LDPC Code Design for Asynchronous Slepian-Wolf Coding

Zhibin Sun, Chao Tian, *Member, IEEE*, Jun Chen, *Member, IEEE*,
and Kon Max Wong, *Fellow, IEEE*

***Abstract*—We consider asynchronous Slepian-Wolf coding where the two encoders may not have completely accurate timing information to synchronize their individual block code boundaries, and propose LDPC code design in this scenario. A new information-theoretic coding scheme based on source splitting is provided, which can achieve the entire asynchronous Slepian-Wolf rate region. Unlike existing methods based on source splitting, the proposed scheme does not require common randomness at the encoder and the decoder, or the construction of super-letter from several individual symbols. We then design LDPC codes based on this new scheme, by applying the recently discovered source-channel code correspondence. Experimental results validate the effectiveness of the proposed method.**

I. Introduction

In the seminal work [1], Slepian and Wolf showed that it is possible to compress two dependent sources in a distributed manner, at rates no larger than those needed when they are compressed jointly. More precisely, when two discrete memoryless sources X and Y, jointly distributed as Q_{XY} in the alphabets $\mathcal{X}$ and $\mathcal{Y}$, are separately compressed using block codes of length-n at rates R_1 and R_2, respectively, both sources can be reconstructed with asymptotically diminishing error probability at a central decoder, using any rates (R_1, R_2) such that

$$R_1 > H(X|Y),\ R_2 > H(Y|X),\ R_1 + R_2 > H(X,Y). \quad (1)$$

This rate region is illustrated in Fig. 1.

This result has been generalized in various ways [2]–[5], one of which is the asynchronous Slepian-Wolf (A-SW) coding scenario considered by Willems [4]. This consideration is practically important, because although we can assume perfect synchronization in the simplistic Slepian-Wolf setting such that the length-n block codes can be applied on each corresponding block pair, in practice the two encoders may not have such a perfectly accurate global clock to synchronize their block code boundaries. It was shown in [4] that even in this case, the rate region result (1) still holds, as long as the decoder is aware of this asynchronism. Thus, the restriction on the synchronization is in fact unnecessary and can be effectively removed, when the system is properly designed.

The material in this paper was presented in part at IEEE International Symposium on Information Theory, Seoul, Korea, Jun.-Jul., 2009.

Z. Sun was with the Department of Electrical and Computer Engineering, McMaster University, Hamilton, ON, Canada. He is now with Hydro One Networks Inc., Toronto, ON, Canada.

C. Tian is with AT&T Las–Research, Florham Park, NJ, USA.

J. Chen and K. M. Wong are with the Department of Electrical and Computer Engineering, McMaster University, Hamilton, ON, Canada.

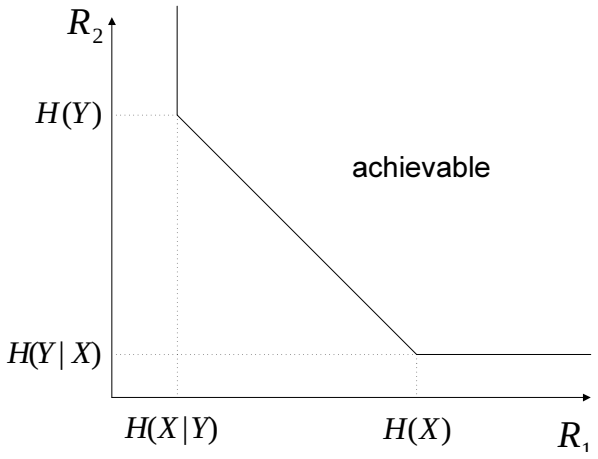


Fig. 1. The achievable rate region for Slepian-Wolf coding.

In this work, we consider the code design in the A-SW setting. Formally, the encoders are associated with an integer delay pair (d_x, d_y), each in the range of 0 to $n-1$. The values of d_x and d_y are unknown to the encoders but known to the decoder. Thus the q-th block of the source X consists of $\{X(t)\}_{t=qn+1+d_x,\ldots,(q+1)n+d_x}$ and the q-th block of the source Y consists of $\{Y(t)\}_{t=qn+1+d_y,\ldots,(q+1)n+d_y}$. It is easy to see that for the two corner points of the achievable rate region, this problem is not very different from that in the synchronous case. Thus our focus is on the general rate pairs, i.e., rate pairs on the dominant face of the rate region.

Previous works on the synchronous Slepian-Wolf (S-SW) code design mostly focus on the corner points of the rate region [7]–[10], with only a few exceptions [11] [12]. The code design at the corner points is reasonably well understood, particularly with the recent development [13], where it was shown that when encoding Y^n with side information X^n at the decoder, the error probability of any single linear coset code is exactly the decoding error probability of the same channel code on a corresponding channel under optimal decoding or belief propagation decoding. Thus the linear Slepian-Wolf code design problem at a corner point can be conveniently converted into the code design problem for a specific channel. Though this connection was mentioned in earlier works [14], [15], it was made precise in [13] for general (non-symmetric non-binary) sources.

The code design for the S-SW problem cannot be applied directly to the asynchronous case. One may wonder if the information-theoretic coding scheme given in [4] can be

used for such a purpose, however, it unfortunately requires optimization of the auxiliary random variable and complex joint typicality encoding usually seen in quantization modules, thus not convenient for practical code design. The usual time-sharing approach to achieve general rate pairs in S-SW coding also does not apply in the asynchronous setting. In [16], an information-theoretic scheme based on source splitting was given to overcome this difficulty due to asynchronism, by introducing common randomness at one encoder and the decoder. However, common randomness is not desirable in practical systems and should be avoided if possible.

It is clear that a new coding approach is needed for the A-SW code design, and since Slepian-Wolf coding is well understood for the corner points of the rate region, it is also desirable to utilize these existing results. Indeed, in this work we first present an information-theoretic scheme based on source splitting, which does not require common randomness (or the construction of a threshold function which operates on multiple source symbols as a super-letter [12]). Then based on this coding scheme, we utilize the source-channel correspondence result in [13] to design good LDPC codes. Experimental results confirm the effectiveness of the proposed design. Moreover, the proposed method does not significantly increase either system design or system implementation complexity compared to those of synchronized systems, and the performance is also comparable to state of the art design for synchronized systems in the literature.

It should be clarified at this point that the asynchronism here only refers to the mismatch in the block code boundaries of the two encoders, but not the timing mismatch in the sampling process. When sampling a continuous process, such timing inaccuracy may incur uncertainty in the probability distribution P_{XY}, and this problem is usually considered in the framework of universal Slepian-Wolf compression [5] [6], thus is beyond the scope of this work.

II. A New Source Splitting Scheme for A-SW Coding

In this section, we first briefly explain the difficulty of using time-sharing in the asynchronous setting and then review how source splitting together with common randomness can be used to overcome this difficulty, as proposed in [16]. Then a new information-theoretic scheme based on source splitting is proposed, which does not require common randomness. An overview of the proposed LDPC code design in the context of this new scheme will be subsequently discussed.

A. Time-sharing and Source Splitting With Common Randomness

It is easy to see that a simple time-sharing can be used to achieve any general rate pair in the achievable rate region for the S-SW problem. Essentially we only need to use the following two kinds of coding alternately: 1) first encode X directly, then encode Y with X as decoder side information, and 2) first encode Y directly, then encode X with Y as decoder side information. When the first kind of code is used with a proportion p, and the second is used with proportion $1-p$, it is clear that the following rate pair is achieved.

