NEW INSIGHTS INTO IMPROVING COMPRESSION EFFICIENCY FOR DISTRIBUTED VIDEO CODING

Guogang Hua[a], Chang Wen Chen[b]

[a]Qualcomm Inc, 5775 Morehouse Drive, San Diego, CA 92121
[b]Dept of CSE, University at Buffalo, The State University of New York, Buffalo, NY 14260
ghua@qualcomm.com, chencw@cse.buffalo.edu

ABSTRACT

In this paper, we investigate a fundamental issue in the distributed video coding (DVC) that, once resolved, would substantially improve the compression efficiency in DVC. This fundamental issue is the underlining relation between the distribution of the prediction errors and the compression efficiency of DVC. In the current approach to DVC, after the construction of prediction frame at decoder, the difference between the prediction and the current frames, or the prediction error, is inversely proportional to the correlation between these frames. Most existing approaches in DVC attempt to maximize such correlation, or to minimize the prediction error, in order to achieve the desired video compression efficiency. Recently, the research in DVC has reached a plateau in terms of its performance in coding efficiency. We believe one key solution to take DVC out of such performance plateau will be to design a better scheme to represent the correlation more effectively in DVC. In this research, we have worked out compelling theoretical analysis and proved that, in order to reduce the number of bits to be sent to decoder, the distribution of the prediction errors needs to be as concentrated as possible. From the practical point of view, we also show that the error control codes adopted for DVC will achieve higher efficiency when the distribution of prediction errors is more concentrated.

Index Terms**—** Distributed video coding, compression efficiency, distribution of prediction errors

1. INTRODUCTION

In traditional video compression, as standardized by ISO MPEG and ITU-T H.264, the encoder is much more complicated than the decoder. This class of codec architecture has been driven predominately by the broadcasting or "downlink" nature of the traditional video applications. With the advances in contemporary technologies, emerging applications demand low complexity encoder, in particular for mobile handheld devices. With more and more mobile handsets supporting multimedia capturing, playing, and communication capacities, the new media-rich "uplink" wireless video transmission applications require a total redesign of these traditional downlink friendly video architectures. The new architecture calls for low-power and low complexity video encoding at the mobile units. Motivated by these emerging applications, DVC schemes [1][2] based on distributed source coding theory have been developed in the last few years.

The traditional hybrid video codec (such as MPEG, H.26x) exploits the spatial and temporal redundancy existing in the video sequences at the encoder end. It is the searching of such spatial and temporal correlations that requires tremendous amount of computation. However, the correlation exploitation alone is not the only means for a codec to achieve the latest compression efficiency in hybrid video codec. Various strategies aiming at representing the de-correlated video efficiently, including the well-known zigzag scan, run length code, entropy code, and skipped macroblock have also been developed. This correlation representation plays a very important role in hybrid video codec. However, in current researches in DVC, there has not been adequate investigation on the efficient representation of corresponding correlations. We believe this is one of the main reasons that there still exists a substantial gap between current DVC and traditional hybrid video coding.

1.1. Existing distributed video coding schemes

Many distributed video coding schemes share a general architecture as shown in Figure 1. Typically, the encoder applies error control codes to each frame and generates syndrome bits. To achieve compression efficiency, only part of the syndrome bits (punctured from the original syndrome bits) is usually sent. The decoder uses the correlation between the video frames to construct an estimation of the current frame and such estimation can be viewed as a noisy version of the original frame. Then the error control decoder combines the received syndrome bits and the estimation of the current frame to decode the current frame.

