$, there is $\lambda_j^{'} = \delta_j \lambda_j$. Let the updating degree distribution be $(\lambda(x)^{'}, \rho(x))$, where $\{\lambda_1^{'}, \lambda_2^{'}, ..., \lambda_{d_v}^{'}\}$. The punctured rate is computed as

$$r_p = \frac{r}{1-p}, \quad \text{where} \quad p = \frac{\sum_{j=1}^{d_v} \delta_j \lambda_j / j}{\sum_{j=1}^{d_v} \lambda_j / j}$$

A variable node with degree d_v, has $d_v + 1$ incoming messages. The variable node decodes by computing $L_{i,out} = L_{ch} + \sum_{j \neq i}^{d_v} L_{j,in}$. With degree distributions (λ, ρ), the average mutual information I_{VND}^E of the variable node is

$$I_{\text{VND},C}^E(I_{\text{VND}}^A, \lambda, \sigma_n^2, r_p) = \sum_{i=1}^{d_v} \lambda \cdot J\left(\sqrt{(i-1)[J^{-1}(I_{\text{VND}}^A)]^2 + \sigma_n^2}\right) \quad (29)$$

Then, for the unpunctured variable node, the EXIT function is

$$I_{\text{VND},C}^E(I_{\text{VND}}^A, \lambda, \sigma_n^2, r_p) = \sum_{i=1}^{d_v} \lambda_i(1-\delta_i) \cdot J\left(\sqrt{(d_v-1)[J^{-1}(I_{\text{VND}}^A)]^2 + \sigma_n^2}\right) \quad (30)$$

And for the punctured variable node, since $L_{ch} = 0$, there is $L_{i,out} = \sum_{j \neq i}^{d_v} L_{j,in}$. The mutual information function is

$$I_{\text{VND},P}^E(I_{\text{VND}}^A, \lambda, \sigma_n^2, r_p) = \sum_{i=1}^{d_v} \lambda_i(\delta_i) \cdot J\left(\sqrt{(d_v-1)[J^{-1}(I_{\text{VND}}^A)]^2}\right) \quad (31)$$

By combining Eqs.(30) and (31), the average mutual information I_{VND}^E of the output of VND under irregular LDPC codes becomes

$$\begin{aligned} I_{\text{VND}}^E(I_{\text{VND}}^A, \lambda, \sigma_n^2, r_p) =& I_{\text{VND},C}^E(I_{\text{VND}}^A, \lambda, \sigma_n^2, r_p) + I_{\text{VND},P}^E(I_{\text{VND}}^A, \lambda, \sigma_n^2, r_p) \\ =& \sum_{i=1}^{d_v} \lambda_i(\delta_i) \cdot J\left(\sqrt{(i-1)[J^{-1}(I_{VND}^A)]^2}\right) \\ &+ \sum_{i=1}^{d_v} \lambda_i(1-\delta_i) \cdot J\left(\sqrt{(i-1)[J^{-1}(I_{VND}^A)]^2 + \sigma_n^2}\right) \end{aligned} \quad (32)$$

This completes the proof. □

Acknowledgment

This work is supported by the National Natural Science Foundation of China (61001194), and by the Beijing Natural Science Foundation (4122078).

References

[1] T. Sikora, "Trends and perspectives in image and video coding," *Proc. IEEE*, vol. 93, no. 1, pp. 6–17, Jan. 2005.

[2] B. Girod, A. M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proc. IEEE*, vol. 93, no. 1, pp. 71–83, Jan. 2005.

[3] R. Puri, A. Majumdar, and K. Ramchandran, "PRISM: A video coding paradigm with motion estimation at the decoder," *IEEE Trans. Image Process.*, vol. 16, no. 10, pp. 2436–2448, Oct. 2007.

[4] G. Esmaili and P. Cosman, "Low complexity spatio-temporal key frame encoding for Wyner-Ziv video coding," in *Proc. DCC*, Mar. 2009, pp. 382–390.

[5] A. Aaron and B. Girod, "Compression with side information using Turbo codes," in *Proc. DCC*, Mar. 2002, pp. 352–261.