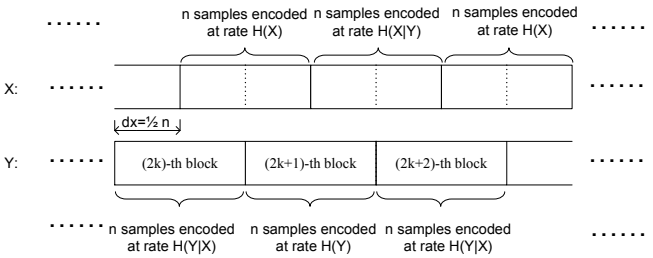


Fig. 2. The difficulty of time-sharing in the asynchronous setting.

$$R_1 = pH(X) + (1-p)H(X|Y)$$
$$R_2 = pH(Y|X) + (1-p)H(Y). \tag{2}$$

An example makes clear the difficulty of using this approach in the asynchronous setting. Let us assume $p = 0.5$ and the two kinds of codes used in timesharing both are of length-n. Thus in the synchronous setting, the first kind of code is used in the even block, i.e., the $(2m)$-th block, and the second kind of code is used in the odd block, i.e., the $(2m+1)$-th block. Now consider the asynchronous setting, and let $d_x = \frac{1}{2}n$ and $d_y = 0$, which are unknown to the encoders but known to the joint decoder. When the original time-sharing codes are used, it is clear that at the first half of the even blocks, source Y is encoded assuming X at the decoder, yet the source X is also encoded assuming Y at the decoder. This clearly results in decoder failure. Another way to see this is that in this portion, the sources are encoded with sum rate $H(Y|X) + H(X|Y)$, which is less than $H(X,Y)$, and thus it is impossible to ensure reliable communication; see Fig. 2 for an illustration.

To overcome this difficulty, common randomness was introduced in [16] such that source splitting can be applied. Let $\{T(t)\}_{t=1,\ldots,\infty}$ be a binary discrete memoryless process distributed in $\{0,1\}$ such that at each time instance, $\Pr[T(t)=1] = p$; it is independent of the sources (X,Y). Let this process be available at both the encoder observing X and the decoder. Without loss of generality, let us assume $\mathcal{X} = \{1,2,\ldots,|\mathcal{X}|\}$. Now define two new sources

$$Z = X \cdot T, \qquad W = X \cdot (1-T). \tag{3}$$

In other words, $\{Z(t)\}_{t=1,\ldots,\infty}$ is $\{X(t)\}_{t=1,\ldots,\infty}$ with certain position assigned to zero, while $\{W(t)\}_{t=1,\ldots,\infty}$ is $\{X(t)\}_{t=1,\ldots,\infty}$ with the complement positions assigned to zero. Thus the source X is split into two sources Z and W. Notice that by definition T is a deterministic function of Z, since T is only zero when Z is zero.

With these two new sources, the original problem is transformed into Slepian-Wolf coding of sources (Z,W) and Y. The encoding can now be performed in a sequential order as follows:

1) Encode the source Z conditioned on T;
2) Encode source Y assuming Z at the decoder;
3) Encode W assuming (Z,Y) at the decoder.

The rates at the two encoders are

$$R_1 = H(Z|T) + H(W|Z,Y)$$
$$R_2 = H(Y|Z) = H(Y|Z,T). \quad (4)$$

Observe that

$$R_1 + R_2 = H(Z|T) + H(W|Z,Y) + H(Y|Z)$$
$$= H(X,Y|T) = H(X,Y). \quad (5)$$

The rate pair in (4) is on the dominant face of the achievable rate region. By varying p from 0 to 1, all the rate pairs on the dominant face can be achieved, at least for the S-SW case.

A moment of thought should convince the readers that by applying the above block codes consecutively, the coding scheme can also be used in the asynchronous case without any change. Although this scheme can indeed achieve all rate pairs in the A-SW setting, it requires common randomness at one encoder and the decoder, which is not desirable in practice because the common randomness essentially requires the system to be more flexible in order to handle a large number of the side information erasure patterns. This problem is exacerbated in practice since pseudo-randomness must be used to replace this common randomness, which introduces the possibility that certain pseudo-random erasure pattern blocks may incur large decoding error probability and it may occur periodically. Next we propose a new scheme which does not require the common randomness (or pseudo-randomness) and thus avoids this problem; this new method only requires the system to handle a few side information erasure patterns, and it can be understood as an efficient de-randomized version of the afore-given scheme using common randomness.

B. A Source Splitting Scheme Without Common Randomness

From the source splitting scheme with common randomness afore-given, we can make the following two observations:

1) In the second coding step, for each length-n block of source Y, there are approximately $p \cdot n$ source X samples available at the decoder; furthermore, the exact positions of these X samples are known at the decoder.
2) Though the random sequence $T(t)$ can potentially split $X(t)$ at each time instance into two random variables $Z(t)$ and $W(t)$, the overall effect is in fact to split the length-n source X sequence in the time domain, such that the first item above can be made true.

Based on these two observations, we propose the following information-theoretic scheme, which does not require common randomness. Instead of the random sequence $T(t)$, let us consider a deterministic one

$$T(t) = \begin{cases} 1 & \lfloor t \rfloor_n - d_x = 1, \ldots, k \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

where $\lfloor \cdot \rfloor_n$ is modulo-n operation. In other words, the first k positions in a block aligned with the length-n source X block are assigned 1, while the remaining $n-k$ positions are assigned 0. After we apply the first coding step in the original source splitting scheme, a partial X sequence is available at the decoder. Such a choice of $T(t)$ indeed approximately satisfies the first observation discussed above, regardless the exact value of d_x; see Fig. 3 for an illustration. At this point, the second coding step in the original source splitting scheme still cannot be directly applied, since now there is no explicit source Z to simplify the coding module.

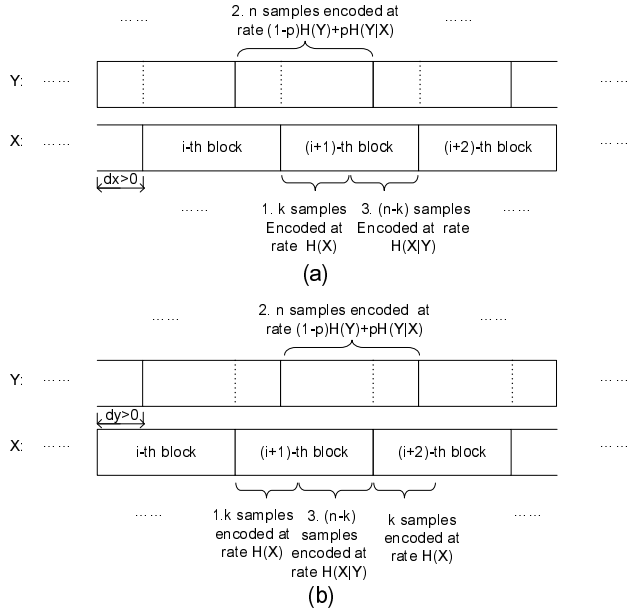


Fig. 3. Illustration of the new scheme without common randomness.

We can now focus on the second coding step of encoding source Y using a length-n block code, where k out of n of the corresponding X source samples are available at the decoder, whose positions are unknown to the encoder, but known to the decoder. These k positions of source X samples can be from a single length-n source X coding block, or two separate length-n source X coding blocks, which are illustrated in Fig. 3(a) and Fig. 3(b), respectively.

The following (random) coding scheme for the second step is the key difference between the one in [16] and the one proposed in this work. For convenience, let us denote the set of positions (indexed) within this length-n block for which source X samples are available at the decoder as $\mathcal{A}$, where $\mathcal{A} \subseteq \{1, 2, \ldots, n\}$. Consequently the set of the remaining positions within this block is denoted as $\mathcal{A}^c$.

