On Practical Design for Joint Distributed Source and Network Coding

Yunnan Wu*, Vladimir Stanković†, Zixiang Xiong†, and Sun-Yuan Kung*
*Dept. of Electrical Engineering, Princeton University, Princeton, NJ, {yunnanwu,kung}@princeton.edu
†Dept. of Electrical Engineering, Texas A&M University, College Station, TX, {stankovi,zx}@ee.tamu.edu

***Abstract*— We consider the problem of communicating correlated information from two source nodes over a network to multiple destination nodes. This problem involves a joint consideration of distributed source coding (compression) and network coding (information relaying). Although the optimal admissible rate region was previously characterized, it was not yet clear how to design practical communication schemes with low complexity. This work provides a partial answer to this question. Specifically, we focus on the case where the two sources are related by a binary symmetric channel (BSC). We propose a general strategy that potentially allows low complexity implementation, and a specific practical design based on the general strategy. The proposed specific practical design is in general suboptimal; but its low complexity and robustness to network dynamics make it suitable for practical implementation.**

I. Introduction

Consider the problem of communicating correlated information from two source nodes over a network to multiple destination nodes (see e.g., Fig. 1). Specifically, assume that the network is modelled as a set of noiseless channels with rate constraints, 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 $e \in E$. Let $\{(X_i, Y_i)\}_{i=1}^{\infty}$ be a sequence of independent drawings of a pair of (dependent) discrete random variables (X, Y). Assume that two source nodes, s_1 and s_2, observe X and Y, respectively. Let $T \subseteq V$ be the set of destination nodes. Then, the *admissible rate region* is defined as the set of $\mathbf{c}$ that allows all destination nodes to (almost) perfectly reconstruct both X and Y.

This problem involves aspects of both distributed source coding [1] and information relaying [2]. We begin with the former and consider the special case when there is only one destination node $T = \{t\}$ and two links in the network: one from source node s_1 to t, $s_1t \in E$, and another from s_2 to t, $s_2t \in E$. For this problem, Slepian and Wolf [3] showed that the admissible rate region is the set of $\mathbf{c}$ satisfying: $c(s_1t) \geq H(X|Y)$, $c(s_2t) \geq H(Y|X)$, and $c(s_1t)+c(s_2t) \geq H(X,Y)$. Subsequently, Csiszár [4] showed the existence of *universal* (i.e., without knowing the correlation statistics) Slepian-Wolf coders with linear encoding that can achieve any interior point of the admission rate region.

We now turn to the aspect of information relaying and consider the special case when X and Y are independent, while the network topology and the number of destination nodes are arbitrary. This problem is addressed by recent advances on *network coding*, which refers to a scheme where each node in the network is allowed to generate output symbols by encoding (i.e., computing certain function of) the symbols it received. Network coding has been primarily considered under the setup of multicasting, where a source node s communicates common information to multiple destination nodes T. Ahlswede *et al.* [2] showed that the maximum achievable rate from s to T, i.e., the *multicast capacity*, is equal to the minimum capacity of cuts separating the source from a destination, and it can be achieved with network coding[1]. Note that the current setup, where each destination $t \in T$ is required to reconstruct the two independent sources X and Y, can be converted into an equivalent multicast problem by introducing a virtual source node s that generates both X and Y and has one edge to s_1 with rate constraint $H(X)$ and another edge to s_2 with rate constraint $H(Y)$.

Another special case, where X and Y are correlated, and there is only one destination $T = \{t\}$, was addressed by Han [5]. For this problem, a separate treatment of compression (i.e., Slepian-Wolf coding) and information relaying (i.e., network coding) turns out to be optimal; the network serves the purpose of supplying multiple paths to stream the Slepian-Wolf coded symbols to the destination t.

The admissible rate region for the general problem where correlated sources X and Y generated at nodes s_1 and s_2, respectively, are to be delivered to multiple destinations T over an arbitrary network, is found by Song and Yeung [6]. It was shown in [6] that a vector $\mathbf{c}$ is admissible if and only if: 1) each cut separating s_1 from any destination node has at least capacity $H(X|Y)$, 2) each cut separating s_2 from any destination node has at least capacity $H(Y|X)$, and 3) each cut separating s_1 and s_2 from any destination node has at least capacity $H(X, Y)$.

In general, to achieve an arbitrary rate point in the admissible rate region [6], Slepian-Wolf coding (SWC) and network coding need to be combined. Generalizing the results of [4], Ho *et al.* [7] derived the error exponents for a random coding scheme. 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 over $GF(2)$). At a destination, decoding is done with the minimum entropy decoder or a maximum a posteriori prob-

[1] For a partition of the nodes $V = V_L + V_R$, the cut refers to the edges going from V_L to V_R, and the capacity of the cut refers to the sum capacity of the constituent edges.

ability decoder as in [4]. The decoding complexity of either decoder generally grows exponentially in the block length and thus is not suitable for practical implementation. Recently, Ramamoorthy *et al.* [8] investigated a *separation approach*, where SWC is performed at the sources, and network coding is used only to stream the Slepian-Wolf coded bits. At a destination, decoding is regarded successful as long as the numbers of coded bits received from the sources fall in the Slepian-Wolf admissible region. They showed that separating SWC and network coding is optimal when there are two sources and two destinations, and presented examples showing the separation is not optimal in general.