From the process shown in Figure 1, it is straightforward to understand that the performance of distributed video coding depends on two factors: the first one is the accuracy of the estimation of current frame while the second one is the decoding bit error rate performance of the error control codes. Most current research on distributed video coding has been worked on these two directions in order to achieve good video coding performance. In [3], the author proposed DCT domain and hash code based distributed video coding method. Block wise DCT is applied to each block in Wyner-Ziv frame in order to exploit the spatial correlation. The transform coefficients are grouped together to form different coefficient bands, and each coefficient band is then encoded independently. A hash code is generated and sent to decoder to help motion search to find best matched block in reference frame. In [4], the authors used highly compressed version of each frame as reference to perform motion estimation at the decoder in order to build more accurate estimation. Although there is cost for compression and transmission of those low quality frames, the overall bit rate can be reduced because more syndrome bits will be saved because of accurate estimation. In [5], the authors proposed a rate-adaptive LDPC Accumulate (LDPCA) codes and Sum LDPC Accumulate (SLDPC) codes for distributed source coding. The authors claimed that those codes with code length of 6336 bits can achieve within 10% of the Slepian-Wolf theoretical bound.

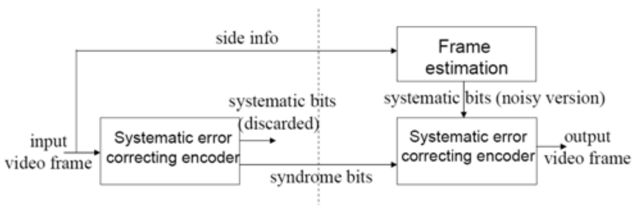

Figure 1: Illustration of a typical syndrome-based distributed video coding scheme

1.2. Overview of the this paper

From the above simple description of the distributed video coding schemes, it appears that after reference frame is reconstructed and error control codes are decided, there is no room for us to further improve the compression efficiency of the DVC. However, the roadmap of the traditional hybrid video codec over the last two decades indicates that significant improvement is still possible when a better representation of correlation can be developed. We believe the same principle can also be applied to DVC and seek to find the underlining relations between the distribution of the prediction errors and the compression efficiency of the DVC.

Some of our previous DVC schemes [6][7] have been developed along a similar line of consideration. We have designed a distributed video coding scheme based on zero motion block skip. In this research, the encoder first compares the current block with the co-located block in the reference frame. If the difference is sufficiently similar, this block is named a zero motion block. During encoding, the zero motion blocks are not coded by Wyner-Ziv encoder and there is no need to send syndrome bits to the decoder. The experimental results show that the scheme based on zero motion skip is able to improve the coding efficiency of DVC by about 1dB.

One natural question to ask will be: why the zero motion skip can substantially improve the compression efficiency of DVC? This investigation reported in this paper answers this question from both theoretical and practical points of view. The theoretical proof and practical demonstration presented in this paper point a new direction to further improve the coding efficiency of DVC.

First, we shall show in theory how the distribution of the prediction errors would affect the compression efficiency of the DVC. We convert the problem of investigating the distribution of prediction errors in DVC to a linear programming problem. By solving this linear programming problem, we prove that, in DVC, in order to improve the compression efficiency, the distribution of the prediction errors should be as concentrated as possible. Based on this result from theoretical analysis, it is easy to explain why zero motion skip can improve the coding efficiency of DVC. This is because there is no prediction error in zero motion blocks. By removing these zero motion blocks out of the encoding bitstream, we actually are able to make the prediction error more concentrated.

Second, from practical point view, we shall demonstrate that error control codes actually perform better when the errors are more concentrated. LDPC is used in this research to illustrate this underlining principle. By performing better, we mean that when the errors are more concentrated, the simulation results of the LDPC are closer to the theoretical boundaries.

Third, from error control codes point of view, our experimental results also show that the performance of distributed source coding can be improved with more concentrated prediction errors.

The rest of this paper is organized as following: Section 2 proves why the more concentrated the error distribution; the better the compression efficiency can be achieved. Section 3 gives some numerical demonstrations on how much the error distribution can affects the compression efficiency of DVC. In section 4 we demonstrate that, in practice, the LDPC expects more concentrated error distribution too. In section 5, we show from error control point of view, in DVC, the more concentrated prediction errors, the better performance can be achieved. Section 6 concludes this paper with some discussions.