- Random binning: each of y^n sequence is uniformly and independently assigned to one of 2^{nR_2} bin indices;
- Encoding: the encoder sends the bin index of the Y source sequence;
- Decoding: if the known length-k source X sequence from $\mathcal{A}$ is δ_1-typical, we then find a length-n source Y sequence in the corresponding bin, such that the following two typicality conditions hold (i) the length-k vector by collecting the Y samples at the positions corresponding to $\mathcal{A}$ is jointly δ_1-typical with the known length-k X sequence, and (ii) the length-$(n-k)$ vector by collecting the Y samples at the positions corresponding to $\mathcal{A}^c$ is δ_2-typical by itself.

In the above procedure, we have used the (weak) typicality definition in [22] (p. 51 and pp. 384-385). It is easier to bound

the decoding error if we view a single length-n source Y sequence as the combination of a length-k and a length-$(n-k)$ sequence. When both k and $n-k$ are sufficiently large, it is clear that with high probability, the original source sequence y^n indeed satisfy the two typicality conditions with high probability; we shall denote the probability of such error event that a source sequence fails one of the two conditions as P_e'. We only need to bound the probability that another Y sequence is decoded instead of the correct one. By the well-known properties of the typical sequences (see [22] pp. 51-53 and pp. 385-387), we see there are less than $2^{k(H(Y|X)+2\delta_1)}$ length-k source Y sequences that are jointly δ_1-typical with the known δ_1-typical length-k source X, and there are less than $2^{(n-k)(H(Y)+\delta_2)}$ δ_2-typical length-$(n-k)$ source Y sequences. Thus the decoding error in this step is bounded by

$$\begin{aligned} P_e &\le P_e' + 2^{-nR_2} 2^{k(H(Y|X)+2\delta_1)} 2^{(n-k)(H(Y)+\delta_2)} + \delta_1 \\ &= P_e' + 2^{-n\left[R_2 - \frac{k}{n}(H(Y|X)+2\delta_1) - \frac{n-k}{n}(H(Y)+\delta_2)\right]} + \delta_1, \quad (7) \end{aligned}$$

where the last term δ_1 accounts for the error event that the known X sample sequence at the decoder is in fact not typical. Thus if we choose sufficiently small δ_1 and δ_2, as long as

$$\begin{aligned} R_2 &> \frac{n-k}{n} H(Y) + \frac{k}{n} H(Y|X) \\ &\approx (1-p)H(Y) + pH(Y|X), \end{aligned}$$

the decoding error vanishes as $n \to \infty$. This solves the second step in the source splitting scheme without any common randomness. It should be noted that the scheme inherently requires both k and $n-k$ to be large.

For the third coding step, the decoding may have to wait until the next Y block is decoded; see Fig. 3(a) for an illustration. Nevertheless, it is not difficult to see that the third coding step in the original source splitting scheme can be used without much change. Thus we only need

$$\begin{aligned} R_1 &> \frac{k}{n} H(X) + \frac{n-k}{n} H(X|Y) \\ &\approx pH(X) + (1-p)H(X|Y). \quad (8) \end{aligned}$$

For sufficiently large n, by adjusting k, all rate pair on the dominant face of the Slepian-Wolf rate region can be effectively approached.

We have shown that for a fixed set $\mathcal{A}$, there indeed exists a sequence of codes that can approach the Slepian-Wolf limit. However, one key requirement in the A-SW problem is that a single code has to guarantee small error probability for all the possible cyclic shifts of $\mathcal{A}$ induced by the asynchronism. By refining the probability bound for decoding error given above, we can indeed show that such a sequence of codes exists. However instead of proving this more technical result here, in the appendix we provide a proof for an even stronger and relevant result that linear codes under the same source splitting paradigm can achieve the A-SW limit. This serves as a more rigorous proof, as well as the theoretical basis for designing the LDPC codes, which are indeed linear.

The following two observations are now worth noting. Firstly, the position of $T(t)$ being 1 does not need to be the first k positions. In fact, any pattern can be used with k positions assigned 1, as long as the pattern is repeated for all the blocks. More specifically, for $k/n = k'/n'$ where k' and n' are coprime of each other, we can choose $T(t)$ to be a sequence alternating between k' ones, and $n'-k'$ zeros; this choice gives the smallest number of side information erasure patterns that need to be considered among all (n,k) pairs of the same ratio. In the simulations given in Section IV, this kind of $T(t)$ sequences will always be assumed. Secondly the proposed scheme has the advantage that decoding errors do not propagate across blocks, because a single encoding (and decoding) step is essentially isolated to two consecutive blocks. The overall decoding procedure restarts when the first decoding step is used in each cycle.

C. Overview of the LDPC Code Design

In each step of the proposed source splitting scheme, with a given LDPC code, the encoding and decoding procedure is well known (see [10]), and thus we only need to focus on finding good codes. The overall code design consists of finding the following three codes.

1) A lossless code for encoding length-k source X sequences. This is a well understood module, and any good lossless compression algorithm can be used.
2) An LDPC code of rate approximately $pH(Y|X)+(1-p)H(Y)$ to encode the length-n source Y block, with length-k source X samples at the decoder, the positions of which are unknown to the encoder but known to the decoder. This step is discussed in more details in Section III.
3) An LDPC code of rate approximately $H(X|Y)$ to encode the rest $n-k$ samples in the source X block, with Y block as side information at the decoder. This step is similar, and in fact simpler than the second step. We also discuss this design step using the result of [13] in Section III.

III. Equivalent Channel Model and Code Design in the A-SW Setting

LDPC code in conjunction with belief propagation has shown extremely good performance in channel coding, which can in fact approach the capacity of many classes of channels [17]. The application of LDPC codes to the Slepian-Wolf problem was first suggested in [18] and further investigated in [10]. These results on S-SW coding mostly focus on the case that the (symmetric) source X and Y can be understood as connected by a symmetric channel, and sources with general distribution structure were largely overlooked.

Recently, a link between Slepian-Wolf coding and channel coding, referred to as source-channel correspondence, has been established in [13]. Through this link, coding for a corner point of the Slepian-Wolf rate region, i.e., the problem of source coding with decoder side information, can be transformed to a channel coding problem with the same error probability; consequently, given an arbitrary source Y and side information X, capacity-achieving LDPC codes for the equivalent channel can be designed using existing algorithms, which also approach the Slepian-Wolf limit of the source. In

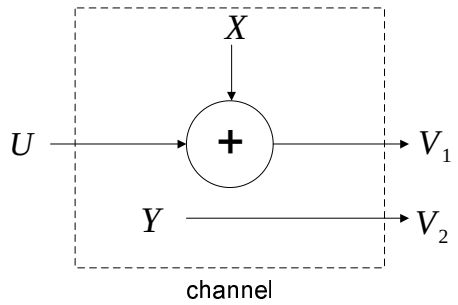


Fig. 4. The equivalent channel for source coding with decoder side information, where the channel input is U and the channel output is (V_1, V_2).

this section, we discuss the code design problems of step two and step three in our source splitting scheme using this link.

A. Source Coding With Decoder Side Information

We now briefly review the source-channel correspondence result in [13]. This will provide an explicit code design for the third step in the splitting scheme, i.e, encoding a block of X samples with the corresponding side information Y block at the decoder.

For notational simplicity, let us only consider the special setting when the source X is in certain finite field, but this requirement is by no means necessary; see [13]. Let the two sources in the alphabets $(\mathcal{X}, \mathcal{Y})$ be distributed as Q_{XY}, where $\mathcal{X}$ is a certain finite field and $\mathcal{Y} = \{1, 2, \ldots, J\}$.