Note that all the results discussed so far are theoretical. Regarding practical achievements, a lot has been done recently in providing efficient code designs for the Slepian-Wolf problem (see [1] and the references therein). Following the syndrome-based approach [9], turbo and LDPC codes were used in [10] to approach any point on the Slepian-Wolf bound.

Practical designs for network coding appeared in the literature as well. For example, in [11] a prototype system for practical network coding in packet networks, using distributed random linear network coding with buffering, was presented. The system achieves throughput close to the capacity with low delay, and is robust to network dynamics. Distributed random network coding was also investigated by Ho *et al.* [12].

With these existing techniques for practical SWC and network coding in hand, our goal is to provide *low-complexity practical* designs for the problem of transmitting two correlated discrete memoryless sources over a noiseless network to multiple destinations. This paper generalizes the work of practical SWC [10], [13] by considering an arbitrary noiseless network between the sources and destinations. It also generalizes practical network coding designs [11], [12] in terms of handling correlated sources.

We focus on the case where the two sources are related by a binary symmetric channel (BSC). In this case, the *correlation noise*, i.e., the difference of the two sources $\boldsymbol{w} = \boldsymbol{y} - \boldsymbol{x}$, accounts for the redundancy in the sources. Due to the close relation between linear source coding and linear channel coding, the parity check matrix $\boldsymbol{H}$ of a good linear channel code (e.g., an LDPC code) for the BSC with noise distribution P_W serves as a good linear source code for the correlation noise, which is a memoryless source with distribution P_W. Moreover, decompression for this linear source code is equivalent to the syndrome decoding of the linear channel code.

In Section III, we present a general design strategy that potentially admits low complexity implementation. The strategy is to convey to each destination node a linearly compressed version of the correlation noise ($\boldsymbol{wH} = \boldsymbol{yH} - \boldsymbol{xH}$), and just enough additional (linear) descriptions of the sources. Decoding is done by first decompressing the correlation noise $\boldsymbol{w}$ from $\boldsymbol{wH}$ and then solving a system of (linear) equations. We then consider an example that was given in [8] to show that separating Slepian-Wolf coding and network coding can be suboptimal. Specifically, while this example is solvable according to the theory, it cannot be solved by separately performing Slepian-Wolf coding and network coding. We give a low-complexity solution for this example based on our proposed strategy. This shows that combining Slepian-Wolf coding and network coding do not necessarily come with a price of high complexity.

To construct a concrete communication scheme based on this general design strategy, we need to address a problem that can be regarded as a network coding one – fulfill the end-to-end 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 refer to as "network coding for distributed computations". We do not have a thorough answer for these challenges. Nonetheless, based on a simple (suboptimal) approach, we construct a practical communication scheme that performs joint Slepian-Wolf coding and network coding. This is presented in details in Section III; an overview is given below.

The available bit-rate resources in the network are partitioned into two disjoint parts. The first part is used to stream certain structured linear combinations ($\boldsymbol{xH}$ and $\boldsymbol{yH}$) from the two sources to the destinations in the manner of conventional network coding. Note that this is just one out of the many possible ways of conveying to each destination $\boldsymbol{wH} = \boldsymbol{yH} - \boldsymbol{xH}$. In the second part, the use of network coding departs from its previous form in that each destination observes less linear equations than the unknowns, which are correlated. In other words, network coding in the second part is used only to provide sufficient *descriptions* of the source data. The admissible rate region of this scheme is characterized by a linear system of inequalities. The proposed practical design is in general suboptimal; but it has low complexity and is robust to network dynamics.

II. Background

First, a word about notation. Random variables are denoted by capital letters, e.g., X, Y. Realizations of random vectors are denoted by bold-face lower-case letters, e.g., $\boldsymbol{x}$, $\boldsymbol{y}$. Matrices are denoted by bold-face upper-case letters; the size of a matrix will often be given as a subscript; $\boldsymbol{I}_k$ and $\boldsymbol{0}_{k_1 \times k_2}$ are $k \times k$ identity matrix and $k_1 \times k_2$ all-zero matrix, respectively.

A. Linear Channel Coding for Binary Symmetric Channel

Consider the memoryless binary symmetric channel (BSC) $Y = X + W$, where the noise process $\{W_i\}$ is i.i.d. with cross-over probability $\Pr(W = 1) < 0.5$ and independent of the input process $\{X_i\}$.

