

# On Practical Design for Joint Distributed Source and Network Coding

Yunnan Wu, *Member, IEEE*, Vladimir Stanković, *Member, IEEE*, Zixiang Xiong, *Fellow, IEEE*, and Sun-Yuan Kung, *Fellow, IEEE*

**Abstract**—This paper considers the problem of communicating correlated information from multiple source nodes over a network of noiseless channels to multiple destination nodes, where each destination node wants to recover all sources. The problem involves a joint consideration of distributed compression and network information relaying. Although the optimal rate region has been theoretically characterized, it was not clear how to design practical communication schemes with low complexity. This work provides a partial solution to this problem by proposing a low-complexity scheme for the special case with two sources whose correlation is characterized by a binary symmetric channel. Our scheme is based on a careful combination of linear syndrome-based Slepian-Wolf coding and random linear mixing (network coding). It is in general suboptimal; however, its low complexity and robustness to network dynamics make it suitable for practical implementation.

**Index Terms**—Network coding, distributed source coding, distributed compression, multicast, low complexity.

## I. INTRODUCTION

CONSIDER the problem of communicating correlated information from two source nodes over a network to multiple destination nodes, illustrated by Fig. 1. Let  $\{(X_i, Y_i)\}_{i=1}^{\infty}$  be a sequence of independent drawings of a pair of correlated discrete random variables  $(X, Y)$ . This paper focuses on the special case where the correlation between binary  $X$  and  $Y$  is characterized by a binary symmetric channel (BSC)

$$Y = X + W \quad (1)$$

where  $W$ , the “correlation noise”, is independent from  $X$  and  $Y$ . Two source nodes,  $s_1$  and  $s_2$ , observe  $\{X_i\}$  and  $\{Y_i\}$ , respectively. There is a set of destination nodes,  $T = \{t_1, \dots, t_K\}$ , each of which wants to recover both sources. The source nodes

Manuscript received May 14, 2007; revised August 11, 2008. Current version published March 18, 2009. The material in this paper was presented in part at the 1st Workshop on Network Coding, Theory, and Applications (NetCod 2005), Riva del Garda, Italy, April 2005.

Y. Wu is with Microsoft Research, One Microsoft Way, Redmond, WA 98052 USA (e-mail: yunnanwu@microsoft.com).

V. Stanković is with the Department of Electronic and Electrical Engineering, University of Strathclyde, Glasgow, G1 1XW, U.K. (e-mail: vladimir.stankovic@eee.strath.ac.uk).

Z. Xiong is with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843 USA (e-mail: zx@ece.tamu.edu).

S.-Y. Kung is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA (e-mail: kung@princeton.edu).

Communicated by M. Effros, Associate Editor for Source Coding.

Color versions of Figs. 1–10 in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TIT.2009.2013016

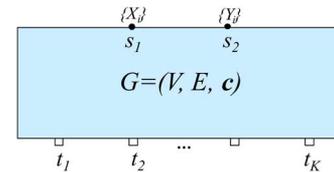


Fig. 1. The problem of multicasting correlated sources. Two correlated, discrete, binary, memoryless sources,  $\{X_i\}$  and  $\{Y_i\}$ , are available at nodes  $s_1$  and  $s_2$ , respectively. There are  $K$  destination nodes,  $t_1, \dots, t_K$ , each of which wants to recover both sources.

and the destination nodes are interconnected by a communication network modelled by a collection of noiseless links with rate constraints. Specifically, the network is represented as a graph  $G = (V, E, \mathbf{c})$ , where the set of nodes  $V$  and the set of edges  $E$  together specify the topology of the network, and  $\mathbf{c}$  is a vector that specifies the rate constraint  $c_e$  for each edge  $e \in E$ . Then, given the sources  $X$  and  $Y$ , the *admissible rate region* is defined as the set of  $\mathbf{c}$  that allows all destination nodes to reconstruct both  $X$  and  $Y$ . The objective is to characterize the admissible rate region and design practical schemes that can achieve points close to the boundary of the admissible region.

This problem can be viewed as a generalization of the distributed source coding problem and the problem of multicasting independent sources to the same set of destination nodes. The former aims at exploiting the redundancy that exists in the two correlated sources; the latter focuses on efficient strategies for managing information flow in a network. Both problems are well understood, both theoretically and practically.

Slepian and Wolf [2] considered lossless compression of two correlated discrete sources, and showed that independent encoding is as efficient as if the sources were compressed together, as long as joint decoding is performed. However, the proof of the Slepian-Wolf theorem is based on random binning, and is thus nonconstructive. A special case of Slepian-Wolf coding, known as source coding with decoder side information (or asymmetric Slepian-Wolf coding), treats the problem of compressing one source when the other is available at the decoder as side information. For this special case, a constructive scheme based on parity-check channel codes was given in [3]. The basic idea is to partition the codeword space of a “good” channel code into cosets and send only the index of a coset (syndrome) to which the source realization belongs. The scheme does not assume any particular correlation model among the sources, and if the employed channel code approaches the capacity of the “virtual correlation channel” between the two sources, the scheme will approach the Slepian-Wolf limit [4].

Following the syndrome-based approach [3], recently, a lot of work has been done in providing efficient practical code designs for the Slepian-Wolf problem (see [4] and the references therein) assuming a BSC or Gauss-Markov correlation model. Following the extension of [3] to the nonasymmetric setup [5], Stanković *et al.* [6] used turbo and low-density parity-check (LDPC) codes to approach any point on the Slepian-Wolf bound for the BSC source correlation model. Note that Slepian-Wolf coding addresses compression of correlated sources assuming direct noiseless links from the encoders to the decoder.

The problem of multicasting independent sources over a network of noiseless channels (with arbitrary topology) to the *same* set of destination nodes is addressed by recent advances on *network coding* [7], which refers to a scheme where each node in the network is allowed to generate output symbols by encoding (i.e., computing a certain function of) the symbols it received. See [8] for a comprehensive review.

Following constructive theoretical results showing that random linear network coding can achieve the multicast capacity (with high probability, and as finite field size goes to infinity)[9], [10], Chou *et al.* [11] proposed a practical scheme for performing network coding in real-world packet networks. Furthermore, the scheme is proven to be asymptotically capacity achieving in [12], [13]. Note that this practical network coding scheme is for multicasting from a single source to multiple destinations (it can also be generalized to the case of multicasting *independent* sources to the same set of destinations).

The focus of this paper is multicasting *correlated* sources, which has been addressed from a theoretical perspective. For two arbitrarily correlated sources  $X$  and  $Y$ , observed by nodes  $s_1 \in V$  and  $s_2 \in V$ , respectively, Song and Yeung [14]<sup>1</sup> and Ho *et al.* [15] showed that a vector  $c$  is admissible if and only if:

- i) each cut separating  $s_1$  from any destination node has at least capacity  $H(X|Y)$ ;
- ii) each cut separating  $s_2$  from any destination node has at least capacity  $H(Y|X)$ ;
- iii) each cut separating  $s_1$  and  $s_2$  from any destination node has at least capacity  $H(X, Y)$ .

Here it is easy to see that the above three conditions are necessary. The nontrivial part lies in establishing the sufficiency. Ho *et al.* [15] derived the error exponents for a random linear coding scheme, generalizing Csiszar's results [16] on linear Slepian-Wolf coding. Specifically, all nodes (including the source nodes) independently select random linear mappings from vectors of input bits onto vectors of output bits (as matrix multiplications of GF(2)). At each destination, decoding is done with the minimum entropy decoder or a maximum *a posteriori* probability decoder as in [16]. The work of [15] establishes that random linear coding is optimal (with high probability, as the block length approaches infinity). However, whereas random linear mixing offers a low complexity solution for single-source multicasting, the above random linear coding does not yield a low-complexity solution for multicasting correlated sources. This is because for single-source multicasting, decoding amounts to solving a system of linear inequalities with equal

number of unknowns and equations, whereas for multicasting correlated sources, a decoder has to decode more unknowns than the number of equations. Indeed, the decoding complexity of either the minimum entropy decoder or the maximum *a posteriori* probability decoder generally grows exponentially in the block length. For this reason, such a random linear coding approach is not suitable for practical implementation. As a reviewer commented, this is the case even in the original Slepian-Wolf coding setting. Practical Slepian-Wolf coding methods carefully design the encoding matrices to achieve good performance and low decoding complexity. However, as mentioned in [17], in multicasting correlated sources over a network, it is challenging to maintain a desirable structure in the codes seen by the receivers, considering the fact that the end-to-end transfer function depends on the operations done at the interior of the network, in addition to the operations at the sources.

Fig. 2 highlights the known and open problems in multicasting discrete sources. For the distributed compression (or Slepian-Wolf coding) problem, syndrome-based coding provides a practical solution (at least for simple correlation models, such as the BSC correlation model). For the single-source multicast problem, random linear mixing leads to a practical low-complexity implementation. Although the problem of multicasting correlated sources has been solved theoretically, the missing piece is a low-complexity coding scheme, which is the aim of this paper.

Motivated also by complexity considerations, previous work by Ramamoorthy *et al.* [17] investigated the performance of separate source and network codes. In the separation approach of [17], Slepian-Wolf coding is performed at the sources, and network coding is used only to stream the Slepian-Wolf coded bits while ignoring the correlation between the sources. At a destination, decoding is regarded successful as long as the set of coded bits received from the sources corresponds to a point in the Slepian-Wolf admissible region. It was shown in [17] that such a separation approach is optimal when there are two sources and two destinations, and examples were presented showing the separation approach is not optimal in general.

While the separation approach can potentially result in low complexity, a challenge with this approach is that the network coding component in the separation approach is by nature a multi-session network coding problem (where multiple multicast sessions share a network), which is still an open problem. The difficulty comes from the fact that the multicast sessions can have different destination sets. (If instead the sessions have a common destination set, then the problem is as easy as the single-source multicast problem.) Due to this difficulty, the admissible rate region associated with the separation approach remains unknown.

In this paper, we demonstrate a different path towards low-complexity coding. Just as the theory for the correlated multicasting problem [14], [15] generalizes Ahlswede *et al.*'s results [7] on single-source multicasting and Slepian-Wolf's results on distributed source coding, our proposed scheme leverages practical techniques for the two special cases, namely, random linear mixing and syndrome-based coding. In the proposed scheme, random linear mixing is applied within the network, yet desir-

<sup>1</sup>The results in [14] are for an arbitrary number of sources.