2. THEORETICAL ANALYSIS

First of all, let the reference frame be Y and the current encoding frame be X. After motion search and reference construction at the decoder, the difference between X and Y is determined. Let us assume there are total E different bits between X and Y. The probability of bit difference between X and Y, p is also determined. From information theory principle, the conditional entropy $H(X|Y)$ is defined as:

$$H(X|Y) = -p \times \log_2(p) - (1-p) \times \log_2(1-p) \quad (1)$$

where $H(X|Y)$ is often interpreted as the "uncertainty" remaining in X given the observation of Y. In coding theory, $H(X|Y)$ is the average number of bits that need to be sent to the decoder to remove the "uncertainty" (to correct the prediction errors).

In distributed video coding, the length of the error control codes used (Turbo or LDPC) is much less than the number of bits for one frame video. Therefore, the data for one video frame is often divided into multiple blocks, and we can assume that there are N blocks. We assume that each block has length of L bits, which is also the length of the error control codes. For each block, the average bits needed to correct the prediction errors is defined by the conditional entropy $H(X|Y)$ in equation (1), so in theory the number of bits that need to be sent in order to correct the difference in each block can be defined as:

$$N_s = LH(X/Y) = -L(p\log_2(p) + (1-p)\log_2(1-p)) \quad (2)$$

Now, let us show how the distribution of the prediction errors would affect the compression efficiency of distributed video coding. Suppose there are e_i errors in the i-th block. In theory, the number of bits needing to be sent to correct the e_i errors will be:

$$N_{si} = -L\left(\frac{e_i}{L}\log_2\left(\frac{e_i}{L}\right) + \left(1-\frac{e_i}{L}\right)\log_2\left(1-\frac{e_i}{L}\right)\right) \quad (3)$$

The total number of bits needing to be sent to correct errors in one frame can be written as a function of errors:

$$f(e) = -L\sum_{i=1}^{N}\left(\frac{e_i}{L}\log_2\left(\frac{e_i}{L}\right) + \left(1-\frac{e_i}{L}\right)\log_2\left(1-\frac{e_i}{L}\right)\right) \quad (4)$$

where $e = [e_1, e_2, \cdots e_N]$.

In video compression, the ultimate goal is to minimize the total number of bits that need to be sent to the decoder. In this case, the problem now becomes minimizing function $f(e)$, subject to:

$$0 \le e_i \le L \qquad \sum_{i=1}^{N} e_i = E \quad (5)$$

This is a typical constrained optimization problem in linear programming which can be solved by the Lagrange multipliers method. The optimization is equivalent to minimizing/maximizing the following objective function $J(e,\lambda)$:

$$\begin{aligned} J(e,\lambda) &= f(e) + \lambda\left(E - \sum_{i=1}^{N} e_i\right) \\ &= -L\sum_{i=1}^{N}\left(\left(\frac{e_i}{L}\right)\log_2\left(\frac{e_i}{L}\right) + \left(1-\frac{e_i}{L}\right)\log_2\left(1-\frac{e_i}{L}\right)\right) \\ &+ \lambda\left(E - \sum_{i=1}^{N} e_i\right) \end{aligned} \quad (6)$$

The gradients of the function can be written:

$$\begin{aligned} \frac{\partial J(e,\lambda)}{\partial e} &= \log_2\left(\frac{L-e_i}{e_i}\right) - \lambda = 0 \qquad i = 1,2,\cdots N \\ \frac{\partial J(e,\lambda)}{\partial \lambda} &= E - \sum_{i=1}^{N} e_i = 0 \end{aligned} \quad (7)$$

These are $N+1$ equations. From these $N+1$ equations, it is easy to find that $f(e)$ achieves the maximum value when:

$$e_1 = e_2 = \cdots = e_N = \frac{E}{N} \quad (8)$$

Notice this is the worst case in data compression and is certainly not what we seek in our research.