For this channel, a linear (n, k) channel code $\mathcal{C}$ can be specified by its generator matrix $\boldsymbol{G}_{k\times n}$. Denote the parity-check matrix for this code by $\boldsymbol{H}_{n\times(n-k)}$, which satisfies $\boldsymbol{GH} = \boldsymbol{0}$. Let ϕ_x denote a decoding function that estimates the transmitted codeword from the channel output $\boldsymbol{y}_{1\times n}$. The probability of reconstruction error is then

$$P_e(\boldsymbol{H}, \phi_x) \equiv \Pr\left[\phi_x(\boldsymbol{y}) \neq \boldsymbol{x}\right]. \quad (1)$$

In one possible implementation, the decoding function is $\phi_x(\boldsymbol{y}) = \boldsymbol{y} - \phi_w(\boldsymbol{yH})$, making use of a function ϕ_w that estimates the noise realization $\boldsymbol{w}$ from the *syndrome* $\boldsymbol{yH} = \boldsymbol{wH}$. With this type of decoder, the probability of error is

$$P_e(\boldsymbol{H}, \phi_x) = P'_e(\boldsymbol{H}, \phi_w) \equiv \Pr\left[\phi_w(\boldsymbol{wH}) \neq \boldsymbol{w}\right]. \quad (2)$$

Assuming each codeword in $\mathcal{C}$ is used as input with equal probability, minimum probability of error decoding amounts to finding a codeword $\boldsymbol{x} \in \mathcal{C}$ that has minimum Hamming distance with $\boldsymbol{y}$. Minimum probability of error decoding can also be accomplished by a syndrome-based decoder $\phi_w(\boldsymbol{s})$ that returns the minimum weight $\boldsymbol{w}$ satisfying $\boldsymbol{s} = \boldsymbol{wH}$.

Elias [14] showed that for sufficiently large n, there exist $\boldsymbol{H}$ and ϕ_x (or ϕ_w), such that the code rate k/n approaches the capacity $1 - H(W)$, and $P_e(\boldsymbol{H}, \phi_x)$ is arbitrarily small.

In terms of practical implementation, LDPC codes are linear block codes that can provide near-capacity performance with low encoding/decoding complexity. There are known iterative algorithms that perform approximately minimum-distance decoding for LDPC codes. For LDPC codes, the sum-product algorithm can also be used to provide an efficient syndrome decoder ϕ_w (see e.g., [15]).

B. Syndrome-based Slepian-Wolf Coding

Slepian and Wolf [3] showed that to compress separately two dependent discrete memoryless sources $\{(X_i, Y_i)\}_{i=1}^{\infty}$, we only need to describe X using $H(X|Y) + r$ bits and Y using $H(Y|X) + (I(X;Y) - r)$ bits, where $0 \leq r \leq I(X;Y)$.

The problem of approaching the rate point corresponding to $r = 0$, now called "asymmetric SWC", is equivalent to a problem of compressing X with side information Y at the decoder. For this problem, according to Wyner [9], Slepian and Wolf gave a constructive scheme for the case where X and Y are uniform and the correlation between X and Y is modeled by the BSC $Y = X + W$. The scheme is as follows. The source node encodes $\boldsymbol{x}$ into $\boldsymbol{xH}$ (of length $(n-k)$). The decoder observes $\boldsymbol{xH}$ and the side information $\boldsymbol{y}$. Assuming a syndrome decoding function ϕ_w, the reconstruction of $\boldsymbol{x}$ is $\hat{\boldsymbol{x}} = \boldsymbol{y} - \phi_w(\boldsymbol{yH} - \boldsymbol{xH})$ and

$$\Pr\left[\hat{\boldsymbol{x}} \neq \boldsymbol{x}\right] = \Pr\left[\phi_w(\boldsymbol{wH}) \neq \boldsymbol{w}\right]. \quad (3)$$

Recently, the syndrome-based scheme in [9] was extended to "nonasymmetric SWC" (enables approaching any rate point on the Slepian-Wolf bound, i.e., $0 \leq r \leq I(X;Y)$) by Pradhan and Ramchandran [13]. The idea underlying the method of [13] is to partition a single linear channel code into two nonoverlapping subcodes. Based on this framework, previous work [10] proposed a practical code design for an arbitrary number of sources and more general distributions and correlations among the sources.

C. Linear Source Coding

Slepian and Wolf's constructive scheme revealed the close relation between linear channel coding and linear source coding. This relation and Elias's result [14] together imply that for an i.i.d. binary source $\{W_i\}$, there exist linear source codes with rate approaching the entropy and probability of error approaching 0. A good linear source code for i.i.d. source $\{W_i\}$ with distribution P_W can thus 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 decoding for the corresponding linear channel code.

III. A General Design Strategy

In this paper we focus on the case where $\{X_i\}$ and $\{Y_i\}$ are binary memoryless sources related by the BSC, $Y = X + W$, with $\Pr(W = 1) < 0.5$.

