On the Separation of Lossy Source-Network Coding and Channel Coding in Wireline Networks

Shirin Jalali
Center for Mathematics of Information
California Institute of Technology
Pasadena, California, 91125
Email: shirin@caltech.edu

Michelle Effros
Department of Electrical Engineering
California Institute of Technology
Pasadena, California, 91125
Email: effros@caltech.edu

***Abstract*—This paper proves the separation between source-network coding and channel coding in networks of noisy, discrete, memoryless channels. We show that the set of achievable distortion matrices in delivering a family of dependent sources across such a network equals the set of achievable distortion matrices for delivering the same sources across a distinct network which is built by replacing each channel by a noiseless, point-to-point bit-pipe of the corresponding capacity. Thus a code that applies source-network coding across links that are made almost lossless through the application of independent channel coding across each link asymptotically achieves the optimal performance across the network as a whole.**

I. Introduction

In his seminal work [1], Shannon separates the problem of communicating a memoryless source across a single noisy, memoryless channel into separate lossless source coding and channel coding problems. The corresponding result for lossy coding in point-to-point channels is almost immediate since lossy coding in a point-to-point channel is equivalent to lossless coding of the codeword indices, and it appears in the same work [1]. For a single point-to-point channel, separation holds under a wide variety of source and channel distributions (see, for example [2] and the references therein). Unfortunately, separation does not necessarily hold in network systems. Even in very small networks like the multiple access channel [3], separation can fail when statistical dependencies between the sources at different network locations are useful for increasing the rate across the channel. Since source codes tend to destroy such dependencies, joint source-channel codes can achieve better performance than separate source and channel codes in these scenarios.

This paper proves the separation between source-network coding and channel coding in networks of independent noisy, discrete, memoryless channels (DMC). Roughly, we show that the vector of achievable distortions in delivering a family of dependent sources across such a network $\mathcal{N}$ equals the vector of achievable distortions for delivering the same sources across a distinct network $\hat{\mathcal{N}}$. Network $\hat{\mathcal{N}}$ is built by replacing each channel $p(y|x)$ in $\mathcal{N}$ by a noiseless, point-to-point bit-pipe of the corresponding capacity $C = \max_{p(x)} I(X;Y)$. Thus a code that applies source-network coding across links that are made almost lossless through the application of independent channel coding across each link asymptotically achieves the optimal performance across the network as a whole. Note that the operations of network source coding and network coding are not separable, as shown in [4] and [5] for non-multicast and multicast lossless source coding, respectively. As a result, a joint network-source code is required, and only the channel code can be separated. While the achievability of a separated strategy is straightforward, the converse is more difficult since preserving statistical dependence between codewords transmitted across distinct edges of a network of noisy links improves the end-to-end network performance in some networks [6].

The results derived here give a partial generalization of [7], [8] and [6], which prove the separation between network coding and channel coding for multicast [7], [8] and general demands [6], respectively, under the assumption that messages transmitted to different subset of users are independent and are uniformly distributed. The shift here is from independent sources to dependent sources, from lossless to lossy data description, and from memoryless to non-memoryless sources.

The remainder of the paper is organized as follows. Sections II and III describe the notation and problem set-up, respectively. Section IV describes a tool called a stacked network that allows us to employ typicality across copies of a network rather than typicality across time in the arguments that follow. Section V gives our main results for both memoryless sources and sources with memory.

II. Notation

Calligraphic letters, like $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{U}$, refer to sets, and the size of a set $\mathcal{A}$ is denoted by $|\mathcal{A}|$. For a random variable X, its alphabet set is represented by $\mathcal{X}$.

While a random variable is denoted by X, $\underline{X}$ represents a random vector. The length of a vector is implied in the context, and its ℓ^{th} element is denoted by $\underline{X}(\ell)$.