Let us define $K = \left\lfloor \frac{E}{L} \right\rfloor$, the maximal integer less than $\frac{E}{L}$. It is easy to find that the function $f(e)$ achieves the minimal value when

$$\begin{aligned} e_k &= L \qquad k = 1,2,\cdots K \\ e_{K+1} &= E - L \times K \\ e_k &= 0 \qquad k = K+1, K+2, \cdots N \end{aligned} \quad (9)$$

It is true that, in practice, it is usually impossible to make the error probability of one block to be 1. However, it is possible to make it to be 0 which represents error free channels. The equations (8) and (9) offer the researchers in DVC effective guidelines to reduce the value of function $f(e)$, or, the number of bits that need to be sent in distributed video coding. In other words, we need to design a strategy that would make the errors concentrated in as few blocks as possible.

3. NUMERICAL EXAMPLES

The previous section offers the theoretical analysis of how the distribution of the prediction errors affects the compression efficiency of DVC. There is no quantitative analysis as to how much will the error distribution affect the compression efficiency. In this section, we provide some numerical examples to show how and how much the distribution of the errors affects the compression efficiency of distributed video coding. Suppose the errors are evenly distributed in n blocks ($n \le N$). The probability of bits difference in each block can be defined as:

$$p = \frac{E}{nL} \quad (10)$$

The total number of bits needed for one frame can be written as:

$$\begin{aligned} N_T &= nLH(p) \\ &= nL(-\frac{E}{nL}\log_2(\frac{E}{nL}) - (1-\frac{E}{nL})\log_2(1-\frac{E}{nL})) \end{aligned} \quad (11)$$

In (11), if we fix E and L (which is usually the case in really application after reference construction at the decoder and selected error control codec), equation (11) is a function of $n : N_T(n)$. The derivatives of (11) are:

$$\frac{dN_T(n)}{dn} = -L\log_2(1-\frac{E}{nL}) \quad (12)$$

We know that $0 < \frac{E}{nL} < 1$, so it is easy to show that

$$\frac{dN_T(n)}{dn} > 0 \quad (13)$$

This proves that function in (11) is monotonically increasing with respect to n when E and L are fixed.

To show visually the relationship between the compression efficiency and error distribution, we provide here a quantitative example. Assume that in a QCIF video frame, the prediction errors for the frame is 3000 bits. We also assume that the block length is the length of LDPC codes as in [5], 6336 bits. Figure 2 gives the number of bits that need to be sent to the decoder to correct the errors in theory with the number of blocks in which the errors evenly distributed. From this figure, we can see that, in theory, the number of bits needs to be sent increases quickly when the number of blocks of error distributed increases.

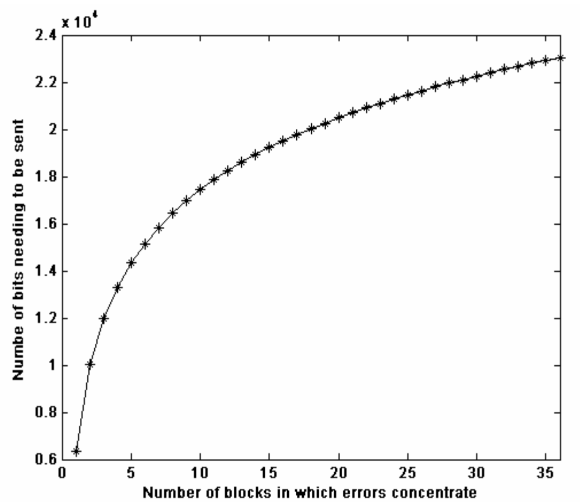

Figure 2: Number of bits needing to be sent with the number of blocks in which prediction errors distributed.

Figure 3 shows the number of bits that need to be sent in theory with the number of error bits in one QCIF video frame. From this figure, it is easy to see that, in theory, the number of bits needing to be sent increases with number of total errors; but the speed of increase is much faster when the number of blocks in which errors were distributed is larger. That is to say, when the number of blocks in which the prediction errors distribute is large, the same amount of increase in prediction errors will result in a much more increase in the number of bits of error correction codes in order to correct the errors. This will in turn result in the decrease in compression efficiency of the DVC schemes.