The equivalent channel [13] with input U in the alphabet $\mathcal{X}$ and output $V = [V_1, V_2]$ is depicted in Fig. 4, where $V_1 = U \oplus X$ and $V_2 = Y$, with the addition $\oplus$ in the finite field $\mathcal{X}$. Here U is independent of X and Y. It can be shown that the capacity of the channel is achieved when U has a uniform distribution [13], resulting in

$$C = \log |\mathcal{X}| - H(X|Y). \tag{9}$$

The key result given in [13] is that when the same linear codes are used, the decoding error probability of the channel coding problem on this equivalent channel and that of the source coding with decoder side information problem are exactly the same, under maximum likelihood decoder or belief propagation decoding. Because in the source coding with side information problem, the compression rate using the linear coset code is given as $\log |\mathcal{X}| - R_c$, where R_c is the rate of the channel code, if we can design the LDPC code to approach the capacity of this channel, then the same code can be used to approach the Slepian-Wolf limit because of the fact $H(X|Y) = \log |\mathcal{X}| - C$, with the same error probability.

The density evolution algorithm developed in [19] can be used to design the parity check matrix for a given channel; more precisely, the degree distribution of the variable nodes and the check nodes in the bipartite graph [17] can be designed this way. An improved density evolution based algorithm, called discretized density evolution [20], was further developed in order to reduce the design complexity. Thus given a source and its side information structure, we can first transform the problem into an equivalent channel coding problem, then use the algorithm in [20] to design good LDPC code. This will yield good codes for the original source coding problem. We give a design example using this method in Section IV.

B. Source Coding With Partial Decoder Side Information

In the previous subsection, we see that the source-channel correspondence result [13] can be used to aid the design of LDPC code for the third step in the splitting scheme. For the second step, this method does not directly apply because only a partial side information sequence is available for the length-n source block.

To apply the source-channel correspondence result [13], an explicit single-letter probability structure between the source and the side information needs to be found. For this purpose, we return to the original source splitting scheme with common randomness. As we have discussed, the deterministic sequence $T(t)$ in fact approximates the effect of the randomized one. Due to this reason, we shall use the split source $Z(t)$ defined in (3) to replace the length-k partial side information sequence, the positions of which are unknown to the encoder.

Now assume the alphabet is given by $\mathcal{X} = \{1, 2, \ldots, |\mathcal{X}|\}$, then the joint distribution Q_{YZ} is clearly

$$\begin{aligned} Q_{YZ}(y, x) &= p \cdot Q_{YX}(y, x) \\ Q_{YZ}(y, 0) &= (1 - p) \cdot Q_Y(y). \end{aligned} \tag{10}$$

With this explicit distribution between the source Y and the side information Z, we can now again first apply the transform to the equivalent channel, then use the design algorithm in [20] to find good LDPC codes. This is turned into an almost identical design problem as that for the third step, with the only difference being the insertion of erasure symbol 0 into the original problem. In Section IV, we give a detailed example on the design of such codes.

IV. Design Examples and Numerical Results

In this section, we give several code design examples and results. We start with the third and the second steps in the splitting scheme, then the overall performance is discussed. To be more concrete, we focus on the distribution Q_{XY}

$$Q_{XY}(x, y) = \begin{cases} 0.45 & x = 1, y = 1 \\ 0.05 & x = 1, y = 2 \\ 0.09 & x = 2, y = 1 \\ 0.41 & x = 2, y = 2, \end{cases} \tag{11}$$

however, the design method is general and can be applied on any joint distributions. Note that we have chosen the alphabet $\{1, 2\}$ instead of the usual binary field $\{0, 1\}$ in order to be consistent with the discussion in the previous section. More precisely, when the source is X and side information is Y, the alphabet $\{1, 2\}$ of source X in this probability distribution is mapped in a one-to-one manner to the finite field $\{0, 1\}$ together with the associated finite field operations; when Y is the source, the alphabet of Y is mapped in a similar manner. This source distribution **cannot** be modeled as a binary symmetric channel and thus using codes designed for binary symmetric channels is not suitable.

A. Example of Source Coding With Decoder Side Information

We first consider the design for the third step in our splitting scheme, i.e., design LDPC code to encode X with side information Y based on the equivalent channel model shown in Fig. 4.

The initial log-likelihood ration (LLR) messages $\log \frac{P_{U|V}(u=1|v)}{P_{U|V}(u=2|v)}$ and their associated probabilities need to be determined in order to apply the discretized density evolution (DE) [20] [21] algorithm. Using the equivalent channel given in Section III-A, it is straightforward to verify that these LLRs and associated probabilities are as follows, assuming an all-zero codeword:

$$\log \frac{P_{U|V}(1|v)}{P_{U|V}(2|v)} = \begin{cases} \log 5 & \text{for } v=(1,1) \text{ with prob. } 0.45 \\ \log \frac{5}{41} & \text{for } v=(1,2) \text{ with prob. } 0.05 \\ \log \frac{1}{5} & \text{for } v=(2,1) \text{ with prob. } 0.09 \\ \log \frac{41}{5} & \text{for } v=(2,2) \text{ with prob. } 0.41 \end{cases}$$

Several good degree distributions with code rates 0.6, 0.602, 0.604, 0.607, 0.609, 0.610, 0.612 0.614 are obtained by setting the maximum variable node degree to be 20, $\delta(e_{l-1}-e_l) = 0.001$ and $\Delta\lambda = 0.0005$, where $\delta(e_{l-1}-e_l)$ and $\Delta\lambda$ are the parameters involved in the linear programming setting of [21]. As an example, the degree distribution of code rate 0.614 is given below:

$$\begin{aligned} \lambda(x) &= 0.213389x + 0.173764x^2 + 0.063x^3 \\ &\quad + 0.063x^4 + 0.056087x^5 + 0.036943x^6 \\ &\quad + 0.37x^7 + 0.42x^8 + 0.314816x^{19} \\ \rho(x) &= x^6. \end{aligned}$$

In the simulation of each code, source sequences of length 10^8 are generated by the joint distribution shown in (11). Each sequence is divided into 500 blocks with block length $n = 2 \times 10^5$ bits. Each block is decoded with the belief propagation algorithm, for which the number of iteration is limited to 150. The same belief propagation algorithm is also used in simulations discussed in later sub-sections.

Fig. 5 shows the performance of these Slepian-Wolf codes under two testing scenarios. In the first test, all the testing source sequences are ϵ-jointly-typical blocks with $\epsilon = 0.001$ [22], and in the second test the testing source sequences are randomly generated by the joint distribution in (11) so that they can be either typical or atypical. Apparently, we expect the first test to yield better results than the second one, since the codes are specifically designed for the given distribution. In Fig. 5, we see that the gap to the Slepian-Wolf limit of 0.57921 is 0.03 bit in the first test and 0.035 bit for the second. These results are comparable to the 0.033 and 0.06 bit gap for results on the symmetric sources using the LDPC codes in [10] [11], with code length 10^5, bit error rate (BER) less than 10^{-5}, and similar decoding algorithms; recall that the source distribution in (11) cannot be modeled as connected by a symmetric channel, and thus it is expected to be more difficult to code.