The length-n source realizations $\boldsymbol{x}$ and $\boldsymbol{y}$ are the unknowns that need to be reconstructed at each destination. Since $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{w}$, we can equivalently treat $\boldsymbol{x}$ and $\boldsymbol{w}$ as the unknowns in the system. Note that $\boldsymbol{x}$ is uniform, $\boldsymbol{w}$ is nonuniform (hence compressible), and $\boldsymbol{x}$ is independent from $\boldsymbol{w}$.

Let $\boldsymbol{H}_{n \times (n-k)}$ be the parity check matrix of a good channel code for the BSC with noise distribution P_W. The rate of the code is $k/n \approx 1 - H(W)$. Assume there is a good syndrome decoder ϕ_w that can perform near-optimal decoding with low complexity. Due to the close relation between linear source coding and linear channel coding, $\boldsymbol{wH}$ serves as a compressed version of $\boldsymbol{w}$, which can be decompressed with low complexity.

Our proposed design strategy is to use the network to:

1) convey to each destination $\boldsymbol{wH} = \boldsymbol{yH} - \boldsymbol{xH}$, with low complexity;
2) present to each destination $t \in T$ sufficient descriptions $\varphi_t(\boldsymbol{x}, \boldsymbol{y})$, which together with $\boldsymbol{w}$ allow $\boldsymbol{x}$ to be reconstructed.

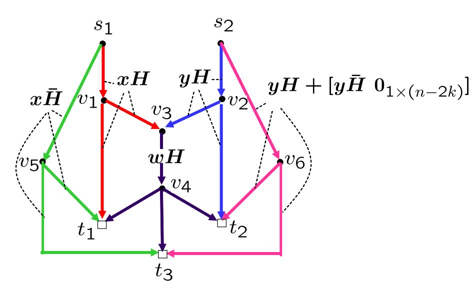

Fig. 1. An example network $G = (V, E, \mathbf{c})$ given in [8]. For the case where $H(X) = H(Y) = 1$, $H(W) = 0.5$, and the rate constraint on each edge is 0.5, we give a solution, shown by labelling the information flowing on each edge.

We now illustrate this strategy with an example. Fig. 1 is an example network that was given in [8] to show that separating SWC and network coding is in general suboptimal. For the case where $H(X) = H(Y) = 1$, $H(W) = 0.5$, and the rate constraint on each edge is 0.5, the example network in Fig. 1 is not admissible if SWC and network coding are separated. We now show that it is admissible with the proposed design strategy. A solution is given in Fig. 1, where the data to be carried on each edge is labelled. Let $\bar{\boldsymbol{H}}$ be an $n \times k$ matrix such that $[\bar{\boldsymbol{H}}\ \boldsymbol{H}]$ is invertible. Assume $k/n \leq 1/2$ and $k/n \approx 1/2$. All destinations receive $\boldsymbol{wH}$ and thus can recover $\boldsymbol{w}$ with high probability. Once $\boldsymbol{w}$ is recovered, it can be easily verified that all destinations can recover the sources $(\boldsymbol{x}, \boldsymbol{y})$. In this example, decoding at each destination node is done by 1) decompressing

$\boldsymbol{w}$ from $\boldsymbol{wH}$, and 2) solving a system of linear equations to recover $\boldsymbol{x}$. Both steps can be done with low complexity.

Based on the proposed strategy, to arrive at a concrete communication scheme as given in Fig. 1, we need to address how to use the available network resources to fulfill the end-to-end traffic demand prescribed by the proposed strategy. This can be regarded as a network coding problem. Whereas previously network coding was primarily examined 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* $\boldsymbol{yH}-\boldsymbol{xH}$, where $\boldsymbol{xH}$ and $\boldsymbol{yH}$ 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 coding for distributed computation): *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 fulfilling this demand?*

There are many ways for conveying $\{P_i+Q_i\}$ to the destinations. For example, as in Fig. 1, 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, in the next section we will take a simple (suboptimal) treatment of Problem 1 and present a specific practical design.

IV. A Practical Design Scheme

Let $\bar{\boldsymbol{H}}$ be an $n\times k$ matrix such that $[\bar{\boldsymbol{H}}\ \boldsymbol{H}]$ is invertible. If $\boldsymbol{H}$ is in systematic form, i.e., $\boldsymbol{H}=[\boldsymbol{P}^T\ \boldsymbol{I}]^T$, $\bar{\boldsymbol{H}}$ can be chosen as $[\boldsymbol{I}\ \boldsymbol{0}]^T$. Define

$$\tilde{\boldsymbol{x}}\equiv\boldsymbol{x}[\bar{\boldsymbol{H}}\ \boldsymbol{H}];\quad \boldsymbol{x}_1\equiv\boldsymbol{x}\bar{\boldsymbol{H}};\tag{4}$$

$$\boldsymbol{y}\equiv\boldsymbol{y}[\bar{\boldsymbol{H}}\ \boldsymbol{H}];\quad \boldsymbol{y}_1\equiv\boldsymbol{y}\bar{\boldsymbol{H}}.\tag{5}$$

In this section we consider the following specific (suboptimal) way of implementing the general design strategy.