From the numerical examples shown in these two figures, we can see that the distribution of the prediction errors indeed has a significant impact on the compression efficiency of DVC. In the next two sections, we shall show two practical implementations that further demonstrate the relationship between the distribution of prediction errors and the compression efficiency of the DVC.

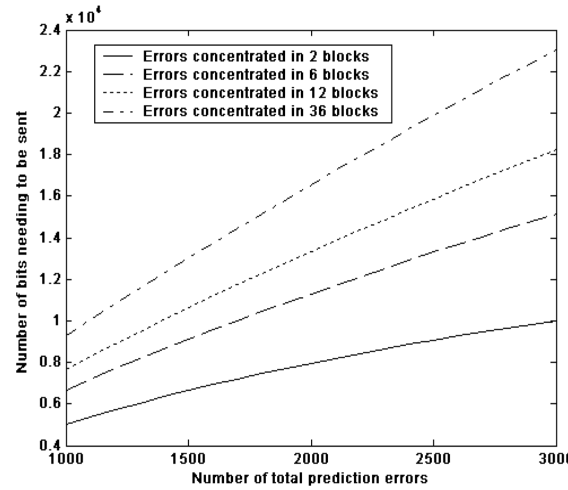

Figure 3: Number of bits needing to be sent in theory with the number of error bits.

4. PERFORMANCE OF LDPC CODES WITH DIFFERENT ERROR CONCENTRATIONS

In the previous section, we show that, in theory, the compression efficiency is proportional to the degree of concentration for the prediction errors. The conclusion is obtained based on theoretical analysis and illustrated with numerical examples. All the values in Figures 2 and Figure 3 are the theoretical bounds, which may not be achievable by any practical Slepian-Wolf coder. In [5], the authors proposed an adaptive Slepian-Wolf coder which can achieve up to 5-10% within the theoretical bounds.

In this section, we report the simulation results on how the distribution of prediction errors would affect the practical coding efficiency of DVC by using the LDPC codes proposed in [5]. The program for LDPC codes used in this study is downloaded from [8].

We simulate different total number of error bits distributed in different blocks. The total number of error bits simulated in this study includes 1000, 1500, 2000, 2500, and 3000, respectively. For each of the total number of error bits, we simulate those errors distributed in 2, 4, 6 and 12 blocks. For each of the total number of error bits and number of block in which the errors concentration combination, an error probability p can be calculated. For each of these combinations, a random series with length of the series being 6336 is generated as reference Y; and another series X is generated from Y by randomly changing some bits in Y with probability p. The pair of X and Y series are input to the program in [8] to test the number of bits needed to correct the difference between X and Y. For each error/block number combination, 3600 pairs of binary series are simulated. The average numbers needed to be sent for each frame are shown in Figure 4. In this figure, each type of line without marker is the theoretical boundary from equation (11); while the same type of line with maker is the simulation results.

From this figure, we can see clearly that, when the number of blocks in which errors concentration is smaller, the simulation results are closer to the theoretical bounds. This observation can be explained as follows: for the codes that have been used in [8], the LDPC is punctured to rates $2/66, 3/66, \cdots 66/66$. Let us assume for a low error probability (errors are not so concentrated), in theory, the rate to correct the errors is $3.1/66$. However, in practice, the best rate that can be achieved will be $4/66$. The difference is about 30%. For a high error probability (errors are more concentrated), the rate to correct the errors will be $25.1/66$. However, in practice, the best rate that can be achieved will be $26/66$. The difference has now become 3.6%, which is much less than the 30% in the case of less concentrated prediction errors.

From these practical simulations, we can reach the same conclusion that is consistent with the theoretical analysis – to achieve better compression ratio, the prediction errors in DVC should be made concentrated in as few blocks as possible.