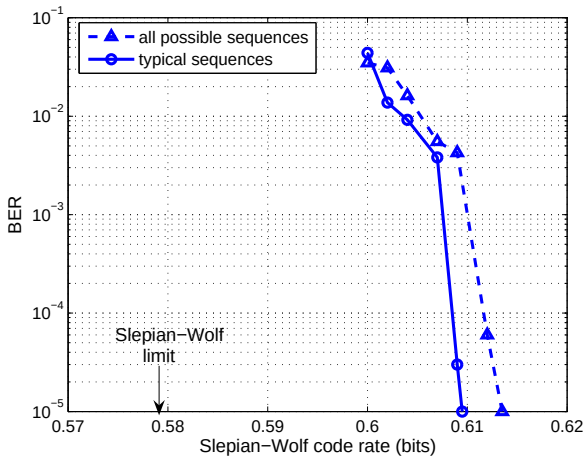


Fig. 5. The performances of 8 irregular LDPC codes of length 2×10^5, with side information Y at the decoder.

B. Example of Source Coding With Partial Decoder Side Information

We now consider the second step in our source splitting scheme, and focus on encoding for the mid-point on the dominant face of the Slepian-Wolf region, i.e, $p = 0.5$. We need to design code for source Y with the (randomized) partial side information Z at the decoder, whose joint distribution is

$$Q_{YZ}(y,z) = \begin{cases} 0.27 & y=1, z=0 \\ 0.225 & y=1, z=1 \\ 0.045 & y=1, z=2 \\ 0.23 & y=2, z=0 \\ 0.025 & y=2, z=1 \\ 0.205 & y=2, z=2 \end{cases}$$

The initial (LLR) messages and their associated probabilities are as shown below:

$$\log \frac{P_{U|V}(1|v)}{P_{U|V}(2|v)} = \begin{cases} \log(\frac{27}{23}) & \text{for } v=(1,0) \text{ with prob. } 0.27 \\ \log(9) & \text{for } v=(1,1) \text{ with prob. } 0.225 \\ \log(\frac{9}{41}) & \text{for } v=(1,2) \text{ with prob. } 0.045 \\ \log(\frac{23}{27}) & \text{for } v=(2,0) \text{ with prob. } 0.23 \\ \log(\frac{1}{9}) & \text{for } v=(2,1) \text{ with prob. } 0.025 \\ \log(\frac{41}{9}) & \text{for } v=(2,2) \text{ with prob. } 0.205 \end{cases}$$

Under the same assumption and the same parameters setting as the previous example, we apply the discretized density evolution algorithm [20] [21], and find several LDPC codes of rates $0.8060.8080.8100.8140.816$, respectively. The Slepian-Wolf limit is 0.7850, and thus these codes are less than 0.031 bit away from this lower bound. As an example, the degree distribution of code rate 0.816 is given below:

$$\begin{aligned} \lambda(x) &= 0.394235x + 0.212846x^2 + 0.011x^3 + 0.092328x^4 \\ &\quad + 0.078893x^5 + 0.210698x^{14} \\ \rho(x) &= 0.1x^2 + 0.9x^3 \end{aligned}$$

Since $p = 0.5$, we can choose the pattern of $T(t)$ to be a sequence alternating between one and zero, and subsequently

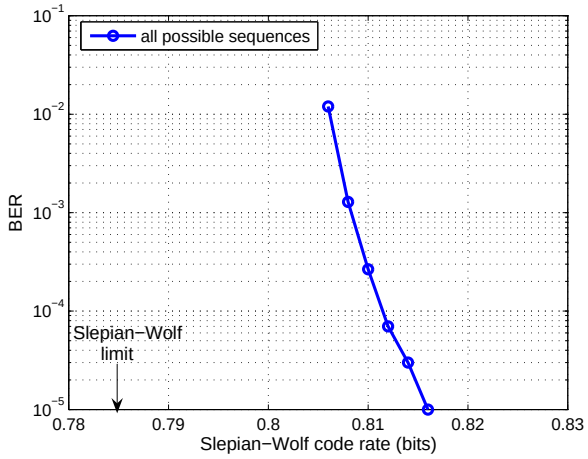


Fig. 6. The performances of 6 irregular LDPC codes of length 2×10^5, with partial X side information at the decoder.

TABLE I
PERFORMANCES OF DIFFERENT CODES UNDER VARIOUS d_y WITH $p = 0.5$.

R_2	d_y	BER
0.812	0	7.01×10^{-5}
	1	6.12×10^{-5}
0.814	0	3.13×10^{-5}
	1	4.21×10^{-5}
0.816	0	0.98×10^{-6}
	1	1.17×10^{-5}

we can assume $d_x = 0$ without loss of generality. By doing so, for a single fixed code, all the odd d_y values induce exactly the same error probability, and all the even d_y values induce exactly the same error probability. Thus we only need to perform the simulation for the two cases $d_y = 0$ and $d_y = 1$.

The performance of these codes is shown in Fig. 6, where the code length is again 2×10^5, and source sequences of length 10^8 are generated by the given joint probability without removing the atypical blocks. Results of the same code for both $d_y = 0$ and $d_y = 1$ are shown in Table I, for three different codes with rates $0.812, 0.814, 0.816$, respectively. It can be seen that for each code the error probability is consistent between the two d_y delay values, implying the effectiveness of this coding step in the asynchronous setting.

C. Overall Code Performance

We are now ready to evaluate the overall code performance in the A-SW setting. It is clear that by using variable-length codes, the first coding step in the source splitting scheme can achieve zero error with a negligible rate increase over $H(X)$ compared to the latter two steps, and thus we shall omit the error probability and assume the rate is simply $H(X)$ in the first step when calculating the overall rates and error probability.

The block length n is fixed at 3×10^5 for all simulation in this subsection and as such the length of code used in the third step is in fact $(1-p)n$. Each source sequence of length 3×10^8 is generated, and the error probability is averaged over both X and Y sequences, and thus the BER may be smaller than the corresponding ones shown in Table. I.

TABLE II
OVERALL CODE PERFORMANCE OPERATING UNDER DIFFERENT d_y FOR, $p = 1/2$, $p = 1/3$ AND $p = 2/3$, RESPECTIVELY.

Encoding rates in step 1, 2, 3	d_y	BER
0.5, 0.812, 0.311 ($p = 1/2$)	0	3.61×10^{-5}
	1	3.21×10^{-5}
0.5, 0.814, 0.311 ($p = 1/2$)	0	1.72×10^{-5}
	1	2.26×10^{-5}
0.5, 0.816, 0.311 ($p = 1/2$)	0	6.41×10^{-6}
	1	7.35×10^{-6}
0.33, 0.8902, 0.412 ($p = 1/3$)	0	9.79×10^{-6}
	1	1.00×10^{-5}
	2	1.01×10^{-5}
0.67, 0.746, 0.208 ($p = 2/3$)	0	2.81×10^{-6}
	1	4.83×10^{-6}
	2	2.87×10^{-6}

For $p = \frac{1}{2}$, in order to drive the overall error probability smaller, we choose a slightly larger rate in the third coding step than the ones given in Section IV-A. We again assume $d_x = 0$, and recall for this case we only need to test the cases $d_y = 0$ and $d_y = 1$. Using the same design method we also find codes for the rate pairs associated with time sharing parameter $p = \frac{1}{3}$ and $p = \frac{2}{3}$, respectively. The resulting performances of these codes are summarized in Table III, where we again assumed $d_x = 0$. The degree distributions in the second encoding step (rate 0.8902) and the third encoding step (rate 0.6185) for the case $p = \frac{1}{3}$ are:

$$\begin{aligned}\lambda(x) &= 0.480869x + 0.206888x^2 + 0.010614x^3 \\ &\quad + 0.070857x^4 + 0.094407x^5 + 0.136366x^{14} \\ \rho(x) &= 0.75x^2 + 0.25x^3.\end{aligned}$$

and

$$\begin{aligned}\lambda(x) &= 0.213124x + 0.170764x^2 + 0.06x^3 \\ &\quad + 0.06x^4 + 0.053352x^5 + 0.039944x^6 \\ &\quad + 0.4x^7 + 0.45x^8 + 0.317816x^{19} \\ \rho(x) &= x^6.\end{aligned}$$