1) Convey to each destination $\boldsymbol{xH}$ and $\boldsymbol{yH}$, which enables the destination to compute $\boldsymbol{wH}$ and hence reconstruct $\boldsymbol{w}$.
2) Present to each destination $t\in T$ sufficient descriptions $\varphi_t'(\boldsymbol{x}_1,\boldsymbol{y}_1)$, which together with $\boldsymbol{w}\bar{\boldsymbol{H}}=\boldsymbol{y}_1-\boldsymbol{x}_1$ allow $\boldsymbol{x}_1$ to be reconstructed. In other words, $\varphi_t'(\boldsymbol{x}_1,\boldsymbol{y}_1)$ satisfies

$$H\left(\boldsymbol{x}_1,\boldsymbol{y}_1\,|\,\varphi_t'(\boldsymbol{x}_1,\boldsymbol{y}_1),\boldsymbol{y}_1-\boldsymbol{x}_1\right)=0,\tag{6}$$

Then the destination can reconstruct $\boldsymbol{x}$ from $\tilde{\boldsymbol{x}}=[\boldsymbol{x}_1\ \boldsymbol{xH}]$ and reconstruct $\boldsymbol{y}$ as $\boldsymbol{x}+\boldsymbol{w}$.

For a linear function $\varphi_t'(\boldsymbol{x}_1,\boldsymbol{y}_1)=\boldsymbol{x}_1\boldsymbol{A}_t+\boldsymbol{y}_1\boldsymbol{B}_t$, the condition in (6) boils down to the invertibility of $\boldsymbol{A}_t+\boldsymbol{B}_t$. As a result, we have the following (not yet very practical) scheme for communicating $\{X_i\}$ and $\{Y_i\}$ over a network to the destinations. Partition the total network bit-rate resources $G=(V,E,\mathbf{c})$ into two disjoint shares, $G_1=(V,E,\mathbf{c}_1)$ and $G_2=(V,E,\mathbf{c}_2)$, with $\mathbf{c}_1+\mathbf{c}_2\leq\mathbf{c}$. First, the source nodes generate $\boldsymbol{xH}$ and $\boldsymbol{yH}$ and stream them to all destinations by performing linear network coding with resource G_1. Second, convey certain sufficient descriptions $\boldsymbol{x}_1\boldsymbol{A}_t+\boldsymbol{y}_1\boldsymbol{B}_t$ to each destination $t\in T$ by performing linear network coding with resource G_2. Specifically, a node in the network receives many input bits, each bit being a certain linear combination of $[\boldsymbol{x}_1\ \boldsymbol{y}_1]$; it also outputs bits as linear combinations of the input bits, which are thus linear combinations of the source bits $[\boldsymbol{x}_1\ \ \boldsymbol{y}_1]$.

A. The Overhead Problem and the Field-mismatch Problem

This scheme is not yet fully implementable. The discussions above assumed the operational field is $GF(2)$. Thus $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$ are binary matrices of size $k\times k$. (For efficient SWC k must be sufficiently large, e.g., 10,000 bits.) 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 predetermined coding operations. However, in real networks with lots of dynamics, conveying to t these matrices, which have even more bits than the data, would be a highly difficult task. Yet another problem is that previous discussions of network coding are typically based on $GF(2^p)$ where p is sufficiently large. This leads to an inconsistency in terms of the operating fields for SWC and network coding, because efficient Slepian-Wolf codes for non-binary sources are difficult to design.

We now describe an approach to deal with these problems. For simplicity, it is explained via examples. In the approach, SWC is performed on $GF(2)$ and at the same time, network coding is defined on $GF(2^8)$. Suppose $\boldsymbol{x}_1$ has $k=40,000$ 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

$$\psi: GF(2)^{1\times 40,000}\longrightarrow GF(2^8)^{50\times 100}.\tag{7}$$

Thus the result $\mathbf{X}_1\equiv\psi(\boldsymbol{x}_1)$ is a matrix of size 50×100 with each element defined in $GF(2^8)$. Then we use network coding to transmit to the destinations *descriptions* of the 100 source packets comprising the 50 rows of $\mathbf{X}_1$ from source s_1 and the 50 rows of $\mathbf{Y}_1\equiv\psi(\boldsymbol{y}_1)$ from source s_2. We now review briefly how this can be implemented practically, as in [11], [12]. Each node receives asynchronously via its incoming links a collection of packets, each in the form of

$$[\boldsymbol{\alpha}\ \ \boldsymbol{\beta}]\begin{bmatrix}\mathbf{X}_1\\ \mathbf{Y}_1\end{bmatrix},\tag{8}$$

where the linear combination coefficients $[\boldsymbol{\alpha}_{1\times 50}\ \ \boldsymbol{\beta}_{1\times 50}]$, called the *global encoding 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 (8).