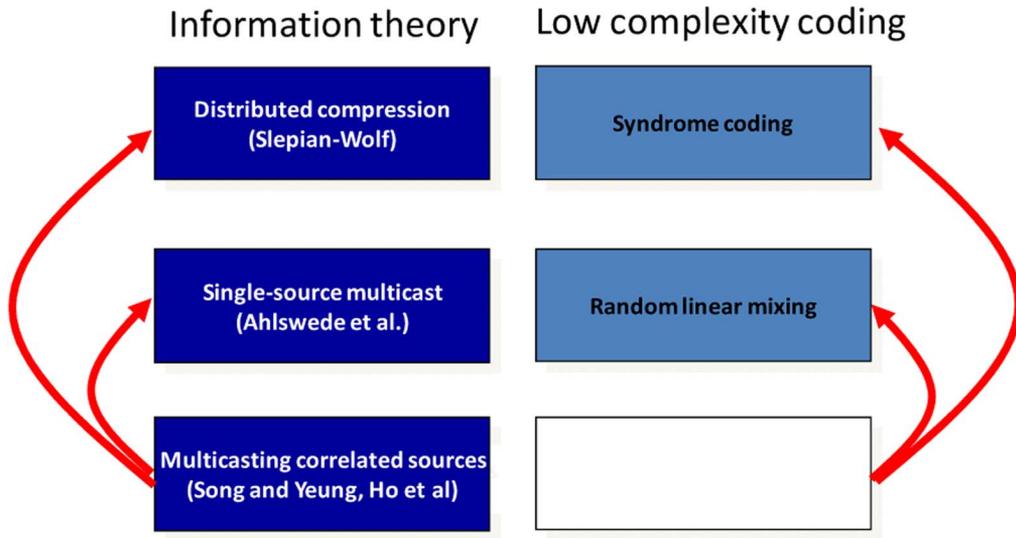


Fig. 2. The big picture: the left column shows information-theoretic results; the right column shows the corresponding low-complexity coding schemes.

able structure is preserved at the decoders to enable low complexity syndrome-based decoding. Furthermore, the admissible rate region is characterized as a concise linear program.

In Section II, we present a general design strategy that potentially leads to low-complexity implementations. Our proposed scheme is described in Section III. Performance analysis is shown in Section IV, and the last section concludes the paper.

### II. A GENERAL DESIGN STRATEGY

In this section we outline the main idea behind our design for multicasting two sources over a network of noiseless channels to multiple destination nodes, where each destination node wants to recover both sources. We keep the notation from the previous section and assume that  $X$  and  $Y$  are binary i.i.d. correlated sources such that  $Y = X + W$ , where  $W$  is an i.i.d. binary source.

The practical Slepian-Wolf coding scheme [3] is illustrated in Fig. 3 with  $\mathbf{x}$  (resp.  $\mathbf{y}$ ) being the realization of  $X$  (resp.  $Y$ ). Here we consider how to represent  $X$  at rate  $H(X|Y)$  and  $Y$  at rate  $H(Y)$  such that the two representations enable  $X$  and  $Y$  to be reconstructed. Since  $Y$  is represented at its entropy  $H(Y)$ , it can be reconstructed at the decoder. Thus the problem boils down to compressing  $X$  to the conditional entropy  $H(X|Y)$ , with side information  $Y$ . Let  $\mathbf{Q}$  be an  $n \times (n - k)$  parity check matrix of a linear channel code that is “good” for the BSC (1), with  $W$  interpreted as the channel noise. The  $n$ -bit realization of  $X$ , denoted by a row vector  $\mathbf{x}$ , is compressed into an  $(n - k)$ -bit syndrome vector  $\mathbf{xQ}$ . The decoder needs to recover  $\mathbf{x}$  from  $\mathbf{xQ}$  and  $\mathbf{y}$ . This can be done by (i) computing  $\mathbf{wQ} = \mathbf{yQ} - \mathbf{xQ}$ , (ii) reconstructing  $\mathbf{w}$  from  $\mathbf{wQ}$  using channel decoding, (iii) reconstructing  $\mathbf{x}$  as  $\mathbf{y} - \mathbf{w}$ .

Note that  $\mathbf{wQ}$  can be viewed as a compressed version of  $\mathbf{w}$ , from which  $\mathbf{w}$  can be recovered with high probability. This is possible because  $W$  has a nonuniform distribution. More formally, for linear compression of an i.i.d. source  $\{W_i\}$  with probability  $\Pr(W = 1) < 0.5$ , the following fact holds.

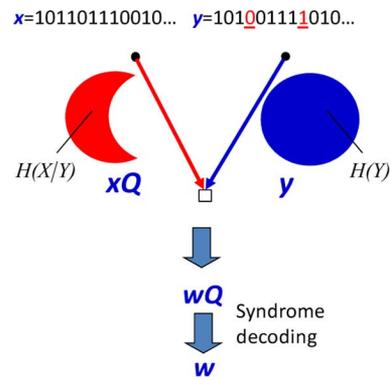


Fig. 3. Illustration of Slepian-Wolf coding scheme [3].

*Fact 1: Optimality of Linear Compression [18], [19]:* There exist a sufficiently large block size  $n$ , an  $n \times (n - k)$  matrix  $\mathbf{Q}$ , and a decoder such that the  $n$ -bit realization  $\mathbf{w}$  can be decoded from the  $(n - k)$ -bit linearly compressed version  $\mathbf{wQ}$  with probability arbitrarily close to 1, and the code rate  $(n - k)/n$  approaches the entropy  $H(W)$ .

Due to the close relation between linear source coding and linear channel coding, a good linear source code for an i.i.d. binary memoryless source  $\{W_i\}$  with distribution  $P_W$  can be found via a good linear channel code for the BSC with noise distribution  $P_W$ ; the decompression of the linear source code amounts to the syndrome-based decoding for the corresponding linear channel code. Thus, for some carefully designed  $\mathbf{Q}$  there exists a decoding algorithm that can decompress  $\mathbf{wQ}$ .

Our proposed design for multicasting correlated information relies on the above observation that the BSC correlation noise  $\mathbf{w}$  can be viewed as the source of redundancy, which can be exploited via linear compression. More specifically, our proposed design strategy is to use the network to:

- i) convey to each destination  $\mathbf{wQ} = \mathbf{yQ} - \mathbf{xQ}$ , with low complexity;

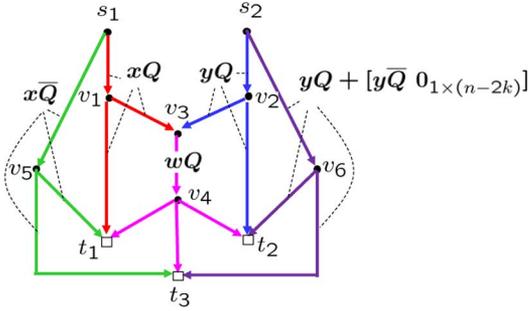


Fig. 4. An example network  $G = (V, E, c)$  given in [17]. For the case where  $H(X) = H(Y) = 1$ ,  $H(W) = 0.5$ , and the rate constraint on each edge is 0.5, we give a solution, shown by labelling the information flowing on each edge.

- ii) present to each destination  $t \in T$  sufficient descriptions  $\varphi_t(\mathbf{x}, \mathbf{y})$ , which, together with  $\mathbf{w}$ , allow  $\mathbf{x}$  to be reconstructed.

We illustrate our strategy with an example shown in Fig. 4 which was given in [17] to prove that separating Slepian-Wolf coding and network coding is in general suboptimal. Consider the case where  $H(X) = H(Y) = 1$ ,  $H(W) = 0.5$ , and the rate constraint on each edge is 0.5. Our low-complexity solution is given in Fig. 4, where the data to be carried on each edge is labelled. Let  $\bar{Q}$  be an  $n \times k$  matrix such that  $[\bar{Q} \ Q]$  is invertible. Assume  $k/n \leq 1/2$  and  $k/n \approx 1/2$ . All destinations receive  $\mathbf{w}Q$  and thus can recover  $\mathbf{w}$  with high probability. Once  $\mathbf{w}$  is recovered, it can be easily verified that all destinations can recover the sources  $(\mathbf{x}, \mathbf{y})$ . In this example, decoding at each destination node is done by (i) decompressing  $\mathbf{w}$  from  $\mathbf{w}Q$  via syndrome-based decoding, and (ii) solving a system of linear equations to recover  $\mathbf{x}$ .

To arrive at a concrete communication scheme, we need to address how to use the available network resources to meet the end-to-end traffic demand prescribed by the proposed strategy. This can be regarded as a network coding problem. Whereas network coding was previously examined primarily under the context of fulfilling certain end-to-end *communication* demand, the current demand formulation is quite unique in that it involves *computing a given function*  $\mathbf{y}Q - \mathbf{x}Q$ , where  $\mathbf{x}Q$  and  $\mathbf{y}Q$  are available at  $s_1$  and  $s_2$ , respectively. This presents new challenges for network coding. For example, consider the following problem formulation.

**Problem 1: (Network Arithmetic):** In a given graph  $(V, E)$ , suppose two source nodes  $s_1$  and  $s_2$  observe independent, uniform, discrete memoryless sources  $\{P_i\}_{i=1}^{\infty}$  and  $\{Q_i\}_{i=1}^{\infty}$ , respectively. There is a set of destination nodes  $T$ , each of which is required to reconstruct  $\{P_i + Q_i\}_{i=1}^{\infty}$ . What is the admissible rate region for meeting this demand?

There are many ways for conveying  $\{P_i + Q_i\}$  to the destinations. For example, one can transmit  $\{P_i\}$  and  $\{Q_i\}$  to an interior node in the network, which can then compute  $\{P_i + Q_i\}$  and multicast them to the destinations. At present we do not know how to characterize the admissible rate region over all possible strategies. Nevertheless, a simple, though suboptimal, approach is to transmit  $\{P_i\}$  and  $\{Q_i\}$  to all destinations. In the next section we present a specific practical design based on this suboptimal approach to Problem 1.