[6] R. Martins, C. Brites, A. Ascenso, and F. Pereira, "Refining side information for improved transform domain Wyner-Ziv video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 6, pp. 1327–1341, Sept. 2009.

[7] D. Varodayan, D. Chen, M. Flierl, and B. Girod, "Wyner-Ziv coding of video with unsupervised motion vector learning," *EURASIP Signal Process.: Image Commun.*, vol. 23, no. 5, pp. 369–378, Jun. 2008.

[8] K. Kasai, T. Tsujimoto, R. Matsumoto, and K. Sakaniwa, "Rate-compatible Slepian-Wolf coding with short non-binary LDPC codes," in *Proc. DCC*, Mar. 2010, pp. 288–296.

[9] Q. Xu, V. Stanković, and Z. Xiong, "Distributed joint source-channel coding of video using Raptor codes," *IEEE J. Select. Areas Commun.*, vol. 25, no. 4, pp. 851–861, May. 2007.

[10] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. 19, no. 4, pp. 471–480, Jul. 1973.

[11] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inf. Theory*, vol. 22, no. 1, pp. 1–10, Jan. 1976.

[12] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *IEEE Trans. Commun.*, vol. 52, no. 4, pp. 670–678, Apr. 2004.

[13] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[14] C. Brites, J. Ascenso, and F. Pereira, "Studying temporal correlation noise modeling for pixel based Wyner-Ziv video coding," in *Proc. ICIP*, Oct. 2006, pp. 273–276.

[15] R. Puri and K. Ramchandran, "PRISM: A new robust video coding architecture based on distributed compression principles," in *Proc. Allerton*, Oct. 2002, pp. 586–595.

[16] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. ICC*, vol. 2, May. 1993, pp. 1064–1070.

[17] V. Stanković, A. Liveris, Z. Xiong, and C. Georghiades, "On code design for the Slepian-Wolf problem and lossless multiterminal networks," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1495–1507, Apr. 2006.

[18] D. Schonberg, K. Ramchandran, and S. Pradhan, "Distributed code constructions for the entire Slepian-Wolf rate region for arbitrarily correlated sources," in *Proc. DCC*, Mar. 2004, pp. 292–301.

[19] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes," *IEEE Commun. Lett.*, vol. 6, no. 10, pp. 440–442, Oct. 2002.

[20] T. Coleman, A. Lee, M. Médard, and M. Effros, "On some new approaches to practical Slepian-Wolf compression inspired by channel coding," in *Proc. DCC*, Mar. 2004, pp. 282–291.

[21] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive codes for distributed source coding," *J. Signal Process.*, vol. 86, no. 11, pp. 3123–3130, Apr. 2006.

[22] C. Hsu and A. Anastasopoulos, "Capacity achieving LDPC codes through puncturing," *IEEE Trans. Inf. Theory*, vol. 54, no. 10, pp. 4698–4706, Oct. 2008.

[23] F. Cen, "Design of punctured LDPC codes for rate-compatible asymmetric Slepian-Wolf coding," in *Proc. DCC*, Mar. 2009, p. 438.

[24] ——, "Design of degree distributions for LDPCA codes," *IEEE Commun. Lett.*, vol. 13, no. 7, pp. 525–527, Jul. 2009.

[25] X. Artigas and L. Torres, "Iterative generation of motion-compensated side information for distributed video coding," in *Proc. ICIP*, vol. 1, Sept. 2005, pp. 833–836.

[26] W. Liu, L. Dong, and W. Zeng, "Motion refinement based progressive side-information estimation for Wyner-Ziv video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 12, pp. 1863–1875, Dec. 2010.

[27] J. Ascenso, C. Brites, and F. Pereira, "Content adaptive Wyner-Ziv video coding driven by motion activity," in *Proc. ICIP*, Oct. 2006, pp. 605–608.

[28] Y. Zhang, D. Zhao, H. Liu, Y. Li, S. Ma, and W. Gao, "Side information generation with auto regressive model for low-delay distributed video coding," *J. Vis. Commun. Image R.*, vol. 23, pp. 229–236, 2012.

[29] T. N. Dinh, G. Lee, J.-Y. Chang, and H.-J. Cho, "Side information generation using extra information in distributed video coding," in *Proc. ISSPIT*, vol. 1, Dec. 2007, pp. 138–143.