A destination t receives a collection of such mixture packets. Let $[\mathbf{A}_t,\ \mathbf{B}_t]$ denote the global encoding matrix obtained by

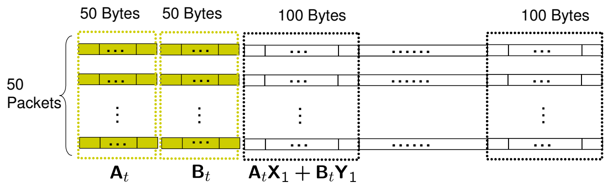

Fig. 2. The structure of packets received by destination t.

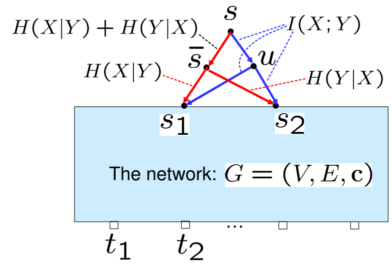

Fig. 3. The admissible rate region for communicating correlated sources over a noiseless network to multiple destinations is equal to the admissible rate region for a multicast problem.

putting together the global encoding vectors of the received packets. Thus t observes $\mathbf{A}_t \cdot \mathbf{X}_1 + \mathbf{B}_t \cdot \mathbf{Y}_1 = \mathbf{A}_t \cdot \psi(\boldsymbol{x}_1) + \mathbf{B}_t \cdot \psi(\boldsymbol{y}_1)$, where $\cdot$ denotes the multiplication in $GF(2^8)$, so as to distinguish from the multiplication in $GF(2)$.

If such 100 independent data packets (50 from s_1, 50 from s_2) 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, our current objective is to present to the destinations sufficient descriptions satisfying (6). As to be shown below, 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.

Suppose $\mathbf{A}_t + \mathbf{B}_t$ is invertible and define $\varphi_t'(\boldsymbol{x}_1, \boldsymbol{y}_1) \equiv \mathbf{A}_t \cdot \psi(\boldsymbol{x}_1) + \mathbf{B}_t \cdot \psi(\boldsymbol{y}_1)$. Now let us verify that condition (6) is satisfied with this definition of φ_t'. Note that the addition (subtraction) operation in any $GF(2^p)$ is the bit-wise XOR operation. Thus,

$$\begin{aligned}&\psi(\boldsymbol{w}\bar{\boldsymbol{H}}) = \psi(\boldsymbol{y}_1 - \boldsymbol{x}_1) = \psi(\boldsymbol{y}_1) - \psi(\boldsymbol{x}_1) = \mathbf{Y}_1 - \mathbf{X}_1,\\ &H(\boldsymbol{x}_1, \boldsymbol{y}_1 | \varphi_t'(\boldsymbol{x}_1, \boldsymbol{y}_1), \boldsymbol{y}_1 - \boldsymbol{x}_1)\\ =&H(\mathbf{X}_1, \mathbf{Y}_1 | \mathbf{A}_t \cdot \mathbf{X}_1 + \mathbf{B}_t \cdot \mathbf{Y}_1, \mathbf{Y}_1 - \mathbf{X}_1) = 0.\end{aligned}$$

Therefore, if $\mathbf{A}_t + \mathbf{B}_t$ is invertible, then $(\boldsymbol{x}, \boldsymbol{y})$ can be decoded with probability of error $P_e'(\boldsymbol{H}, \phi_w)$.

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 $\boldsymbol{x}_1$ of length $40,000$ bits. For example, assuming the packet has $1,100$ bytes, the packet format is illustrated in Fig. 2 and shown below

$$\boldsymbol{\alpha},\ \boldsymbol{\beta}, \boldsymbol{\alpha} \cdot \mathbf{X}_1(1) + \boldsymbol{\beta} \cdot \mathbf{Y}_1(1), \ldots, \boldsymbol{\alpha} \cdot \mathbf{X}_1(10) + \boldsymbol{\beta} \cdot \mathbf{Y}_1(10). \quad (9)$$

B. Probability of error

If $\boldsymbol{x}\boldsymbol{H}$ and $\boldsymbol{y}\boldsymbol{H}$ are successfully delivered to the destination t, decoding is successful with probability

$$\Pr[\mathbf{A}_t + \mathbf{B}_t \text{ is invertible}] \left(1 - P_e'(\boldsymbol{H}, \phi_w)\right). \quad (10)$$

We now show that $\Pr[\mathbf{A}_t + \mathbf{B}_t$ is invertible$]$ is essentially a quantity that has arisen in the conventional network coding context, for multicasting information from a single source to multiple destinations. To see this, consider the case where the two sources are multicasting the same data. Since $\boldsymbol{x}_1 = \boldsymbol{y}_1$, recording $[\boldsymbol{\alpha}\ \boldsymbol{\beta}]$ in (8) is functionally equivalent to recording $\boldsymbol{\alpha} + \boldsymbol{\beta}$. Recall that $G_2 = (V, E, \mathbf{c}_2)$ denotes the allocated bit-rate resources for streaming the descriptions $\mathbf{A}_t \cdot \mathbf{X}_1 + \mathbf{B}_t \cdot \mathbf{Y}_1$. According to known results on distributed random linear network coding (e.g, [11], [12]), roughly speaking, $(1 - \Pr[\mathbf{A}_t + \mathbf{B}_t$ is invertible$])$ can be made very small if the minimum capacity of cuts separating s_1 and s_2 from a destination $t \in T$ in G_2 is above the required communication rate $I(X;Y)$ and the field size for network coding is sufficiently large.