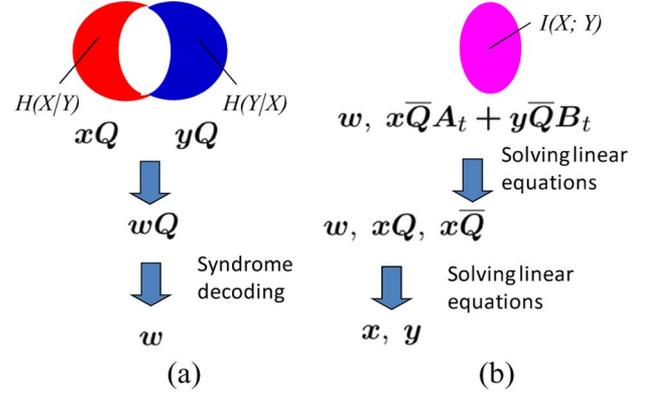


Fig. 5. The decoding process of the proposed scheme. (a) The node receives  $\mathbf{x}Q$  and  $\mathbf{y}Q$ . It computes  $\mathbf{w}Q$  and then performs syndrome-based decoding to recover  $\mathbf{w}$ . (b) Then the node solves a system of linear equations formed by  $\mathbf{w}$ ,  $\mathbf{x}Q$ , and  $\mathbf{x}Q A_t + \mathbf{y}Q B_t$  to recover  $\mathbf{x}$ ,  $\mathbf{y}$ .

### III. A PRACTICAL DESIGN SCHEME

We partition the available bit-rate resource  $\mathbf{c}$  into  $\mathbf{c}_1$  and  $\mathbf{c}_2$  with  $\mathbf{c} = \mathbf{c}_1 + \mathbf{c}_2$ . The first part  $\mathbf{c}_1$  is used to convey to each destination  $\mathbf{x}Q$  and  $\mathbf{y}Q$ . Specifically,  $s_1$  forms  $\mathbf{x}Q$ ;  $s_2$  forms  $\mathbf{y}Q$ ; then, random linear mixing (network coding) is applied over network resource  $\mathbf{c}_1$  to ensure that each destination gets  $\mathbf{x}Q$  and  $\mathbf{y}Q$ .

The second part  $\mathbf{c}_2$  is used to convey to each destination sufficient linear descriptions so that each destination can then solve a system of linear equations to recover  $\mathbf{x}$  and  $\mathbf{y}$ . Specifically, let  $\bar{Q}$  be an  $n \times k$  matrix such that  $[\bar{Q} \ Q]$  is invertible. For instance, if  $Q$  is in systematic form, i.e.,  $Q = [P^T \ I]^T$ , where  $I$  is an  $(n - k) \times (n - k)$  identity matrix,  $\bar{Q}$  can be chosen as  $[I \ 0]^T$ . The source node  $s_1$  forms  $\mathbf{x}Q$ ;  $s_2$  forms  $\mathbf{y}Q$ ; then random linear mixing (network coding) is applied over network resource  $\mathbf{c}_2$  to ensure that each destination gets a certain linear combination

$$\mathbf{x}\bar{Q}A_t + \mathbf{y}\bar{Q}B_t. \quad (2)$$

Specifically, a node in the network receives many input bits, each bit being a certain linear combination of  $[\mathbf{x}Q \ \mathbf{y}Q]$ ; it also outputs bits as linear combinations of the input bits, which are thus linear combinations of the source bits  $[\mathbf{x}Q \ \mathbf{y}Q]$ . Hence at the end, a destination node gets a certain linear combination of  $\mathbf{x}\bar{Q}$  and  $\mathbf{y}\bar{Q}$ , as in (2).

#### A. Decoding

Suppose for simplicity that both  $X$  and  $Y$  are uniform, hence  $H(X) = H(Y) = 1$ . Fig. 5 illustrates the decoding process, assuming that a destination node  $t$  receives  $\mathbf{x}Q$ ,  $\mathbf{y}Q$ , and a linear combination  $\mathbf{x}\bar{Q}A_t + \mathbf{y}\bar{Q}B_t$ . As shown in Fig. 5(a), a destination node computes  $\mathbf{w}Q$  from  $\mathbf{x}Q$  and  $\mathbf{y}Q$  and then performs syndrome-based decoding to recover  $\mathbf{w}$ . Note that the sizes of  $\mathbf{x}Q$  and  $\mathbf{y}Q$  are both approximately  $nH(X|Y) = nH(Y|X) = nH(W)$ . (More precisely, the size is  $nH(W) + o(n)$  where  $\lim_{n \rightarrow \infty} o(n)/n = 0$ .) As shown in Fig. 5(b), the destination node first computes  $\mathbf{x}\bar{Q}$  from  $\mathbf{w}$  and  $\mathbf{x}\bar{Q}A_t + \mathbf{y}\bar{Q}B_t$ . Specifically, this can be done by first computing

$$\mathbf{x}\bar{Q}(A_t + B_t) = (\mathbf{x}\bar{Q}A_t + \mathbf{y}\bar{Q}B_t) - \mathbf{w}\bar{Q}B_t \quad (3)$$

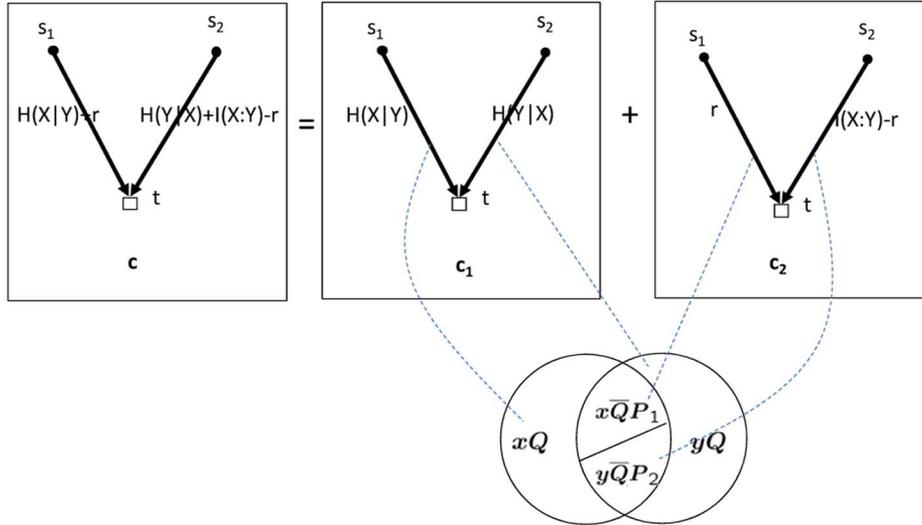


Fig. 6. Illustration of how the proposed scheme can achieve any point in the Slepian-Wolf region. The available resource is partitioned into two parts:  $\mathbf{c} = \mathbf{c}_1 + \mathbf{c}_2$ , where  $\mathbf{c}_1$  provides  $\mathbf{xQ}$  and  $\mathbf{yQ}$  to the receiver and  $\mathbf{c}_2$  provides  $\mathbf{xQP}_1$  and  $\mathbf{yQP}_2$  to the receiver.

and then inverting the matrix  $\mathbf{A}_t + \mathbf{B}_t$  to reconstruct  $\mathbf{x}\bar{\mathbf{Q}}$ . After  $\mathbf{x}\bar{\mathbf{Q}}$  is reconstructed, a system of linear equations formed by  $\mathbf{xQ}$ ,  $\mathbf{x}\bar{\mathbf{Q}}$ , and  $\mathbf{w}$  can be solved to recover  $\mathbf{x}$  and  $\mathbf{y}$ .

*B. An Example: Application to the Slepian-Wolf Scenario*

To better illustrate the proposed approach, we examine how the proposed scheme achieves an arbitrary point on the boundary of the Slepian-Wolf region. Note that several other approaches have been proposed for achieving an arbitrary point on the boundary of the Slepian-Wolf region. For instance, we can timeshare between two codes, one achieving the corner point  $(c_{s_1t}, c_{s_2t}) = (H(X|Y), H(Y))$  and the other achieving the corner point  $(c_{s_1t}, c_{s_2t}) = (H(Y|X), H(X))$ . Some other more direct methods can be found in [4]–[6] and references therein. The purpose of doing this exercise here is to facilitate the understanding of the proposed approach.

Suppose we want to achieve a point  $(c_{s_1t}, c_{s_2t})$  arbitrarily close to a point  $(H(X|Y)+r, H(Y|X)+I(X;Y)-r)$ , which lies on the boundary of the Slepian-Wolf region, for  $0 \leq r \leq I(X;Y)$ . As illustrated in Fig. 6, the available resource is partitioned into two parts:  $\mathbf{c} = \mathbf{c}_1 + \mathbf{c}_2$ , where  $\mathbf{c}_1$  provides  $\mathbf{xQ}$  and  $\mathbf{yQ}$  to the receiver and  $\mathbf{c}_2$  provides  $\mathbf{xQA}_t + \mathbf{yQB}_t$  to the receiver. Link  $s_1t$  is used to transmit  $\mathbf{xQ}$  of size  $\approx nH(X|Y)$  and  $\mathbf{xQP}_1$  of size  $\approx nr$ ; link  $s_2t$  is used to transmit  $\mathbf{yQ}$  of size  $\approx nH(Y|X)$  and  $\mathbf{yQP}_2$  of size  $\approx n(I(X;Y) - r)$ . The receiver  $t$  gets

$$\mathbf{x}\bar{\mathbf{Q}}[P_1 \ 0] + \mathbf{y}\bar{\mathbf{Q}}[0 \ P_2].$$

Thus  $\mathbf{A}_t = [P_1 \ 0]$  and  $\mathbf{B}_t = [0 \ P_2]$ . The decoding can succeed as long as  $\mathbf{A}_t + \mathbf{B}_t = [P_1 \ P_2]$  has full row rank. For this case, we can simply let  $[P_1 \ P_2] = \mathbf{I}$  to make it invertible.

In the above, we have shown how the proposed approach provides a solution to the Slepian-Wolf scenario. Due to the degenerated topology in this scenario, we are able to directly control  $\mathbf{Q}$  and  $\mathbf{A}_t + \mathbf{B}_t$ ; the entries in these matrices can be arbitrarily set. More generally, for an arbitrary network topology, we can still freely set the matrix  $\mathbf{Q}$ , but we cannot directly control the matrix  $\mathbf{A}_t + \mathbf{B}_t$  because  $\mathbf{A}_t$  and  $\mathbf{B}_t$  characterize the end-to-end

transfer function. For an arbitrary network topology, our proposed approach makes use of random linear mixing (network coding). In  $\mathbf{c}_1$ , random linear mixing is used to provide  $\mathbf{xQ}$  and  $\mathbf{yQ}$  to the receivers. In  $\mathbf{c}_2$ , random linear mixing is used to provide a suitable linear combination  $\mathbf{x}\bar{\mathbf{Q}}\mathbf{A}_t + \mathbf{y}\bar{\mathbf{Q}}\mathbf{B}_t$  to receiver  $t$  such that  $\mathbf{A}_t + \mathbf{B}_t$  is invertible with high probability.