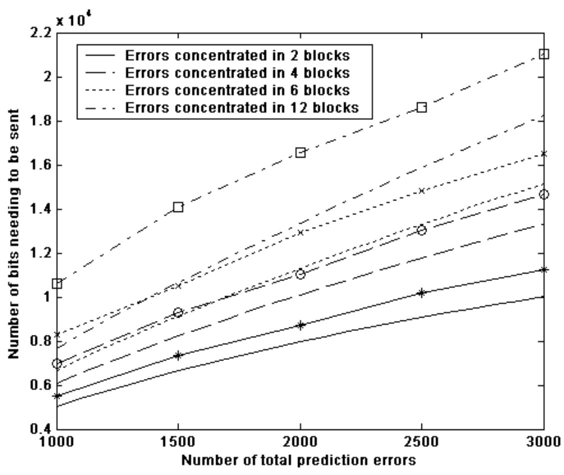

Figure 4: Average number of bits needing to be sent in LDPC simulations.

5. FROM ERROR CONTROL POINT OF VIEW

As is well-known, in distributed source coding, error control codes are used to construct the practical encoding and decoding. This is based on the intrinsic analogies between channel coding and source coding. As a result, design principles used in error control codes can also be applied to the design of efficient distributed video coding.

In the application of error control codes, unequal error protection (UEP) is usually adopted when the transmitter is given the estimate of the channel state [9]. This is because the transmission channels are dynamic with an ever changing channel error rates. For a given time instant when the noise level is low, very few FEC bits need to be assigned to protect the information transmission. As matter of fact, when the channel is noise free at a given time instant, no FEC is needed at all. However, when the channel noise level is high, more FEC bits are needed to protect the information from the noise corruption.

In distributed video coding, the channel "errors" come from the difference between the block in the current frame and the corresponding block in the reference frame. In a video sequence, there are two different types of blocks in adjacent frames: some blocks are considered unchanged within the acceptable distortion range. They may include the blocks in the background with negligible change in intensity. Other blocks may experience significant changes. They may include the blocks in the high motion areas. Therefore, different blocks will have different probabilities in terms of changes they experience over consecutive frames. Such observation motivates the DVC researchers to differentiate the blocks and DCT coefficients. Applying the idea of UEP, we can treat the different blocks and coefficients differently in order to minimize the total rate of error control rates for DVC so as to achieve improved compression efficiency.

In our previous studies [6][7], we have developed a zero motion skip scheme for DVC. In this approach, the encoder first compares the current block with the co-located block in the reference frame. When they are sufficiently similar, this block is considered zero motion block. The basic idea behind the zero motion skip distributed video coding is that in the encoding of the video frames, zero motion blocks can be identified and made known to the decoder to achieve an improved compression efficiency.

The benefits of identifying zero motion vector blocks are twofold to improve both aspects of rate-distortion behavior critical to the performance of distributed video coding as we discussed earlier. First, it improves the accuracy of frame estimation in the decoder. Since some blocks remain unchanged, when the decoder estimates the current frame, it can just use the corresponding blocks in the reference frame to substitute for the current block to produce correct estimation. This is equivalent to the error free channels and no FEC is necessary. Second, the skipped zero motion vector blocks will also help to improve the decoding bit error rate performance of the Turbo decoder. In the next section, we will show that the skipped blocks will work as constraints in Turbo decoding, which will improve the decoding bit error rate performance of Turbo codes. This is shown as "scheme 1" diagram in Figure 5.

Another way to take advantage of the zero motion skip blocks is not to encode the zero motion skip blocks by Wyner-Ziv encoder, since we know these blocks are the same as the co-located blocks in the reference frame. For a Turbo encoder, the number of syndrome bits generated can be calculated as follows:

$$N_s = \left(\frac{1}{r_c} - 1\right) \times N_{in} \tag{14}$$

where r_c is the code rate of the Turbo encoder, and N_{in} is the number of input bits. Since we know the skipped blocks in both encoder and decoder, we do not have to feed those bits into the Turbo encoder. In this case, the number of bits to be input to the Turbo encoder is reduced. From Equation (14), for the same number of syndrome bits, if N_{in} is reduced, r_c will be reduced too. Therefore, if we need not to encode the skipped blocks, we will have a lower code rate Turbo code. As we know, the low code rates Turbo codes perform better than the high code rate ones. This scheme is shown as "scheme 2" diagram in Figure 5.