The first term in (10), $\Pr[\mathbf{A}_t + \mathbf{B}_t$ is invertible$]$, comes from network coding, while the second, $(1 - P_e'(\boldsymbol{H}, \phi_w))$, is the probability of successful Slepian-Wolf decoding. Thus the error performance can be fully characterized by the error performance of the two components (SWC and network coding). Simulation results for both practical SWC and practical network coding can be found in the literature (see e.g., [10], [11]). In view of this, we do not include them in this paper.

Note that the streaming of the descriptions $\mathbf{A}_t \cdot \mathbf{X}_1 + \mathbf{B}_t \cdot \mathbf{Y}_1$ can operate in the following mode: the destination can wait until sufficient number of packets have arrived such that an invertible sub-matrix $\mathbf{A}_t + \mathbf{B}_t$ can be found from the observations.

C. Admissible rate region

We begin by reviewing some basics. An *s-t-flow with rate $r \geq 0$* in a graph (V, E) is a length-$|E|$ vector $\mathbf{f}$ satisfying

$$\begin{aligned} f(vw) &\geq 0, \forall vw \in E,\\ \sum_{w \in V: vw \in E} f(vw) - \sum_{u \in V: uv \in E} f(uv) &= 0, \forall v \in V \backslash \{s, t\},\\ \sum_{w \in V: sw \in E} f(sw) - \sum_{u \in V: us \in E} f(us) &= r, \end{aligned} \quad (11)$$

where $f(vw)$ is the component for edge vw. Let $\mathcal{F}(s, t, r)$ denote the solution space of (11).

For multicasting from s to T at rate r, the network coding results in [2] imply that the admissible rate region is

$$C(s, T, r) = \{\mathbf{c} : \exists \mathbf{f}_t \in \mathcal{F}(s, t, r), \mathbf{c} \geq \max_{t \in T} \mathbf{f}_t\}, \quad (12)$$

which can be expressed as a set of linear inequalities.

Recall from the introduction, that the admissible rate region for communicating correlated sources over a noiseless network to multiple destinations, denoted by C, is characterized by three sets of cut conditions [6]. It is easy to see that C is equal to the admissible rate region for a multicasting problem; indeed, introduce a virtual source node s and two other virtual nodes, $\bar{s}$ and u, as illustrated in Fig. 3. Then from the Max-Flow-Min-Cut Theorem, it follows that $C = C(s, T, H(X,Y))$.

We now examine the admissible rate region for our proposed scheme. In the proposed scheme, the network is used for two purposes. First, $\boldsymbol{xH}$ and $\boldsymbol{yH}$ need to be streamed from s_1 and s_2, respectively, to the destinations. This can be viewed as a multicast session from the virtual source $\bar{s}$ in Fig. 3 to the destinations T at rate $H(X|Y)+H(Y|X)$. Second, sufficient descriptions $\mathbf{A}_t \cdot \mathbf{X}_1 + \mathbf{B}_t \cdot \mathbf{Y}_1$ need to be presented to each destination. This can be characterized by a multicast session from u in Fig. 3 to the destinations T at rate $I(X;Y)$.

Note that in general, for multiple multicast sessions, it is suboptimal to separately process the sessions, even if the data in different sessions are independent [16]. For optimality, it is in general necessary to use cross-session network coding. However, cross-session network coding is still not well understood. In addition, separately processing different sessions is easy to implement and analyze. As a result, this approach, although suboptimal, has been applied to multi-source multicasting in recent works, e.g., [8], [17]. Here we restrict our attention to a simple separation solution that allocates disjoint resources, G_1 and G_2, to the two multicast sessions, respectively. The associated admissible rate region, denoted by C^*, is comprised of rate vectors that can supply a SUM of two MAX of flows, corresponding to the two multicast sessions. In other words,

$$C^* = C\left(\bar{s}, T, H(X|Y)+H(Y|X)\right) + C\left(u, T, I(X;Y)\right), \quad (13)$$

which can be expressed by a system of linear inequalities.