To illustrate the power of random linear mixing, we now show how to use random linear mixing to achieve an arbitrary point on the boundary of the Slepian-Wolf region. Let  $\mathbf{M} \triangleq [P_1 \ P_2]$  and let  $\mathbf{M}_i$  denote the  $i$ -th column of  $\mathbf{M}$ . If  $\mathbf{M}$  is of size  $k \times k(1 + \epsilon)$  ( $\epsilon > 0$ ) and each entry of  $\mathbf{M}$  is uniformly drawn from  $\{0, 1\}$ , then for large  $k$ , with high probability  $\mathbf{M}$  will be of full rank  $k$ , which can be seen from the following derivation:

$$\begin{aligned} Pr[\mathbf{M} \text{ has rank } k] &= 1 - Pr[\exists \mathbf{x} \neq 0, \mathbf{xM} = 0] \\ &\geq 1 - \sum_{\mathbf{x} \neq 0} Pr[\mathbf{xM} = 0] \\ &= 1 - \sum_{\mathbf{x} \neq 0} \prod_{i=1}^{k(1+\epsilon)} Pr[\mathbf{xM}_i = 0] \\ &= 1 - \sum_{\mathbf{x} \neq 0} \prod_{i=1}^{k(1+\epsilon)} \frac{1}{2} \\ &= 1 - (2^k - 1) \cdot 2^{-k(1+\epsilon)} \\ &> 1 - 2^{-k\epsilon} \end{aligned}$$

Thus for this scenario, random coding performs very well, except for a small overhead ( $\epsilon$ ) and a small probability of failure  $2^{-k\epsilon}$ . For an arbitrary network topology, the analysis for random mixing is more involved; see, e.g., Ho *et al.* [9] and Sanders *et al.* [10]. The main point of the above exercise is to shed some light on the effectiveness of random coding. While for a fixed scenario there may be a better deterministic solution, random coding has many advantages. In particular, it enables robust, distributed solutions that are useful for practical, dynamic networks.

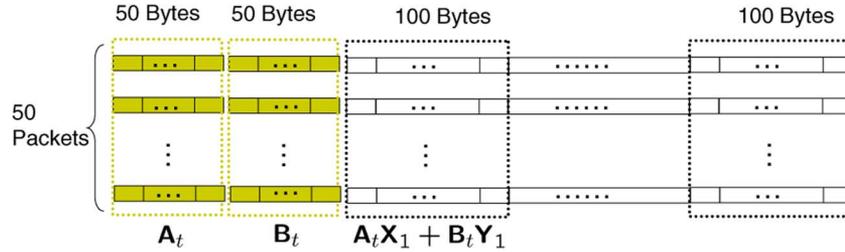


Fig. 7. The structure of packets received by destination  $t$ .

### C. Making System Practical for Packet Networks

The above scheme is not yet fully practical for use in a packet network. Since we assumed the operational field is  $\text{GF}(2)$ ,  $\mathbf{A}_t$  and  $\mathbf{B}_t$  are binary matrices. Knowledge of these two matrices is required at destination  $t$  for decoding. If the network is clock-synchronized and perfectly coordinated, it is in principle possible to have all the nodes in the network agree on some pre-determined coding operations. However, in real networks with lots of dynamics, conveying to  $t$  these matrices, which may have even more bits than the data, would be a difficult task. Another problem is that previous works on network coding are typically based on  $\text{GF}(2^p)$  where  $p$  is sufficiently large. These lead to an inconsistency in terms of the operating fields for linear syndrome-based coding and network coding.

We now explain how these issues can be addressed. For simplicity, the approach is explained via examples. Assume that  $\mathbf{xQ}$ ,  $\mathbf{yQ}$ ,  $\mathbf{x\bar{Q}}$ , and  $\mathbf{y\bar{Q}}$  are all computed over the binary field. Assume random linear mixing (network coding) is defined on  $\text{GF}(2^8)$ .

Consider the use of random linear mixing (network coding) in  $\mathbf{c}_2$ . Suppose  $\mathbf{x\bar{Q}}$  has 40 000 bits. We can packetize these 40 000 bits into 50 packets, each of length 100 bytes. This packetization operation can be represented as a one-to-one mapping

$$\psi : \text{GF}(2)^{1 \times 40\,000} \longrightarrow \text{GF}(2^8)^{50 \times 100}. \quad (4)$$

Thus the result  $\psi(\mathbf{x\bar{Q}})$  is a matrix of size  $50 \times 100$  with each element defined in  $\text{GF}(2^8)$ . The practical network coding scheme of [11] is used over the 100 source packets comprising the 50 rows of  $\psi(\mathbf{x\bar{Q}})$  from source  $s_1$  and the 50 rows of  $\psi(\mathbf{y\bar{Q}})$  from source  $s_2$ . We now briefly review the operations. As in [11], each node receives asynchronously via its incoming links a collection of packets, each in the form of

$$[\alpha \beta] \begin{bmatrix} \psi(\mathbf{x\bar{Q}}) \\ \psi(\mathbf{y\bar{Q}}) \end{bmatrix} \quad (5)$$

where the linear combination coefficients  $[\alpha_{1 \times 50} \beta_{1 \times 50}]$ , called the *global coding vector*, are recorded in the packet header. These packets are stored into a buffer as they arrive. Whenever there is a transmission opportunity available on one of its outgoing links, a node also generates an output packet by linearly combining the packets in its buffer with random coefficients, which again has the form of (5).

A destination  $t$  receives a collection of such mixture packets. Each mixture packet essentially represents an equation in terms of the source packets. Let  $[\mathbf{A}_t, \mathbf{B}_t]$  denote the global encoding matrix obtained by putting together the global encoding vectors

of the received packets; then  $t$  observes  $\mathbf{A}_t \cdot \psi(\mathbf{x\bar{Q}}) + \mathbf{B}_t \cdot \psi(\mathbf{y\bar{Q}})$ , where  $\cdot$  denotes multiplication in  $\text{GF}(2^8)$  (to distinguish from multiplication in  $\text{GF}(2)$ ).

The practical network coding scheme reviewed above is used both in  $\mathbf{c}_1$  for multicasting  $\mathbf{xQ}$  and  $\mathbf{yQ}$  to the destinations, and in  $\mathbf{c}_2$  for presenting sufficient linear descriptions of  $\mathbf{x\bar{Q}}$  and  $\mathbf{y\bar{Q}}$  to the destinations. For  $\mathbf{c}_1$ , a destination node needs to collect enough independent packets until it can solve the system of linear equations to recover the source packets. Then it can recover the packetized versions of  $\mathbf{xQ}$  and  $\mathbf{yQ}$  and solve  $\mathbf{w}$  by syndrome-based decoding.

The case for  $\mathbf{c}_2$  is more involved. If such 100 packets (50 from  $s_1$  about  $\psi(\mathbf{x\bar{Q}})$ , 50 from  $s_2$  about  $\psi(\mathbf{y\bar{Q}})$ ) were to be multicast to the destinations, the global encoding matrix  $[\mathbf{A}_t, \mathbf{B}_t]$  would need to have full column rank (100 in this example). However, this is stronger than necessary for our purpose. In fact, we only require  $\mathbf{A}_t + \mathbf{B}_t$  to have full rank 50, which translates to a demand of roughly half of the network resources required for multicasting the 100 packets. We perform the following computation:

$$(\mathbf{A}_t \cdot \psi(\mathbf{x\bar{Q}}) + \mathbf{B}_t \cdot \psi(\mathbf{y\bar{Q}})) - \mathbf{B}_t \cdot \psi(\mathbf{w\bar{Q}}) \quad (6)$$

$$= \mathbf{A}_t \cdot \psi(\mathbf{x\bar{Q}}) + \mathbf{B}_t \cdot \psi(\mathbf{x\bar{Q}}) \quad (7)$$

$$= (\mathbf{A}_t + \mathbf{B}_t) \cdot \psi(\mathbf{x\bar{Q}}). \quad (8)$$

In (6), the first term  $\mathbf{A}_t \cdot \psi(\mathbf{x\bar{Q}}) + \mathbf{B}_t \cdot \psi(\mathbf{y\bar{Q}})$  is received; the second term is computed from  $\mathbf{w}$ . The first equality is because the addition (subtraction) operation in any  $\text{GF}(2^p)$  is a bit-wise XOR operation. If  $\mathbf{A}_t + \mathbf{B}_t$  is invertible, then we can obtain  $\psi(\mathbf{x\bar{Q}})$  from (8). Next, we can recover  $\mathbf{x\bar{Q}}$  and then  $\mathbf{x}$  by solving a system of linear equations defined by  $\mathbf{xQ}$ ,  $\mathbf{x\bar{Q}}$ , and  $\mathbf{x}$ .