The degree distributions in the second encoding step (rate 0.746) and the third encoding step (rate 0.623) for the case $p = \frac{2}{3}$ are respectively:

$$\begin{aligned}\lambda(x) &= 0.356665x + 0.248526x^2 + 0.028x^3 \\ &\quad + 0.027575x^4 + 0.173207x^5 + 0.028611x^6 \\ &\quad + 0.072795x^{15} + 0.064621x^{16} \\ \rho(x) &= 0.7x^3 + 0.3x^4.\end{aligned}$$

and

$$\begin{aligned}\lambda(x) &= 0.212855x + 0.167764x^2 + 0.057x^3 \\ &\quad + 0.057x^4 + 0.050621x^5 + 0.042944x^6 \\ &\quad + 0.43x^7 + 0.48x^8 + 0.320816x^{19} \\ \rho(x) &= x^6.\end{aligned}$$

To quantify the coding efficiency without the asynchronous requirement, we include the performances for $d_y = 0$ in Table

TABLE III
RESULTS IN TERMS OF DISTANCE TO THE SLEPIAN-WOLF LIMIT.

p	Target rates	Actual rates	Gap	BER
$\frac{1}{3}$	(0.7195, 0.8551)	(0.7457, 0.8902)	0.044	6.88×10^{-6}
$\frac{1}{2}$	(0.7896, 0.7850)	(0.8108, 0.8162)	0.038	9.96×10^{-6}
$\frac{2}{3}$	(0.8597, 0.7148)	(0.8743, 0.7460)	0.034	3.50×10^{-6}

II. The desired rate pair on the dominant face of the Slepian-Wolf region is shown together with the actual code rate pair (R_1, R_2) from the design in Table III. The average BER is kept below 10^{-5} and the gap is measured in terms of the Euclidean distance. These results are roughly on the same order as the best known designs [23] for symmetric sources to achieve general S-SW rate pairs, where by using IRA code with block length 10^5, a gap of 0.039 to the Slepian-Wolf limit is reported. Thus the design proposed in this work can achieve satisfactory performance in the S-SW setting, even it is in fact designed for the more general A-SW setting.

V. CONCLUSION

In this paper we introduced a new information-theoretic scheme based on source splitting for the asynchronous Slepian-Wolf problem. Combined with the source-channel correspondence result, the proposed LDPC code design leads to promising performance which is validated by simulations. The advantage of the proposed method is that it does not require common randomness and super-letter construction, and it can utilize existing code design results for the corner points of S-SW problem.

APPENDIX

In this appendix, we prove the sufficiency of linear codes in the A-SW setting. Since the sufficiency for the first and third steps is well known, we only need to focus on the second step, more precisely as follows: encoding a length-n source Y sequence, where $k = np$ out of n of the corresponding X side information samples are available at the decoder, whose positions (the non-erased pattern within a block due to shifts of a given $T(t)$) are unknown to the encoder, but known to the decoder. We have the following theorem.

Theorem 1: There exists a sequence of linear codes with rate approaching $pH(Y|X) + (1-p)H(Y)$, indexed by the code length n, with uniformly diminishing error probability for the above problem of source coding with partial decoder side information for all cyclic shifts.

We prove this theorem using the method of types [5], and particularly the techniques in [24]. The type of a sequence $x^n \in \mathcal{X}^n$ is the distribution P_X on $\mathcal{X}$ by

$$P_X(a) \triangleq \frac{1}{n} N(a|x^n) \text{ for every } a \in \mathcal{X},$$

where $N(a|x^n)$ is the number of occurrences of symbol a in the sequence x^n. Denote the set of types for length-n sequences in the alphabet $\mathcal{X}$ as $\mathcal{P}_n(\mathcal{X})$. The set of length-$n$ sequences of type P will be denoted as $\mathcal{T}_P^n$. Similar concepts can be defined for joint types, marginal types and conditional types, but are omitted here for brevity. We need the following elementary results in [5].

Lemma 1: The number of different types of sequences in $\mathcal{X}^n$ is less than $(n+1)^{|\mathcal{X}|}$, i.e., $|\mathcal{P}_n(\mathcal{X})| \le (n+1)^{|\mathcal{X}|}$.

Lemma 2: For any type P_X of sequences in $\mathcal{X}^n$, $|\mathcal{T}_{P_X}^n| \le 2^{nH(P_X)}$. Similarly, for any conditional type $P_{Y|X}$ of sequences in $\mathcal{Y}^n$, $|\mathcal{T}_{P_{Y|X}}^n(x^n)| \le 2^{nH(P_{Y|X})}$ for any x^n of the consistent type. Here we use the notation $H(P_{Y|X})$ to denote the conditional entropy of the given joint type.

Let Q be an arbitrary distribution on $\mathcal{X}$ and Q^n be the corresponding product distribution on $\mathcal{X}^n$. We have the following simple identity [24],

$$Q^n(x^n) = 2^{-n(D(P_X\|Q)+H(P_X))} \tag{12}$$

for any $P_X \in \mathcal{P}_n(\mathcal{X})$ and $x^n \in \mathcal{T}_{P_X}^n$.

Proof of Theorem 1: Consider the source sequence Y^n and the side information X^k. Divide Y^n into two parts with length k and $n-k$, respectively, the first of which is aligned with the known side information sequence X^k. We associate the first part with a generic random variable Y_1 and the second part with a generic random variable Y_2. It is understood that the sequence associated with Y_1 is of length k while the sequence associated with Y_2 is of length $n-k$.

Let us consider constructing linear encoding function $f(\cdot)$ using a $n \times l$ parity check matrix with entries independently and uniformly selected from $\mathcal{Y}$, where $\mathcal{Y}$ is assumed to be a finite field. For each joint type pair

$$(P_{Y_1\tilde{Y}_1X}, P_{Y_2\tilde{Y}_2}) \in (\mathcal{P}_k(\mathcal{Y}\times\mathcal{Y}\times\mathcal{X}), \mathcal{P}_{n-k}(\mathcal{Y}\times\mathcal{Y})), \tag{13}$$

we also write the marginal type associated with (Y_1, X) as P_{Y_1X}, and write the marginal type associated with Y_2 also as P_{Y_2}. The basic idea is to investigate the probability that the decoder decodes incorrectly some pair of sequences $(\tilde{y}_1^k, \tilde{y}_2^{n-k})$, when the original pair of sequences observed is in fact (y_1^k, y_2^{n-k}), by analyzing their joint types.