[30] A. Aaron, S. Rane, and B. Girod, "Wyner-Ziv video coding with hash-based motion compensation at the receiver," in *Proc. ICIP*, vol. 5, Oct. 2004, pp. 3097–3100.

[31] J. Ascenso, C. Brites, and F. Pereira, "A flexible side information generation framework for distributed video coding," *Multimedia Tools Appl.*, vol. 48, no. 3, pp. 381–409, Jul. 2010.

[32] M. Tagliasacchi and S. Tubaro, "Hash-based motion modeling in wyner-ziv video coding," in *Proc. ICASSP.*, Apr. 2007, pp. 569–573.

[33] F. Verbist, N. Deligiannis, M. Jacobs, J. Barbarien, P. Schelkens, and A. Munteanu, "A statistical approach to create side information in distributed video coding," in *Proc. ICDSC.*, Aug. 2011, pp. 1–6.

[34] F. Verbist, N. Deligiannis, M. Jacobs, J. Barbarien, A. Munteanu, P. Schelkens, and J. Cornelis, "Maximum likelihood estimation for the generation of side information in distributed video coding," in *Proc. IWSSIP.*, Jun. 2011, pp. 1–4.

[35] N. Deligiannis, J. Barbarien, M. Jacobs, A. Munteanu, A. Skodras, and P. Schelkens, "Side-information-dependent correlation channel estimation in hash-based distributed video coding," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1934–1949, Apr. 2012.

[36] N. Deligiannis, F. Verbist, J. Slowack, R. Van de Walle, P. Schelkens, and A. Munteanu, "Joint successive correlation estimation and side information refinement in distributed video coding," in *Proc. EUSIPCO.*, Aug. 2012, pp. 569–573.

[37] N. Merhav and T. Weissman, "Coding for the feedback Gel'fand-Pinsker channel and the feedforward Wyner-Ziv source," *IEEE Trans. Inf. Theory*, vol. 52, no. 9, pp. 4207–4211, Sept. 2006.

[38] Y. Steinberg and N. Merhav, "On hierarchical joint source-channel coding with degraded side information," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 886–903, Mar. 2006.

[39] C. Brites, J. Ascenso, and F. Pereira, "Improving transform domain Wyner-Ziv video coding performance," in *Proc. ICASSP*, May. 2006, pp. II–525–II–528.

[40] X. Artigas, J. Artigas, M. Dalai, S. Klomp, D. Kubasov, and M. Quaret, "The DISCOVER codec: Architecture, techniques and evaluation," in *Proc. PCS*, Nov. 2007.

[41] C. Brites, J. Ascenso, J. Q. Pedro, and F. Pereira, "Evaluating a feedback channel based transform domain Wyner-Ziv video codec," *EURASIP Signal Process.: Image Commun.*, vol. 23, no. 4, pp. 269–297, Apr. 2008.

[42] C. Brites and F. Pereira, "An efficient encoder rate control solution for transform domain Wyner-Ziv video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 9, pp. 1278–1292, Sep. 2011.

[43] J. Martinez, G. Fernandez-Escribano, H. Kalva, W. Weerakkody, W. Fernando, and A. Garrido, "Feedback free DVC architecture using machine learning," in *Proc. ICIP*, Oct. 2008, pp. 1140–1143.

[44] W. Ji, P. Frossard, and Y. Chen, "EXIT chart-based side information refinement for Wyner-Ziv video coding," in *Proc. DCC*, Apr. 2012, pp. 209–218.

[45] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: Model and erasure channel properties," *IEEE Trans. Inf. Theory*, vol. 50, no. 11, pp. 2657–2673, Nov. 2004.

[46] C. Brites and F. Pereira, "Correlation noise modeling for efficient pixel and transform domain Wyner-Ziv video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 9, pp. 1177–1190, Sept. 2008.

[47] G. Esmaili and P. Cosman, "Wyner-Ziv video coding with classified correlation noise estimation and key frame coding mode selection," *IEEE Trans. Image Process.*, vol. 20, no. 9, pp. 2463–2474, Sept. 2011.