With this scheme, in each packet, the global encoding vector takes 100 bytes and the useful data are 100 bytes. Thus the overhead is still significant. To further reduce the overhead, we can amortize it by putting data corresponding to multiple blocks into one packet. Each block contains a distinct  $\mathbf{x}(i)\bar{Q}$  of length 40 000 bits; here  $i$  is the block index. For example, assuming the packet has 1 100 bytes, the packet format is illustrated in Fig. 7 and shown below

$$\alpha, \beta, \alpha \cdot \psi(\mathbf{x}(1)\bar{Q}) + \beta \cdot \psi(\mathbf{y}(1)\bar{Q}), \\ \dots, \alpha \cdot \psi(\mathbf{x}(10)\bar{Q}) + \beta \cdot \psi(\mathbf{y}(10)\bar{Q}). \quad (9)$$

## IV. PERFORMANCE ANALYSIS

In this section we derive the admissible rate region for the scheme proposed in Section III and compare it with the theoretically optimal admissible rate region. First we introduce our

notations. Given a graph  $G = (V, E, \mathbf{c})$ , a source node  $s$ , a destination node  $t$ , an  $s$ - $t$  flow is a nonnegative vector  $\mathbf{f}$  of length  $|E|$  satisfying the flow conservation constraint

$$\text{excess}_v(\mathbf{f}) = 0, \quad \forall v \in V - \{s, t\} \quad (10)$$

where

$$\text{excess}_v(\mathbf{f}) \equiv \sum_{e \in \text{In}(v)} f_e - \sum_{e \in \text{Out}(v)} f_e. \quad (11)$$

Let  $\mathcal{F}_{s,t}(r)$  denote the set of  $s$ - $t$  flows, each with its flow value equal to  $r$ . Then  $\mathbf{f} \in \mathcal{F}_{s,t}(r)$  if and only if

$$\begin{aligned} \mathbf{f} &\geq \mathbf{0} \\ \text{excess}_s(\mathbf{f}) &= -r \\ \text{excess}_t(\mathbf{f}) &= r \\ \text{excess}_v(\mathbf{f}) &= 0, \quad \forall v \in V - \{s, t\}. \end{aligned}$$

A useful property of  $\mathcal{F}_{s,t}(r)$  is its linearity in  $r$ , i.e.,

$$\mathcal{F}_{s,t}(r) = r\mathcal{F}_{s,t}(1) \equiv \{r\mathbf{f} \mid \mathbf{f} \in \mathcal{F}_{s,t}(1)\}. \quad (12)$$

The Max-Flow-Min-Cut Theorem says that the minimum cut capacity  $\rho_{s,t}(\mathbf{c})$  is equal to the maximum value of an  $s$ - $t$  flow within  $G = (V, E, \mathbf{c})$ . It follows then

$$r \leq \rho_{s,t}(\mathbf{c}) \iff \exists \mathbf{f}_{s,t} \in \mathcal{F}_{s,t}(r), \mathbf{f}_{s,t} \leq \mathbf{c}. \quad (13)$$

#### A. Admissible Rate Region for the Proposed Scheme

Recall from Section III that the proposed scheme partitions the available bit-rate resource in the network into two disjoint parts:  $\mathbf{c} = \mathbf{c}_1 + \mathbf{c}_2$ . The first part  $\mathbf{c}_1$  is used to multicast  $\mathbf{xQ}$  and  $\mathbf{yQ}$  from the two source nodes to the destination nodes, using network coding. Here network coding is used as a method for transferring some arbitrary data from  $s_1$  and  $s_2$  to the destination nodes. In other words,  $\mathbf{xQ}$  and  $\mathbf{yQ}$  are treated as independent sources. (As a matter of fact, if  $\mathbf{Q}$  is an ideal linear compressor for  $\mathbf{w}$ , then  $\mathbf{xQ}$  and  $\mathbf{yQ}$  are independent.)

Therefore, the underlying problem is to multicast two independent sources such that each destination can recover both sources. This problem is a simple extension of the single-source multicasting problem. For example, suppose two independent sources  $\mathbf{p}$  (with rate  $r_1$ ) and  $\mathbf{q}$  (with rate  $r_2$ ) are available initially at  $s_1$  and  $s_2$ , respectively, and a set of destinations  $T$  wants to receive  $\mathbf{p}$  and  $\mathbf{q}$ . Then the admissible rate region for such two-source multicasting problem is the set of  $\mathbf{c}$  such that

$$\mathbf{c} \geq \mathbf{f}_t^{(1)} + \mathbf{f}_t^{(2)}, \quad \forall t \in T \quad (14)$$

$$\mathbf{f}_t^{(1)} \in \mathcal{F}_{s_1,t}(r_1), \quad \forall t \in T \quad (15)$$

$$\mathbf{f}_t^{(2)} \in \mathcal{F}_{s_2,t}(r_2), \quad \forall t \in T. \quad (16)$$

To see this, note that we can convert the problem into an equivalent virtual multicast problem, by introducing a virtual source node  $\hat{s}$  that has a link with capacity  $r_1$  to  $s_1$  and a link with capacity  $r_2$  to  $s_2$ . The original problem is equivalent to the virtual single-source multicast problem from  $\hat{s}$  to  $T$  at rate  $r_1 + r_2$ , because we can assume edge  $\hat{s}s_1$  carries  $\mathbf{p}$  and edge  $\hat{s}s_2$  carries

$\mathbf{q}$  without loss of generality. Applying Ahlswede *et al.* [7]'s result to the virtual single-source multicast problem, we see that the admissible rate region is:

$$\mathbf{c} \geq \mathbf{f}_t, \quad \forall t \in T \quad (17)$$

$$\mathbf{f}_t \in \mathcal{F}_{\hat{s},t}(r_1 + r_2), \quad \forall t \in T \quad (18)$$

Note further that an  $\hat{s}$ - $t$  flow with rate  $r_1 + r_2$  from  $\hat{s}$  to  $t$  can always be partitioned into an  $s_1$ - $t$  flow with rate  $r_1$  plus an  $s_2$ - $t$  flow with rate  $r_2$ . Thus we obtain (14)–(16) as the admissible rate region.

Let us now return to our specific problem. From the above discussion, we conclude that the admissible rate region for the first part is the set of  $\mathbf{c}_1$  such that

$$\mathbf{c}_1 \geq \mathbf{f}_t^{(1)} + \mathbf{f}_t^{(2)}, \quad \forall t \in T \quad (19)$$

$$\mathbf{f}_t^{(1)} \in \mathcal{F}_{s_1,t}(H(X|Y)), \quad \forall t \in T \quad (20)$$

$$\mathbf{f}_t^{(2)} \in \mathcal{F}_{s_2,t}(H(Y|X)), \quad \forall t \in T. \quad (21)$$

The second part  $\mathbf{c}_2$  is used to provide each destination  $A_t \cdot \psi(\mathbf{xQ}) + B_t \cdot \psi(\mathbf{yQ})$  via random linear mixing (network coding), with  $\psi(\mathbf{xQ})$  injected at  $s_1$  and  $\psi(\mathbf{yQ})$  injected at  $s_2$ . Since the first part  $\mathbf{c}_1$  provides each destination with  $\mathbf{xQ}$  and  $\mathbf{yQ}$  and each destination node can decompress  $\mathbf{w}$  from  $\mathbf{yQ} - \mathbf{xQ} = \mathbf{wQ}$ , upon receiving  $A_t \cdot \psi(\mathbf{xQ}) + B_t \cdot \psi(\mathbf{yQ})$ , a destination can readily compute  $A_t \cdot \psi(\mathbf{xQ}) + B_t \cdot \psi(\mathbf{xQ})$ . It can then invert the matrix  $(A_t + B_t)$  to recover  $\mathbf{xQ}$  (cf. Fig. 5(b)). Here random linear mixing (network coding) is only used to provide roughly  $nI(X; Y)$  linear equations to each destination such that  $(A_t + B_t)$  has full column rank.

The above requirement for  $\mathbf{c}_2$  can be stated as a condition on the end-to-end transfer function. A linear network code on the graph induces a linear transfer function from the two inputs at  $s_1$  and  $s_2$  to the outputs at the receivers in  $T$ . If we denote the inputs at  $s_1$  and  $s_2$  by  $\mathbf{a}$  and  $\mathbf{b}$ , respectively, and the output at  $t$  by  $A_t \mathbf{a} + B_t \mathbf{b}$ , then the requirement on  $\mathbf{c}_2$  is that  $(A_t + B_t)$  has full column rank. Note that this condition on the end-to-end transfer function is exactly what is needed for a conventional linear network coding problem, where an information source, initially available at both  $s_1$  and  $s_2$ , is to be multicast to the destinations in  $T$ . To see this, consider the transfer function of a linear network coding solution for this conventional multicasting problem. Denote the inputs at  $s_1$  and  $s_2$  by  $\mathbf{a}$  and  $\mathbf{b}$ , respectively, and the output at  $t$  by  $A'_t \mathbf{a} + B'_t \mathbf{b}$ . Since for this problem we are inputting  $\mathbf{a} = \mathbf{b}$  at both  $s_1$  and  $s_2$ , the output at  $t$  is  $(A'_t + B'_t) \mathbf{a}$ . In order to recover  $\mathbf{a}$ ,  $(A'_t + B'_t)$  needs to have full column rank. Thus the two problems have the same requirement in terms of the transfer function. As a result, the rate regions are the same. Consequently, the problem boils down to characterizing the rate region for the problem of multicasting a common source from  $s_1$  and  $s_2$  to  $T$  at rate  $I(X; Y)$ . Introduce a virtual source node  $s$  that has infinite capacity links to  $s_1$  and  $s_2$ . The corresponding rate region is the set of  $\mathbf{c}_2$  that satisfies the following system of linear constraints together with variable  $\mathbf{f}_t^{(3)}$ :

$$\mathbf{c}_2 \geq \mathbf{f}_t^{(3)}, \quad \forall t \in T \quad (22)$$

$$\mathbf{f}_t^{(3)} \in \mathcal{F}_{s,t}(I(X; Y)), \quad \forall t \in T. \quad (23)$$

This scheme can be generalized by reallocating a portion of bitrate from the second session (using  $\mathbf{c}_2$ ) to the first session (using  $\mathbf{c}_1$ ). Specifically, for any  $r_1, r_2, r_3 \geq 0$  satisfying  $I(X; Y) = r_1 + r_2 + r_3$ , we can use  $\mathbf{c}_1$  to provide each destination a flow of rate  $H(X|Y) + r_1$  from  $s_1$  and a flow of rate  $H(Y|X) + r_2$  from  $s_2$ . We use  $\mathbf{c}_2$  to provide each destination a flow of rate  $r_3$  from the virtual node  $s$ . Then the admissible rate region of the proposed scheme, denoted by  $\mathcal{A}_1$ , is the set of  $\mathbf{c}$  satisfying the following inequalities:

$$\mathbf{c} \geq \mathbf{c}_1 + \mathbf{c}_2 \quad (24)$$

$$\mathbf{c}_1 \geq \mathbf{f}_t^{(1)} + \mathbf{f}_t^{(2)}, \quad \forall t \in T \quad (25)$$

$$\mathbf{c}_2 \geq \mathbf{f}_t^{(3)}, \quad \forall t \in T \quad (26)$$

where

$$\mathbf{f}_t^{(1)} \in \mathcal{F}_{s_1, t}(H(X|Y) + r_1), \quad \forall t \in T \quad (27)$$

$$\mathbf{f}_t^{(2)} \in \mathcal{F}_{s_2, t}(H(Y|X) + r_2), \quad \forall t \in T \quad (28)$$

$$\mathbf{f}_t^{(3)} \in \mathcal{F}_{s, t}(r_3), \quad \forall t \in T \quad (29)$$

$$r_1 + r_2 + r_3 = I(X; Y) \quad (30)$$

$$r_1, r_2, r_3 \geq 0. \quad (31)$$

### B. Comparison With the Optimal Region

Recall from the introduction that the optimal admissible rate region for communicating correlated sources over a noiseless network to multiple destinations is characterized by three sets of cut conditions [14], [15]. That is, a vector  $\mathbf{c}$  is admissible if and only if:

- i) each cut separating  $s_1$  from any destination node has at least capacity  $H(X|Y)$ ;
- ii) each cut separating  $s_2$  from any destination node has at least capacity  $H(Y|X)$ ;
- iii) each cut separating  $s_1$  and  $s_2$  from any destination node has at least capacity  $H(X, Y)$ .