For each joint type pair in (13), let $N_f(P_{Y_1\tilde{Y}_1X}, P_{Y_2\tilde{Y}_2})$ denote the number of sequences (y_1^k, x^k, y_2^{n-k}) with $(y_1^k, x^k) \in \mathcal{T}_{P_{Y_1X}}^k$ and $y_2^{n-k} \in \mathcal{T}_{P_{Y_2}}^{n-k}$, such that for some $(\tilde{y}_1^k, \tilde{y}_2^{n-k}) \neq (y_1^k, y_2^{n-k})$ with $(y_1^k, \tilde{y}_1^k, x^k) \in \mathcal{T}_{P_{Y_1\tilde{Y}_1X}}^k$ and $(y_2^{n-k}, \tilde{y}_2^{n-k}) \in \mathcal{T}_{P_{Y_2\tilde{Y}_2}}^{n-k}$, and furthermore, the relation $f(y_1^k y_2^{n-k}) = f(\tilde{y}_1^k \tilde{y}_2^{n-k})$ holds. Due to the random construction of the linear function $f(\cdot)$, it is straightforward to see that for two distinct sequences, we have

$$\Pr[f(y_1^k y_2^{n-k}) = f(\tilde{y}_1^k \tilde{y}_2^{n-k})] = |\mathcal{Y}|^{-l}. \tag{14}$$

It follows that for any joint type pair in (13), we have

$$\begin{aligned}
&\mathbb{E}N_f(P_{Y_1\tilde{Y}_1X}, P_{Y_2\tilde{Y}_2}) \\
&\overset{(a)}{\le} |\mathcal{T}_{P_{Y_1X}}^k||\mathcal{T}_{P_{Y_2}}^{n-k}| 2^{kH(P_{\tilde{Y}_1|Y_1X})} 2^{(n-k)H(P_{\tilde{Y}_2|Y_2})} |\mathcal{Y}|^{-l} \\
&\overset{(b)}{\le} ||\mathcal{T}_{P_{Y_1X}}^k||\mathcal{T}_{P_{Y_2}}^{n-k}| 2^{kH(P_{\tilde{Y}_1|X})} \times 2^{(n-k)H(P_{\tilde{Y}_2})} |\mathcal{Y}|^{-l},
\end{aligned} \tag{15}$$

where in (a) we have used Lemma 2, and (b) is due to conditioning reduces entropy. By applying Markov's inequality

and defining $R = \frac{l}{n}\log|\mathcal{Y}|$, we have

$$\Pr\Bigg\{ N_f(P_{Y_1\tilde{Y}_1X}, P_{Y_2\tilde{Y}_2}) \geq |\mathcal{T}^k_{P_{Y_1X}}||\mathcal{T}^{n-k}_{P_{Y_2}}|2^{-n[R-\frac{k}{n}H(P_{\tilde{Y}_1|X})-\frac{n-k}{n}H(P_{\tilde{Y}_2})-\delta_n]}\Bigg\} \leq 2^{-n\delta_n}. \quad (16)$$

As afore-mentioned, there are at most a total of n side information erasure patterns (within the cyclic group), and the number of types is upper-bounded by $(k+1)^{|\mathcal{X}||\mathcal{Y}|^2}(n-k+1)^{|\mathcal{Y}|^2}$ by applying Lemma 1, and thus the probability of the event E_0, that for some type pair and some side information erasure pattern the condition in the brace of (16) is satisfied, is bounded by

$$\Pr(E_0) \leq n(k+1)^{|\mathcal{X}||\mathcal{Y}|^2}(n-k+1)^{|\mathcal{Y}|^2}2^{-n\delta_n}, \quad (17)$$

which is strictly less than one for sufficiently large n, by choosing δ_n appropriately such $\delta_n \to 0$ as $n \to \infty$; such a sequence of δ_n indeed exists by the δ-convention in [5] (p. 34). It is thus seen that there exists $f(\cdot)$ such that for sufficiently large n

$$N_f(P_{Y_1\tilde{Y}_1X}, P_{Y_2\tilde{Y}_2}) \leq |\mathcal{T}^k_{P_{Y_1X}}||\mathcal{T}^{n-k}_{P_{Y_2}}|2^{-n[R-\frac{k}{n}H(P_{\tilde{Y}_1|X})-\frac{n-k}{n}H(P_{\tilde{Y}_2})-\delta_n]}, \quad (18)$$

for all joint type pairs in (13) and all possible erasure patterns. By observing

$$N_f(P_{Y_1\tilde{Y}_1X}, P_{Y_2\tilde{Y}_2}) \leq |\mathcal{T}^k_{P_{Y_1X}}||\mathcal{T}^{n-k}_{P_{Y_2}}|. \quad (19)$$

we may write without loss of generality that

$$N_f(P_{Y_1\tilde{Y}_1X}, P_{Y_2\tilde{Y}_2}) \leq |\mathcal{T}^k_{P_{Y_1X}}||\mathcal{T}^{n-k}_{P_{Y_2}}|2^{-n|R-\frac{k}{n}H(P_{\tilde{Y}_1|X})-\frac{n-k}{n}H(P_{\tilde{Y}_2})-\delta_n|^+}, \quad (20)$$

where $|x|^+ \triangleq \max(x, 0)$.

We shall use the linear code $f(\cdot)$ with the above property (20) as the encoding function. The decoder now chooses $(\hat{y}_1^k, \hat{y}_2^{n-k})$ such that $\frac{k}{n}H(P_{\hat{Y}_1|X})+\frac{n-k}{n}H(P_{\hat{Y}_2})$ is minimized. The decoding error probability can be bounded as

$$\begin{aligned} P_e &\overset{(a)}{\leq} \sum N_f(P_{Y_1\tilde{Y}_1X}, P_{Y_2\tilde{Y}_2})2^{-k[D(P_{Y_1X}\|Q_{YX})+H(P_{Y_1X})]} \\ &\qquad \times 2^{-(n-k)[D(P_{Y_2}\|Q_Y)+H(P_{Y_2})]} \\ &\overset{(b)}{\leq} \sum 2^{-n|R-\frac{k}{n}H(P_{\tilde{Y}_1|X})-\frac{n-k}{n}H(P_{\tilde{Y}_2})-\delta_n|^+} \\ &\qquad \times 2^{-kD(P_{Y_1X}\|Q_{YX})}2^{-(n-k)D(P_{Y_2}\|Q_Y)} \\ &\overset{(c)}{\leq} (k+1)^{|\mathcal{X}||\mathcal{Y}|^2}(n-k+1)^{|\mathcal{Y}|^2}2^{-nE_1}, \end{aligned} \quad (21)$$

where (a) is by (12) and by taking summation over joint type pairs in (13) such that

$$\frac{k}{n}H(P_{\tilde{Y}_1|X}) + \frac{n-k}{n}H(P_{\tilde{Y}_2}) \leq \frac{k}{n}H(P_{Y_1|X}) + \frac{n-k}{n}H(P_{Y_2}), \quad (22)$$

because only this case may lead to a decoding error; in (b) we used (20) and Lemma 2; in (c) we define

$$E_1 = \min\Bigg[\Bigg|R - \frac{k}{n}H(P_{\tilde{Y}_1|X}) - \frac{n-k}{n}H(P_{\tilde{Y}_2}) - \delta_n\Bigg|^+ + \frac{k}{n}D(P_{Y_1X}\|Q_{YX}) + \frac{n-k}{n}D(P_{Y_2}\|Q_Y)\Bigg],$$

and the minimization is over the joint type pairs in (13) such that (22) holds. By choosing R such that

$$R > \frac{k}{n}H(Y|X) + \frac{n-k}{n}H(Y), \quad (23)$$

it is clear that E_1 is bounded below from zero when n is sufficiently large, which completes the proof. ■

REFERENCES

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

[2] R. F. Ahlswede and J. Korner, "Source coding with side information and a converse for degraded broadcast channels," *IEEE Trans. Inform. Theory*, vol. 21, no. 6, pp. 629-637, Nov. 1975.

[3] T. M. Cover, "A proof of the data compression theorem of Slepian and Wolf for ergodic sources," *IEEE Trans. Inform. Theory*, vol. 21, no. 2, pp. 226-228, Mar. 1975.

[4] F. Willems, "Totally asynchronous Slepian-Wolf data compression", *IEEE Transactions on Inform. Theory*, vol. 34, no. 1, pp. 35-44, Jan. 1988.