[48] X.-Y. Hu, E. Eleftheriou, D.-M. Arnold, and A. Dholakia, "Efficient implementations of the sum-product algorithm for decoding LDPC codes," in *Proc. GLOBECOM*, vol. 2, Nov. 2001, pp. 1036–1036E.

[49] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727–1737, Oct. 2001.

[50] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Inf. Theory*, vol. 42, no. 2, pp. 429–445, Mar. 1996.

[51] E. Sharon, A. Ashikhmin, and S. Litsyn, "Analysis of low-density parity-check codes based on EXIT functions," *IEEE Trans. Commun.*, vol. 54, no. 8, pp. 1407–1414, Aug. 2006.

[52] J. Karjalainen, M. Codreanu, A. Töli, M. Juntti, and T. Matsumoto, "EXIT chart-based power allocation for iterative frequency domain MIMO detector," *IEEE Trans. Signal Process.*, vol. 59, no. 4, pp. 1624–1641, Apr. 2011.

[53] H.264/AVC Software [Online]. Available: http://iphome.hhi.de/suehring/tml.

[54] DISCOVER-Distributed Coding for Video Services [Online]. Available: http://www.img.lx.it.pt/~discover.

[55] J. Ascenso and F. Pereira, "Adaptive hash-based side information exploitation for efficient wyner-Ziv video coding," in *Proc. ICIP*, vol. 3, Sept. 2007, pp. III–29–III–32.

[56] N. Deligiannis, A. Munteanu, T. Clerckx, J. Cornelis, and P. Schelkens, "Overlapped block motion estimation and probabilistic compensation with application in distributed video coding," *IEEE Sig. Lett.*, vol. 16, no. 9, pp. 743–746, Sept. 2009.

Wen Ji (M'09) received the M.S. and Ph.D. degrees in communication and information systems from Northwestern Polytechnical University, China, in 2003 and 2006, respectively.

She is an Associate Professor in the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China. She was a Post Doctoral Fellow from 2007 to 2009, and was an Assistant Professor from 2009 to 2010 at ICT, CAS. Her research areas include video communication and networking, video coding, channel coding, information theory, optimization, and pervasive computing.

Pascal Frossard (S'96-M'01-SM'04) received the M.S. and Ph.D. degrees in electrical engineering from the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, in 1997 and 2000, respectively.

He was a Research Staff Member with the IBM T. J. Watson Research Center, Yorktown Heights, NY, from 2001 to 2003, where he worked on media coding and streaming technologies. Since 2003, he has been a Faculty at EPFL, where he heads the Signal Processing Laboratory - LTS4. His current research interests include image representation and coding, visual information analysis, distributed image processing and communications, and media streaming systems.

Dr. Frossard was the General Chair of IEEE ICME in 2002 and Packet Video in 2007. He has been the Technical Program Chair of EUSIPCO in 2008, and a member of the organizing or technical program committees of numerous conferences. He has been an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA since 2004, the IEEE TRANSACTIONS ON IMAGE PROCESSING since 2010, and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from 2006 to 2011. He has been the Vice-Chair of the IEEE Image, Video and Multidimensional Signal Processing Technical Committee since 2007. He has been an Elected Member of the IEEE Visual Signal Processing and Communications Technical Committee since 2006 and the IEEE Multimedia Systems and Applications Technical Committee since 2005. He served as Vice-Chair of the IEEE Multimedia Communications Technical Committee from 2004 to 2006 and as a member of the IEEE Multimedia Signal Processing Technical Committee from 2004 to 2007. He was a recipient of the Swiss NSF Professorship Award in 2003, the IBM Faculty Award in 2005, the IBM Exploratory Stream Analytics Innovation Award in 2008, and the IEEE Transactions on Multimedia Best Paper Award in 2011.

Yiqiang Chen (M'03) received the B.A.Sc. and M.A. degrees from the University of Xiangtan, Xiangtan City, China, in 1996 and 1999, respectively, and the Ph.D. degree from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, in 2002.

In 2004, he was a Visiting Scholar Researcher at the Department of Computer Science, Hong Kong University of Science and Technology (HKUST), Hong Kong. Currently, he is a Professor and Director of the pervasive computing research center at ICT, CAS. His research interests include artificial intelligence, pervasive computing, and human computer interface.