We can turn these cut conditions into equivalent flow conditions. The optimal admissible rate region, denoted by  $\mathcal{A}_{\text{optimal}}$ , comprises the set of  $\mathbf{c}$  satisfying the following system of linear inequalities:

$$\mathbf{c} \geq \mathbf{f}_t^{(1)}, \quad \forall t \in T \quad (32)$$

$$\mathbf{c} \geq \mathbf{f}_t^{(2)}, \quad \forall t \in T \quad (33)$$

$$\mathbf{c} \geq \mathbf{f}_t^{(3)}, \quad \forall t \in T \quad (34)$$

where

$$\mathbf{f}_t^{(1)} \in \mathcal{F}_{s_1, t}(H(X|Y)), \quad \forall t \in T \quad (35)$$

$$\mathbf{f}_t^{(2)} \in \mathcal{F}_{s_2, t}(H(Y|X)), \quad \forall t \in T \quad (36)$$

$$\mathbf{f}_t^{(3)} \in \mathcal{F}_{s, t}(H(X, Y)), \quad \forall t \in T. \quad (37)$$

Here,  $s$  is a virtual source node that has a link to  $s_1$  and a link to  $s_2$ , both with infinite capacity.

We now present an example that shows that the proposed scheme is in general suboptimal. Consider Fig. 4 with  $H(X) = H(Y) = 1$ ,  $H(W) = 0.5$ . Each link has the same capacity, 0.5. This example network satisfies all three cut conditions; hence it is theoretically admissible. However, it does not reside in  $\mathcal{A}_1$ , the admissible rate region of the proposed low-complexity scheme. In fact, by solving a linear program, it can be shown

that the proposed scheme cannot fulfill the demand in  $\mathbf{c} = 0.5\mathbf{1}$  but can fulfill the demand in  $\frac{7}{12}\mathbf{1}$ ; here  $\mathbf{1}$  is a length- $|E|$  vector consisting of all ones. Specifically, the linear program is

$$\begin{aligned} & \min \alpha \\ & \text{subject to : } (24)\text{--}(31), \quad \text{and } \mathbf{c} = \alpha\mathbf{1}. \end{aligned}$$

The linear program looks for the minimum  $\alpha$  such that the demand can be met in  $\alpha\mathbf{1}$ .

### C. Comparison With a Separation Approach

For the problem of communicating correlated sources from multiple sources to a single destination, Han [20] showed that it is optimal to separately treat the compression aspect and the network information relay part. More specifically, the sources can perform Slepian-Wolf coding to form a set of representations whose rates fall into the Slepian-Wolf region. Then the network serves the purpose of supplying multiple paths for streaming the Slepian-Wolf coded symbols to the destination.

Generalizing this, recently, Ramamoorthy *et al.* [21], [17] investigated a *separation approach*, where Slepian-Wolf coding is performed at the sources, and network coding is used only to stream the Slepian-Wolf coded bits while ignoring the correlation between the sources. Specifically, each source  $s_i$  encodes the source into a set of coded bits, say  $U_i$ . The network is used to provide a transfer function such that each destination  $t$  receives a subset  $E_i^t \subseteq U_i$ , for each  $i$ . Then two separate subproblems are treated as follows:

- i) design Slepian-Wolf codes such that for each receiver  $t$ , its received bits  $\{E_i^t, \forall i\}$  can be used to recover the sources with low complexity;
- ii) design network codes to implement the needed transfer function, i.e., let destination  $t$  receive the bits  $\{E_i^t, \forall i\}$ , with low complexity; this is done while ignoring the correlation among the coded bits  $U_i$ .

Problem i) can be viewed as a Slepian-Wolf coding problem. In theory, using random codes, decoding can succeed at a receiver with high probability as long as the rates at which coded bits are received from the sources fall in the interior of the Slepian-Wolf admissible region. Problem ii) can be viewed as a multi-session network coding problem, which is still an open problem in general. Ramamoorthy *et al.* [21] investigated a suboptimal treatment to this problem, where the network capacity is partitioned into a collection of  $(2^{N_S} - 1)(2^{N_T} - 1)$  multicast sessions. Here  $N_S$  and  $N_T$  are the number of sources and destinations, respectively. Each session streams certain bits from a subset of sources to a subset of destinations. The partitioning can be optimized via a linear program.

The separation approach can also potentially lead to a low complexity scheme, if the aforementioned linear optimization is carried over a small collection of multicast sessions. For problems with a small  $(2^{N_S} - 1)(2^{N_T} - 1)$  value, this is manageable. For larger problems, we may need to select a small collection of multicast sessions from the exponential set with  $(2^{N_S} - 1)(2^{N_T} - 1)$  sessions. One advantage of the separation approach, however, is that it can work for arbitrary correlation, whereas the proposed approach leverages the linearity in the syndrome coding for BSC correlation.

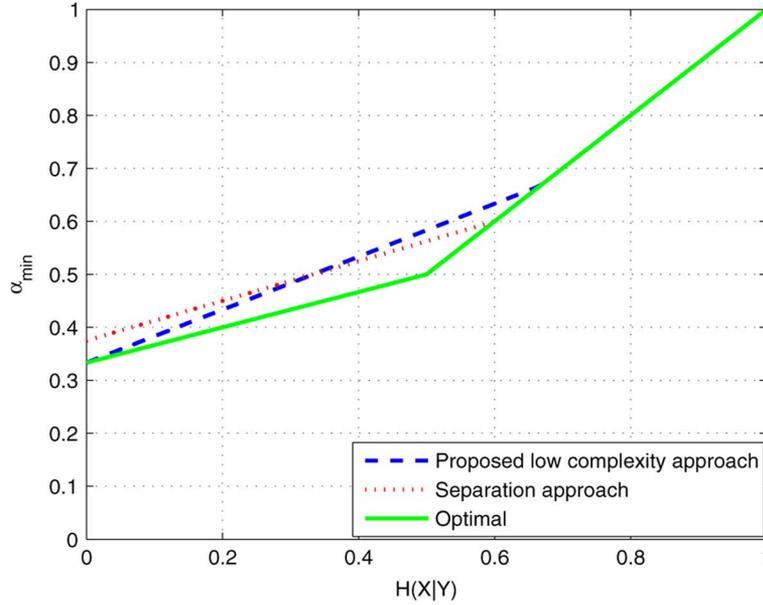


Fig. 8. Performance comparison among the proposed low complexity approach, the separation approach, and the optimal solution, using the network shown in Fig. 4.

Ramamoorthy *et al.* [17], [21] investigated the optimality of the separation approach, assuming linear network coding is used for problem ii) above. It was shown in [17] that such a separation approach is optimal when there are two sources and two destinations. This is done with a graph theoretic path-packing procedure, which shows that any admissible network with capacity  $\mathbf{c}$  can be decomposed into

$$\mathbf{c} \geq \mathbf{f}^{(1)} + \mathbf{f}^{(2)} + \max\{\mathbf{f}^{(3)} + \mathbf{f}^{(4)}, \mathbf{f}^{(5)} + \mathbf{f}^{(6)}\} \quad (38)$$

where

$$\mathbf{f}^{(1)} \in \mathcal{F}_{s_1, t_1}(R) \quad (39)$$

$$\mathbf{f}^{(2)} \in \mathcal{F}_{s_2, t_2}(R) \quad (40)$$

$$\mathbf{f}^{(3)} \in \mathcal{F}_{s_1, t_1}(R_1) \quad (41)$$

$$\mathbf{f}^{(4)} \in \mathcal{F}_{s_2, t_1}(R_2) \quad (42)$$

$$\mathbf{f}^{(5)} \in \mathcal{F}_{s_1, t_2}(R_1) \quad (43)$$

$$\mathbf{f}^{(6)} \in \mathcal{F}_{s_2, t_2}(R_2) \quad (44)$$

$$R_1 \geq H(X|Y) \quad (45)$$

$$R_2 \geq H(Y|X) \quad (46)$$

$$R + R_1 + R_2 \geq H(X, Y) \quad (47)$$

$$R \geq 0. \quad (48)$$

The decomposition result (38) implies that for any admissible  $\mathbf{c}$ , we can apply Slepian-Wolf coding at the two source nodes and use the network to implement a transfer function that streams the needed coded bits to the destinations. Specifically, form a set of bits  $U_{11}$  with rate  $R$  and  $U_{12}$  with rate  $R_1$  at node  $s_1$ ; form  $U_{21}$  with rate  $R$  and  $U_{22}$  with rate  $R_2$  at node  $s_2$ . Then  $\mathbf{f}^{(1)}$  is used to route  $U_{11}$  to node  $t_1$  and  $\mathbf{f}^{(2)}$  is used to route  $U_{21}$  to node  $t_2$ . The resource  $\max\{\mathbf{f}^{(3)} + \mathbf{f}^{(4)}, \mathbf{f}^{(5)} + \mathbf{f}^{(6)}\}$  can be used to multicast  $U_{12}$  and  $U_{22}$  to  $\{t_1, t_2\}$  using network coding. Thus, in this case, the separation approach provides a low complexity scheme, assuming there is a low complexity Slepian-Wolf coding scheme

that enables the sources to be recovered from the bits received by each decoder.

This decomposition result also implies the optimality of the low-complexity scheme proposed in Section III. This is because the conditions in (24)–(31) can be satisfied upon setting  $\mathbf{c}_1 = \max\{\mathbf{f}^{(3)} + \mathbf{f}^{(4)}, \mathbf{f}^{(5)} + \mathbf{f}^{(6)}\}$  and  $\mathbf{c}_2 = \mathbf{f}^{(1)} + \mathbf{f}^{(2)}$ .