This scheme can be generalized by reallocating a portion of bit-rate from the second session to the first session. Specifically, for any $r_1, r_2, \tilde{r} \geq 0$ satisfying $I(X;Y) = r_1 + r_2 + \tilde{r}$, reassign the edge capacities in Fig. 3 as $c(\bar{s}s_1) = H(X|Y)+r_1$, $c(\bar{s}s_2) = H(Y|X)+r_2$, $c(s\bar{s}) = c(\bar{s}s_1)+c(\bar{s}s_2)$, $c(su) = c(us_1) = c(us_2) = \tilde{r}$; then (13) is still admissible. Let C_r^* be the union of the regions (13) for all possible $r_1, r_2, \tilde{r} \geq 0$ satisfying $I(X;Y) = r_1 + r_2 + \tilde{r}$.

We now compare C_r^* to the optimal admissible region C and also to the region of the separation approach studied in [8]. First, if there are only two destinations, then $C = C_r^*$. This follows directly from a graph theoretic result in [8], which was used there to prove that separating SWC and network coding does not lose optimality for the case of two sources and two destinations. Next, recall that the network in Fig. 1 was given in [8] to show that the separation approach is suboptimal. For the case where $H(X) = H(Y) = 1$, $H(W) = 0.5$, and the rate constraint on each edge is 0.5, this example also shows that our proposed scheme is suboptimal. However, for the case where $H(X) = H(Y) = 1$, $H(W) \to 0$, and the rate constraint on each edge $\to 1/3$, $\mathbf{c}$ is admissible in our scheme but not admissible with the separation approach [8].

V. Conclusion

We considered the problem of communicating correlated information from two source nodes over a network to multiple destination nodes. We first presented a design strategy that potentially admits low complexity implementation. The strategy is to convey to each destination node a linearly compressed version of the correlation noise (difference of the two sources), 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 then gave a simple (suboptimal) approach to this problem and constructed a practical communication scheme.

We now mention how the proposed strategy can potentially be generalized to the settings with more than two sources, more general correlation structures, and lossy compression of correlated continuous sources. First, try to redefine the unknowns with a set of independent random processes, some compressible and some not. Next, use the network to: 1) convey to each destination compressed versions of the compressible random processes with low complexity; 2) present sufficient additional descriptions (e.g., random linear descriptions) of the unknowns to the destinations.

Acknowledgement

The authors would like to thank Angelos Liveris and Chih-Chun Wang for the helpful discussions with them.

References

[1] Z. Xiong, A. Liveris, and S. Cheng, "Distributed source coding for sensor networks," *IEEE Signal Processing Magazine*, vol. 21, pp. 80–94, Sept. 2004.

[2] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inform. Theory*, vol. IT-46, no. 4.

[3] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, vol. 19, pp. 471–480, July 1973.

[4] I. Csiszár, "Linear codes for sources and source networks: error exponents, universal coding," *IEEE Trans. Inform. Theory*, Nov. 1982.

[5] T. S. Han, "Slepian-Wolf-Cover theorems for networks of channels," *Inform. and Control*, vol. 47, pp. 67–83, 1980.

[6] L. Song and R. W. Yeung, "Network information flow – multiple sources," in *Proc. Int'l Symp. Inform. Theory*, June 2001.

[7] T. Ho, M. Médard, M. Effros, and R. Koetter, "Network coding for correlated sources," in *Proc. Conf. on Inform. Sci. and Systems (CISS)*, Princeton, NJ, Mar. 2004.

[8] A. Ramamoorthy, K. Jain, P. A. Chou, and M. Effros, "Separating distributed source coding from network coding," in *Proc. 42nd Allerton Conf. on Comm., Control and Computing*, Monticello, IL, Oct. 2004.

[9] A. D. Wyner, "Recent results in the Shannon theory," *IEEE Trans. Inform. Theory*, vol. 20, pp. 2–10, Jan. 1974.

[10] V. Stanković, A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Design of Slepian-Wolf codes by channel code partitioning," in *Proc. Data Compression Conference*, Snowbird, UT, Mar. 2004.

[11] P. A. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Proc. 41st Allerton Conf. Comm., Control, Comp.*, Monticello, IL, Oct. 2003.

[12] T. Ho, R. Koetter, M. Médard, M. Effros, J. Shi, and D. Karger, "Toward a random operation of networks," *IEEE Trans. Inform. Theory*, 2004, submitted.

[13] S. Pradhan and K. Ramchandran, "Generalized coset codes for symmetric distributed source coding," Feb. 2003, submitted.

[14] P. Elias, "Coding for noisy channels," *IRE Convention Record*, pp. 37–46, 1955.

[15] D. J. C. MacKay, *Information Theory, Inference, and Learning Algorithms*. Cambridge, UK: Cambridge Univ. Press, Sept. 2003.

[16] R. W. Yeung, "Multilevel diversity coding with distortion," *IEEE Trans. Inform. Theory*, vol. 41, pp. 412–422, Mar. 1995.

[17] Y. Wu, P. A. Chou, Q. Zhang, K. Jain, W. Zhu, and S.-Y. Kung, "Network planning in wireless ad hoc networks: a cross-layer approach," *IEEE J. Selected Areas in Communications*, vol. 23, no. 1, Jan. 2005.