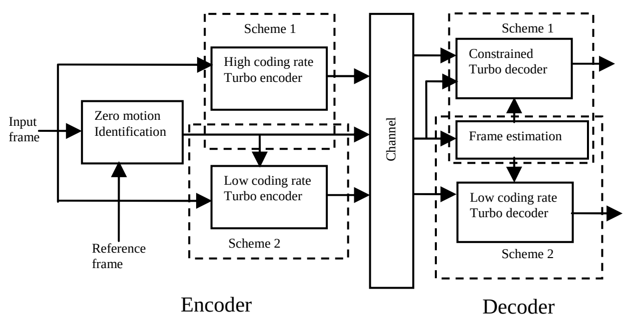

Figure 5: Two schemes of the proposed Distributed Video Coding with skipped zero motion blocks.

.

A more general approach to distributed source coding is presented in [10]. We proposed an improved Turbo code for distributed source coding, in which the parity bits remain unchanged. The unchanged parity bits actually work as a constraint in state transition in the decoding process. It indicates that some state of the transition is impossible, and thus reduces the error probability. In the scheme 1 illustrated in Figure 5, we indicate that, in distributed video coding, if zero motion blocks are identified and the decoder is informed, the decoder will know that, besides the parity bits, some of the systematic bits also remain unchanged. In this case, the constrained bits are all the parity bits as well as some of the systematic bits. Based on this principle, we have developed an improved BCJR algorithm in [6] in order to take advantage of the unchanged parity bits as well as the systematic bits to achieve improved compression efficiency for DVC.

For the scheme 2 illustrated in Figure 5, since fewer input bits are fed into the encoder, the resulting output will have fewer bits too and this will also improve the performance of DVC. A natural question to ask will be which of these two schemes is better in term of end-to-end rate distortion measure. We have carried out simulations to compare the performance of these two schemes.

To test the performances of both schemes, 10000 randomly generated binary sequences are used. For scheme 1, each sequence is 2048 bits, with half of them assumed to remain unchanged in both encoder and decoder. For scheme 2, each sequence is 1024 bits, and all of them have the chance to be changed. To generate the same number of syndrome bits, then, in the first scheme, Turbo codes with code rate of 4/5 are used, while in the second scheme, Turbo codes with code rate of 2/3 are used. The first scheme uses the constrained Turbo codes algorithm to decode, while the second scheme uses the improve Turbo codes for distributed source coding to decode. Figure 6 illustrates the decoding BER performances of the two schemes. To compare the performances with the case without zero motion blocks identification, the performance for Turbo codes for distributed source coding at rate 4/5 is also presented.

From Figure 6, it is clear that both schemes have a significant improvement in the decoding BER performance over the scheme without the knowledge of the unchanged bits. It is also clear that the scheme 2 as illustrated in Figure 5 achieves even better performance.

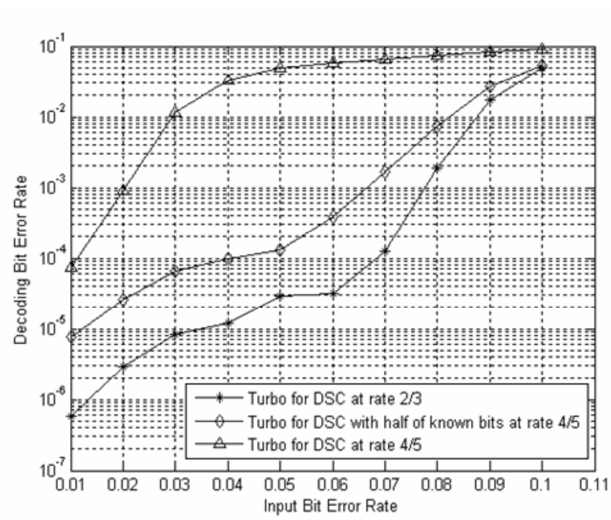

Figure 6: Decoding BER performances of the two proposed schemes.