The optimality proofs for one and two destinations do not generalize to more sources or more destinations. Ramamoorthy *et al.* [17] presented examples showing the separation of Slepian-Wolf coding and linear network coding is not optimal in general. The example of Fig. 4 with  $H(X) = H(Y) = 1$  and  $H(W) = 0.5$  was used in [17] to show the suboptimality of the separation approach. As discussed in the previous subsection, this example in fact also shows that the low-complexity approach proposed in Section III is suboptimal. For this example, a manually obtained low-complexity solution is given in Section II, demonstrating the potential of the proposed general design strategy.

Using the network shown in Fig. 4, we now compare the performance of the proposed low complexity approach with the separation approach and the optimal solution. As before, we assume two binary, uniform sources whose correlation follows a binary symmetric channel. In the evaluation, we vary  $H(X|Y)$  from 0 to 1. We assume each edge in Fig. 4 has the same capacity  $\alpha$ . For each value of  $H(X|Y)$ , we compute the minimum bandwidth scaling factor  $\alpha_{\min}$  needed to fulfill the traffic demand, for all three schemes. The results are shown in Fig. 8. From Fig. 4, it is seen that the proposed low complexity approach performs better than the separation approach in some cases, and worse in some other cases.

By evaluating the cut conditions characterizing the optimal solution (Section IV-B), we see that the optimal  $\alpha_{\min}$  has the following expression:

$$\alpha_{\min}^{\text{opt}} = \max \left\{ H(X|Y), \frac{1 + H(X|Y)}{3} \right\}. \quad (49)$$

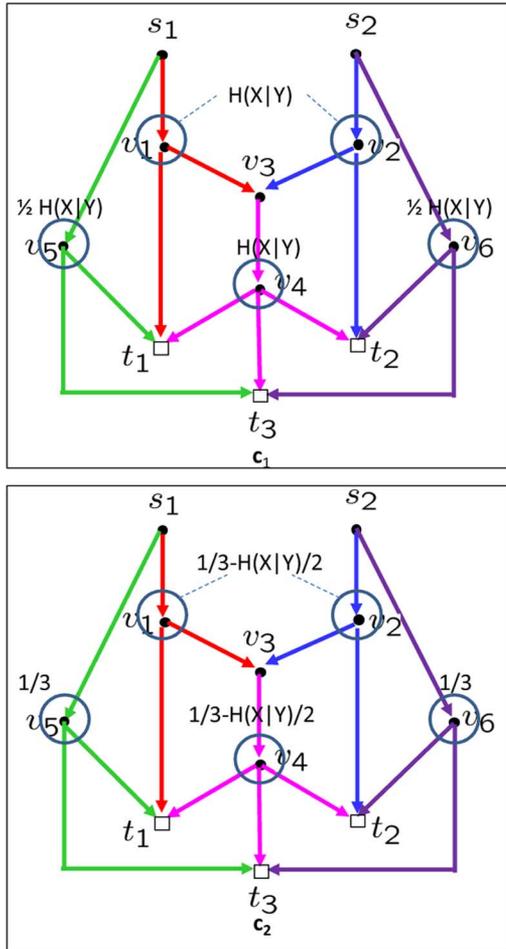


Fig. 9. Illustration of how the proposed low complexity approach achieves the first line segment  $\alpha_{\min} = 1/3 + H(X|Y)/2$  for  $H(X|Y) \in [0, 2/3]$ . The original network resources are partitioned into two parts:  $\mathbf{c} = \mathbf{c}_1 + \mathbf{c}_2$ .

The  $\alpha_{\min}$  with the separation approach appears to have the expression:

$$\alpha_{\min}^{\text{separation}} = \max\{H(X|Y), 0.375(1 + H(X|Y))\}. \quad (50)$$

The  $\alpha_{\min}$  with the proposed low complexity approach appears to have the expression:

$$\alpha_{\min}^{\text{proposed}} = \max\left\{H(X|Y), \frac{1}{3} + \frac{H(X|Y)}{2}\right\}. \quad (51)$$

To better understand these results, we now describe some details of the solutions, which we reconstructed with the help of the outputs of the linear program. For the proposed low complexity approach, the original network resources are partitioned into two parts:  $\mathbf{c} = \mathbf{c}_1 + \mathbf{c}_2$ ; these two parts are shown in Fig. 9. In Fig. 9, we label the capacities for five groups of edges (shown by the circles). Note that we can assume without loss of generality that edges touched by each circle have the same capacity (because we can simply forward whatever information the center node gets to downstream nodes). Thus to specify a solution for this problem, we only need to specify the capacity assignments on the five circles. For instance,  $\mathbf{c}_1$  in Fig. 9 can be specified as  $c_{s_1 v_1} = c_{s_2 v_2} = c_{v_3 v_4} = H(X|Y)$  and

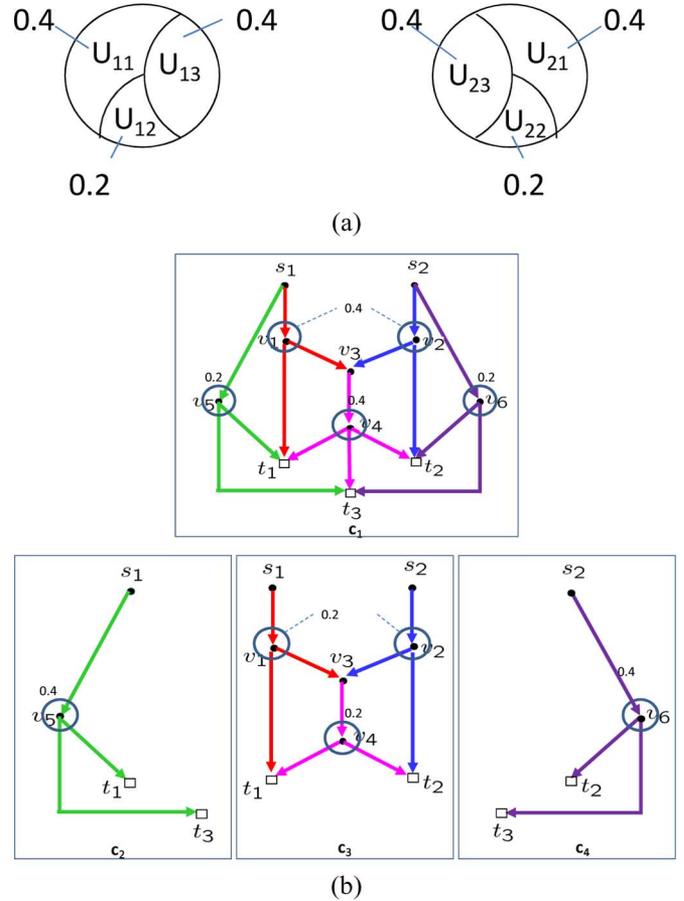


Fig. 10. Illustration of how the separation approach achieves the point  $\alpha_{\min} = 0.6$  with  $H(X|Y) = 0.6$ . (a) Source  $s_1$  forms three sets of coded bits  $U_{11}, U_{12}, U_{13}$ , whose sizes are  $0.4, 0.2, 0.4$ , respectively; source  $s_2$  forms three sets of coded bits  $U_{21}, U_{22}, U_{23}$ , whose sizes are  $0.4, 0.2, 0.4$ , respectively. (b) The network is partitioned into four parts,  $\mathbf{c} = \mathbf{c}_1 + \mathbf{c}_2 + \mathbf{c}_3 + \mathbf{c}_4$ .  $\mathbf{c}_1$  is used to send  $\{U_{11}, U_{21}\}$  to  $\{t_1, t_2, t_3\}$ .  $\mathbf{c}_2$  is used to send  $U_{13}$  to  $t_1$  and  $t_3$ .  $\mathbf{c}_3$  is used to send  $\{U_{12}, U_{22}\}$  to  $t_1$  and  $t_2$ .  $\mathbf{c}_4$  is used to send  $U_{23}$  to  $t_2$  and  $t_3$ .

$c_{s_1 v_5} = c_{s_2 v_6} = \frac{1}{2}H(X|Y)$ . From Fig. 9, it can be verified that i) the maximum edge capacity in the sum of these two graphs is  $1/3 + H(X|Y)/2$ , ii) in  $\mathbf{c}_1$ , each receiver can receive information at rate  $H(X|Y)$  from  $s_1$  and  $s_2$  (cf. (19)–(21)), and iii) in  $\mathbf{c}_2$ , each receiver has a flow of  $1 - H(X|Y) = I(X;Y)$  from the set  $\{s_1, s_2\}$  (cf. (22)–(23)). This shows how the proposed low complexity approach achieves the first line segment  $\alpha_{\min} = 1/3 + H(X|Y)/2$  for  $H(X|Y) \in [0, 2/3]$ . To achieve the second line segment,  $\alpha_{\min} = H(X|Y)$  for  $H(X|Y) \in [2/3, 1]$ , the proposed approach uses the same  $\mathbf{c}_1$  as in Fig. 9(a) but with  $\mathbf{c}_2$  having  $c_{s_1 v_5} = c_{s_2 v_6} = 1 - H(X|Y)$  and  $c_{s_1 v_1} = c_{s_2 v_2} = c_{v_3 v_4} = 0$ .

Why can the separation approach achieve the point  $\alpha_{\min} = 0.6$  when  $H(X|Y) = 0.6$  whereas the proposed scheme cannot? To see this, we now explain how the separation approach achieves this point. As illustrated by Fig. 10(a), source  $s_1$  forms three sets of coded bits  $U_{11}, U_{12}, U_{13}$ ; source  $s_2$  forms three sets of coded bits  $U_{21}, U_{22}, U_{23}$ . As illustrated by Fig. 10(b), the separation solution partitions the network into four parts,  $\mathbf{c} = \mathbf{c}_1 + \mathbf{c}_2 + \mathbf{c}_3 + \mathbf{c}_4$ .  $\mathbf{c}_1$  is used to send  $\{U_{11}, U_{21}\}$  to  $\{t_1, t_2, t_3\}$ . The structure of  $\mathbf{c}_1$  in Fig. 10(b) is similar to  $\mathbf{c}_1$  in Fig. 9. However,  $\mathbf{c}_1$  in Fig. 10(b) is only used

to transfer information at rate  $0.4 < H(X|Y)$  to  $\{t_1, t_2, t_3\}$ . A crucial difference is that whereas the proposed approach always communicates information to  $\{t_1, t_2, t_3\}$  together, the separation approach uses  $\mathbf{c}_2$  to send  $U_{13}$  to  $t_1$  and  $t_3$ ,  $\mathbf{c}_3$  to send  $\{U_{12}, U_{22}\}$  to  $t_1$  and  $t_2$ , and  $\mathbf{c}_4$  to send  $U_{23}$  to  $t_2$  and  $t_3$ . Since  $U_{12}$  and  $U_{22}$  are available only at  $\{t_1, t_2\}$  but not at  $t_3$ , the proposed scheme cannot be used here.