[5] I. Csiszar and J. Korner, *Information Theory: Coding Theorems for Discrete Memoryless Systems.* New York: Academic, 1981.

[6] I. Csiszar and J. Korner, "Towards a general theory of source networks," *IEEE Transactions on Inform. Theory*, vol. 26, no. 2, pp. 155-165, Mar. 1980.

[7] J. Garcia-Frias and Y. Zhao, "Compression of correlated binary sources using turbo codes," *IEEE Communications Letters*, vol. 5, no. 10, 417-419, Oct. 2001.

[8] J. Bajcsy and P. Mitran, "Coding for the Slepian-Wolf problem with turbo codes," in *Proc. IEEE GLOBECOM*, San Antonio, Texas, Nov. 2001, pp. 1400-1404.

[9] A. Aaron and B. Girod, "Compression with side information using turbo codes," in *Proc. Data Compression Conference*, Snowbird, Utah, Apr. 2002, pp. 252-261.

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

[11] D. Schonbery, S. S. Pradhan, and K. Ramchandran, "Distributed code constructions for the entire Slepian-Wolf rate region for arbitrarily correlated sources," in *Proc. Data Compression Conference*, Snowbird, Utah, Mar. 2004, pp. 292.

[12] T. P. Coleman, A. H. Lee, M. Medard, M. Effros, "Low-complexity approaches to Slepian-Wolf near-lossless distributed data compression," *IEEE Trans. Inform. Theory*, vol. 52, no. 8, pp. 3546-3561, Aug. 2006.

[13] J. Chen, D.-k He, E.-h Yang, "On the codebook-level duality between Slepian-Wolf Coding and channel coding," *Information Theory and Apllications Workshop*, San Diego, CA, Jan. 2007.

[14] A. Wyner, "Recent results in the Shannon theory," *IEEE Trans. Inform. Theory*, vol. 20, no. 1, pp. 2-10, Jan. 1974.

[15] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction," *IEEE Trans. Inform. Theory*, vol. 49, no. 3, pp. 626-643, Mar. 2003.

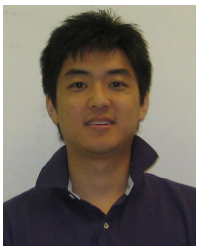
[16] B. Rimoldi and R. Urbanke, "Asynchronous Slepian-Wolf coding via source-splitting," in *Proc. IEEE Int.Symp. on Info. Theory*, Ulm, Germany, July 1997.

[17] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 619-637, Feb. 2001.

[18] T. Murayama, "Statistical mechanics of linear compression codes in network communication," *Europhysics Lett.*, preprint, 2001.

[19] T. J. Richardson and R. L. Urbanke, "The capacity of low-density parity check codes under message-passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 599-618, 2001.

[20] S.-Y. Chung, G. D. Forney, T. J. Richardson,and R. Urbanke, "On the design of low-desity parity check codes within 0.0045 dB of the Shannon limit," *IEEE Communications Letters*, vol. 5, no. 2, pp. 58-60, Feb. 2001.
[21] S.-Y. Chung,"On the construction of some capacity-approaching coding schemes," Ph.D. dissertation, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, Combrige, MA, 2000.
[22] T. M. Cover, J. A. Thomas, *Elements of Information Theory*, New York:Wiley, 1991.
[23] V. Stankovic, A. D.Liveris, Z. Xiong and C. N. Georghiades, "On code design for the Slepian-Wolf problem and lossless multiterminal networks," *IEEE Transactions on Information Theory*, vol. 52, no. 4, Apr. 2006.
[24] I. Csiszar, "Linear codes for sources and source networks: error exponents, universal coding," *IEEE Transactions on Information Theory*, vol. 28, no. 4, pp. 585–592, Jul. 1982.



Zhibin Sun received the B.Eng and M.A.Sc degrees in Electrical Engineering from McMaster University, Hamilton, ON, Canada, in 2007 and 2009, respectively. He joined the Department of Standards and New Technology Engineering at Hydro One Networks Inc. Toronto, ON, Canada in 2009, where he is now an Engineering Trainee.

Chao Tian (S'00, M'05) received the B.E. degree in Electronic Engineering from Tsinghua University, Beijing, China, in 2000 and the M.S. and Ph. D. degrees in Electrical and Computer Engineering from Cornell University, Ithaca, NY in 2003 and 2005, respectively.

Dr. Tian was a postdoctoral researcher at Ecole Polytechnique Federale de Lausanne (EPFL) from 2005 to 2007. He joined AT&T Labs–Research, Florham Park, NJ in 2007, where he is now a Senior Member of Technical Staff. His research interests include multi-user information theory, joint source-channel coding, quantization design and analysis, as well as image/video coding and processing.

Jun Chen (S'03, M'06) received the B.E. degree with honors in communication engineering from Shanghai Jiao Tong University, Shanghai, China, in 2001 and the M.S. and Ph.D. degrees in electrical and computer engineering from Cornell University, Ithaca, NY, in 2003 and 2006, respectively.

He was a Postdoctoral Research Associate in the Coordinated Science Laboratory at the University of Illinois at Urbana-Champaign, Urbana, IL, from 2005 to 2006, and a Josef Raviv Memorial Postdoctoral Fellow at the IBM Thomas J. Watson Research Center, Yorktown Heights, NY, from 2006 to 2007. He is currently an Assistant Professor of Electrical and Computer Engineering at McMaster University, Hamilton, ON, Canada. He holds the Barber-Gennum Chair in Information Technology. His research interests include information theory, wireless communications, and signal processing.

Kon Max Wong (SM'81-F'02) received his BSc(Eng), DIC, PhD, and DSc(Eng) degrees, all in electrical engineering, from the University of London, England, in 1969, 1972, 1974 and 1995, respectively.

He started working at the Transmission Division of Plessey Telecommunications Research Ltd., England, in 1969. In October 1970 he was on leave from Plessey pursuing postgraduate studies and research at Imperial College of Science and Technology, London. In 1972, he rejoined Plessey as a research engineer and worked on digital signal processing and signal transmission. In 1976, he joined the Department of Electrical Engineering at the Technical University of Nova Scotia, Canada, and in 1981, moved to McMaster University, Hamilton, Canada, where he has been a Professor since 1985 and served as Chairman of the Department of Electrical and Computer Engineering in 1986-87, 1988-94 and 2003-08. Professor Wong was on leave as Visiting Professor at the Department of Electronic Engineering of the Chinese University of Hong Kong from 1997 to 1999. At present, he holds the Canada Research Chair in Signal Processing at McMaster University. His research interest is in signal processing and communication theory and has published over 240 papers in the area.

Professor Wong was the recipient of the IEE Overseas Premium for the best paper in 1989, and is also the co-author of the papers that received the IEEE Signal Processing Society "Best Young Author" awards of 2006 and 2008. He is a Fellow of IEEE, a Fellow of the Institution of Electrical Engineers, a Fellow of the Royal Statistical Society, and a Fellow of the Institute of Physics. More recently, he has also been elected as Fellow of the Canadian Academy of Engineering as well as Fellow of the Royal Society of Canada. He was an Associate Editor of the IEEE Transaction on Signal Processing, 1996-98 and served as Chair of the Sensor Array and Multi-channel Signal Processing Technical Committee of the IEEE Signal Processing Society in 2002-04. Professor Wong was the recipient of a medal presented by the International Biographical Centre, Cambridge, England, for his "outstanding contributions to the research and education in signal processing" in May 2000, and was honoured with the inclusion of his biography in the two books: "Outstanding People of the 20th Century" and "2000 Outstanding Intellectuals of the 20th Century" published by IBC to celebrate the arrival of the new millennium