The additional performance improvement of scheme 2 is actually due to its more favorable error distribution characteristics. This is another example to confirm that with fixed amount of prediction errors, more concentrated error distribution will result in an improved performance in DVC.

The difference between the two schemes in Figure 5 is that, in scheme 1, the changed and unchanged bits are mixed together, while in scheme 2, the changed and unchanged bits are separated. This is equivalent to that, for the same number of changed bits (prediction errors), the distribution of the errors in scheme 2 is more concentrated than that in scheme 1. The experimental results confirm that scheme 2 clearly outperforms scheme 1. Therefore, from an error control coding point of view, we arrive at the same conclusion: to improve the compression efficiency of DVC, the prediction error should be made as concentrated as possible.

6. CONCLUSIONS

In this paper, we reported the investigation about the relationship between the distribution of prediction errors and compression efficiency of distributed video coding. We demonstrate such relationship based on theoretical analysis, practical simulations, and from the error control coding point of view. All these methodologies arrive at the same conclusion: in distributed video coding, in order to improve the coding efficiency, the prediction errors should be made as concentrated as possible.

This study offers the researchers in DVC a new direction to explore about how to improve the video coding efficiency beyond the current existing schemes in improving the motion search at decoder and in improving efficiency of the used error control codes. We conclude that, in addition to better prediction, we should try to make the prediction errors distributed in as fewer blocks as possible. This investigation also explains clearly why the schemes proposed in [6][7] are able to achieve the desired performance enhancement.

We are currently investigating more practical schemes that will make the prediction errors in distributed video coding more concentrated to achieve additional performance enhancement that cannot be achieved by current DVC schemes. With these new insights, we expect more schemes will be developed to further improve the compression efficiency in distributed video coding.

7. REFERENCES

[1] B. Girod, A. Aaron, S. Rane and D. Rebollo-Monedero, "Distributed video coding," *Proceedings of IEEE*, Special Issue on Video Coding and Delivery, vol. 93, no. 1, pp. 71-83, Jan. 2005.

[2] M. Wu, G. Hua, C.W. Chen, "Syndrome-Based Light-Weight Video Coding for Mobile Wireless Application", in *Proc, of IEEE ICME*, 2006.

[3] A. Aaron, S. Rane, and B. Girod, "Wyner-ziv video coding with hashbased motion compensation at the receiver," in *Proc. IEEE ICIP*, Singapore, October 2004.

[4] M. Wu, A. Vetro, J.S. Yedidia, H. Sun, and C.W. Chen, "A study of encoding and decoding techniques for syndrome based video coding," in *Proc. IEEE ISCAS*, Kobe, Japan, May 2005, vol. 4, pp. 3527-3530.

[5] D. Varodayan, A. Aaron and B. Girod, "Rate-adaptive codes for distributed source coding", *EURASIP Signal Processing Journal*, Special Section on Distributed Source Coding, vol. 86, no. 11, pp. 3123-3130, Nov. 2006.

[6] G. Hua, and C.W. Chen, "Low Punctured Turbo Codes and Zero Motion Skip Encoding Strategy for Distributed Video Coding", in *Proc.of Globecom 2006*, San Francisco, USA, Nov., 2006.

[7] G. Hua and C.W. Chen, "Distributed Video Coding with Zero Motion Skip and Efficient DCT Coefficient Encoding", *Proc. IEEE ICME*, Hannover, Germany, June 2008

[8] D.Varodayan, http://www.stanford.edu/%7Edivad/ldpca.html, 2005.

[9] G. Hua, D. Song, D. Hench and C.W. Chen, "H.264 Scalable Video over Finite-State Markov Chain Wireless Channels", *Proceedings of SPIE*, Vol. 6777, 2007.

[10] G. Hua and C.W. Chen, "Distributed Source Coding in Wireless Sensor Networks", *Proc. Second International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks* (QShine), Orlando, USA, August, 2005.