## V. CONCLUSION

We considered the problem of communicating correlated information from two source nodes over a network of noiseless channels to multiple destination nodes. We presented a design strategy that potentially admits low-complexity implementation for binary sources correlated in such a way that one source is the output of a binary symmetric channel whose input is the other source. The strategy is to convey to each destination node a linearly compressed version of the correlation noise (that is, the binary symmetric channel noise), and just enough additional linear descriptions of the sources. Decoding is done by first decompressing the correlation noise and then solving a system of linear equations. The remaining problem is to fulfill the end-to-end traffic demand prescribed by the strategy using certain network resources. Such a demand formulation is unique since it involves *computation*, in addition to communication aspects. This opens up new research challenges, which we call “network coding for distributed computations”. We gave a simple (suboptimal) approach to this problem and constructed a practical communication scheme. The admissible rate region of the proposed scheme is characterized via a system of linear inequalities.

Our proposed strategy can potentially be generalized to settings with more than two sources, more general correlation structures, and lossy compression of correlated continuous sources. This can be done by first trying to redefine the unknowns with a set of independent random processes, some compressible and some not, and then using the network to (i) convey to each destination (linearly) compressed versions of the compressible random processes with low complexity and (ii) present sufficient additional descriptions (e.g., random linear descriptions) of the unknowns to the destinations.

## ACKNOWLEDGMENT

The authors would like to thank Angelos Liveris and Chih-Chun Wang for the helpful discussions with them, and the reviewers and the associate editor for their comments.

## REFERENCES

- [1] Y. Wu, V. Stanković, Z. Xiong, and S.-Y. Kung, “On practical design for joint distributed source and network coding,” in *Proc. 1st Workshop Netw. Coding, Theory, Applicat.*, Riva del Garda, Italy, Apr. 2005.
- [2] D. Slepian and J. K. Wolf, “Noiseless coding of correlated information sources,” *IEEE Trans. Inf. Theory*, vol. 19, pp. 471–480, July 1973.
- [3] A. D. Wyner, “Recent results in the Shannon theory,” *IEEE Trans. Inf. Theory*, vol. 20, pp. 2–10, Jan. 1974.
- [4] Z. Xiong, A. Liveris, and S. Cheng, “Distributed source coding for sensor networks,” *IEEE Signal Processing Mag.*, vol. 21, pp. 80–94, Sept. 2004.
- [5] S. Pradhan and K. Ramchandran, “Generalized coset codes for distributed binning,” *IEEE Trans. Inf. Theory*, vol. 51, no. 10, pp. 3457–3474, Oct. 2005.
- [6] V. Stanković, A. D. Liveris, Z. Xiong, and C. N. Georghiades, “On code design for the general slepian-wolf problem and for lossless multiterminal communication networks,” *IEEE Trans. Inf. Theory*, vol. 52, pp. 1495–1507, Apr. 2006.
- [7] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, “Network information flow,” *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, July 2000.
- [8] P. A. Chou and Y. Wu, “Network coding for the internet and wireless networks,” *IEEE Signal Processing Mag.*, vol. 24, no. 5, pp. 77–85, 2007.
- [9] T. Ho, M. Médard, R. Koetter, D. R. Karger, M. Effros, J. Shi, and B. Leong, “A random linear network coding approach to multicast,” *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4413–4430, Oct. 2006.
- [10] P. Sander, S. Egner, and L. Tolhuizen, “Polynomial time algorithms for network information flow,” in *Proc. Symp. Parallel Algorithms Architect. (SPAA)*, San Diego, CA, Jun. 2003, pp. 286–294, ACM.
- [11] P. A. Chou, Y. Wu, and K. Jain, “Practical network coding,” in *Proc. 41st Allerton Conf. Comm., Ctrl. and Comp.*, Monticello, IL, Oct. 2003.
- [12] Y. Wu, “A trellis connectivity analysis of random linear network coding with buffering,” in *Proc. IEEE Int. Symp. Inf. Theory*, Seattle, WA, Jul. 2006.
- [13] D. S. Lun, M. Médard, R. Koetter, and M. Effros, “Further results on coding for reliable communication over packet networks,” in *Proc. IEEE Int. Symp. Inf. Theory*, Sep. 2005, IEEE.
- [14] L. Song and R. W. Yeung, “Network information flow—Multiple sources,” in *Proc. Int’l Symp. Information Theory*, Jun. 2001.
- [15] T. Ho, M. Médard, M. Effros, and R. Koetter, “Network coding for correlated sources,” in *Proc. Conf. Inf. Sci. Syst. (CISS)*, Princeton, NJ, Mar. 2004.
- [16] I. Csiszár, “Linear codes for sources and source networks: Error exponents, universal coding,” *IEEE Trans. Inf. Theory*, Nov. 1982.
- [17] A. Ramamoorthy, K. Jain, P. A. Chou, and M. Effros, “Separating distributed source coding from network coding,” *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2785–2794, Jun. 2006.
- [18] P. Elias, “Coding for noisy channels,” *IRE Conv. Rec.*, pp. 37–46, 1955.
- [19] T. C. Anчета, Jr., “Syndrome-source-coding and its universal generalization,” *IEEE Trans. Inf. Theory*, vol. IT-22, no. 4, pp. 432–436, Jul. 1976.
- [20] T. S. Han, “Slepian-wolf-cover theorems for networks of channels,” *Inf. Contr.*, vol. 47, pp. 67–83, 1980.
- [21] A. Ramamoorthy, K. Jain, P. A. Chou, and M. Effros, “Separating distributed source coding from network coding,” in *Proc. 42nd Allerton Conf. Commun., Contr. Computing*, Monticello, IL, Oct. 2004.

**Yunnan Wu** (S’02–M’06) received the Ph.D. degree from Princeton University in January 2006.

Since August 2005, he has been a Researcher at Microsoft Corporation (Redmond, WA, USA). His research interests include networking, graph theory, information theory, game theory, and wireless communications.

Dr. Wu was a recipient of the Best Student Paper Award at the 2000 SPIE and IS&T Visual Communication and Image Processing Conference, and a recipient of the Student Paper Award at the 2005 IEEE International Conference on Acoustics, Speech, and Signal Processing. He was awarded a Microsoft Research Graduate Fellowship for 2003–2005.

**Vladimir Stanković** (M’04) received the Dipl.-Ing. degree in electrical engineering from the University of Belgrade, Serbia, in 2000, and the Dr.-Ing. degree from the University of Leipzig, Germany, in 2003.

From June 2003 to February 2006, he was with the Department of Electrical and Computer Engineering at Texas A&M University, College Station. From 2006, he was with the Lancaster University, U.K., as a Lecturer. In October 2007, he joined the Department of Electronic and Electrical Engineering, at the University of Strathclyde, Glasgow, where he is currently a Lecturer. He has published three book chapters and over 70 papers in international journals and conference proceedings. His research focuses on image/signal processing, network information theory, wireless communications, and wireless ad hoc/sensor networks.

Dr. Stanković serves as Associate Editor of IEEE COMMUNICATIONS LETTERS.

**Zixiang Xiong** (S'91–M'96–SM'02–F'07) received the Ph.D. degree in electrical engineering in 1996 from the University of Illinois at Urbana-Champaign.

From 1995 to 1997, he was with Princeton University, first as a visiting student, then as a research associate. From 1997 to 1999, he was with the University of Hawaii. Since 1999, he has been with the Department of Electrical and Computer Engineering at Texas A&M University, where he is a Professor. His research interests are network information theory, code designs and applications, biomedical engineering, genomic signal processing, and networked multimedia.

Dr. Xiong received a National Science Foundation Career Award in 1999, an Army Research Office Young Investigator Award in 2000, and an Office of Naval Research Young Investigator Award in 2001. He also received the 2006 IEEE Signal Processing Magazine best paper award. He served as associate editor for the IEEE Trans. on Circuits and Systems for Video Technology (1999–2005), the IEEE Trans. on Image Processing (2002–2005), and the IEEE Trans. on Signal Processing (2002–2006). He is currently an associate editor for the IEEE Trans. on Communications and the IEEE Trans. on Systems, Man, and Cybernetics (part B). He was the publications chair of ICASSP'07 and the technical program committee co-chair of ITW'07.

**Sun-Yuan Kung** (F'88) is a Professor at Department of Electrical Engineering in Princeton University, Princeton, NJ. His research areas include VLSI array processors, system modeling and identification, neural networks, wireless communication, sensor array processing, multimedia signal processing, bioinformatic data mining and biometric authentication. He has authored and coauthored more than 400 technical publications and numerous textbooks including *VLSI and Modern Signal Processing* (Prentice-Hall, 1985), *VLSI Array Processors* (Prentice-Hall, 1988); *Digital Neural Networks* (Prentice-Hall, 1993); *Principal Component Neural Networks* (Wiley, 1996); and *Biometric Authentication: A Machine Learning Approach* (Prentice-Hall, 2004).

Dr. Kung was a founding member of several Technical Committees (TC) of the IEEE Signal Processing Society, and was appointed as the first Associate Editor in VLSI Area (1984) and later the first Associate Editor in Neural Network (1991) for the IEEE TRANSACTIONS ON SIGNAL PROCESSING. He served as a Member of the Board of Governors of the IEEE Signal Processing Society (1989–1991). Since 1990, he has been the Editor-In-Chief of the *Journal of VLSI Signal Processing Systems*. He was a recipient of IEEE Signal Processing Society's Technical Achievement Award for the contributions on "parallel processing and neural network algorithms for signal processing" (1992); a Distinguished Lecturer of IEEE Signal Processing Society (1994); a recipient of IEEE Signal Processing Society's Best Paper Award for his publication on principal component neural networks (1996); and a recipient of the IEEE Third Millennium Medal (2000).