For two vectors $\underline{x}_1$ and $\underline{x}_2$ of the same length r, $\|\underline{x}_1 - \underline{x}_2\|_1$ denotes the ℓ_1 distance between the two vectors defined as $\|\underline{x}_1 - \underline{x}_2\|_1 = \sum_{i=1}^{r} |\underline{x}_1(i) - \underline{x}_2(i)|$. If $\underline{x}_1$ and $\underline{x}_2$ represent probability distributions, i.e., $\sum_{i=1}^{r} \underline{x}_1(i) = \sum_{i=1}^{r} \underline{x}_2(i) = 1$ and $\underline{x}_1(i), \underline{x}_2(i) \geq 0$ for all $i \in \{1, \ldots, r\}$, then the total variation

distance between $\underline{x}_1$ and $\underline{x}_2$ is defined as $\|\underline{x}_1 - \underline{x}_2\|_{\mathrm{TV}} = 0.5\|\underline{x}_1 - \underline{x}_2\|_1$.

Unlike [6], this paper uses strong typicality arguments to demonstrate the equivalence between noisy channels and noiseless bit-pipes of the same capacity. We therefore assume that the channel input and output alphabets are finite. The alphabets for the sources described across the channel may be discrete or continuous.

III. The problem setup

Consider a multiterminal network $\mathcal{N}$ consisting of m nodes interconnected via some point-to-point, independent DMCs. The network structure is represented by a directed graph G with node set $\mathcal{V} = \{1, \ldots, m\}$ and edge set $\mathcal{E}$. Each directed edge $e = [v_1, v_2] \in \mathcal{E}$ implies a point-to-point DMC between nodes v_1 (input) and v_2 (output). Each node a observes some source process $\mathbf{U}^{(a)} = \{U_k^{(a)}\}_{k=1}^{\infty}$, and is interested in reconstructing a subset of the processes observed by the other nodes. The alphabet of source $\mathbf{U}^{(a)}$, $\mathcal{U}^{(a)}$, can be either scalar or vector-valued. This allows node a to have a vector of sources. For achieving this goal in a block coding framework, source output symbols are divided into non-overlapping blocks of length L. Each block is described separately. At the beginning of the j^{th} coding period, each node a has observed a length-L block of the process $\mathbf{U}^{(a)}$, i.e., $U_{(j-1)L+1}^{(a),jL} = (U_{(j-1)L+1}^{(a)}, \ldots, U_{jL}^{(a)})$. The blocks $\{U_{(j-1)L+1}^{(a),jL}\}_{a\in\mathcal{V}}$ observed at different nodes are described over the network in n uses of the network (The rate $\kappa \triangleq \frac{L}{n}$ is a parameter of the code). For those n time steps, at each step $t \in \{1, \ldots, n\}$, each node a generates its next channel inputs as a function of $U_{(j-1)L+1}^{(a),jL}$ and its channels' outputs up to time $t-1$, here denoted by $Y^{(a),t-1} = (Y_1^{(a)}, \ldots, Y_{t-1}^{(a)})$, according to

$$X_t^{(a)} : (\mathcal{Y}^{(a)})^{t-1} \times \mathcal{U}^{(a),L} \to \mathcal{X}^{(a)}. \tag{1}$$

Note that each node might be the input to more than one channel and/or the output of more than one channel. Hence, both $X_t^{(a)}$ and $Y_t^{(a)}$ might be vectors depending on the indegree and outdegree of node a. The reconstruction at node b of the block observed at node a is denoted by $\hat{U}^{(a\to b),L}$. This reconstruction is a function of the source observed at node b and node b's channel outputs, i.e., $\hat{U}^{(a\to b),L} = \hat{U}^{(a\to b)}(Y^{(b),n}, U^{(b),L})$, where

$$\hat{U}^{(a\to b)} : (\mathcal{Y}^{(b)})^n \times \mathcal{U}^{(a),L} \to \hat{\mathcal{U}}^{(a\to b),L}. \tag{2}$$

The performance criterion for a coding scheme is its induced expected average distortions between sources and reconstruction blocks, i.e., for all $a, b \in \mathcal{V}$

$$\mathrm{E}\, d_L^{(a\to b)}(U^{(a),L}, \hat{U}^{(a\to b),L}) \triangleq \mathrm{E}\, \frac{1}{L}\sum_{k=1}^{L} d^{(a\to b)}(U_k^{(a)}, \hat{U}_k^{(a\to b)}),$$

where $d^{(a\to b)} : \mathcal{U}^{(a)} \times \hat{\mathcal{U}}^{(a\to b)} \to \mathbb{R}^+$ is a per-letter distortion measure. As mentioned before $\mathcal{U}^{(a)}$ and $\hat{\mathcal{U}}^{(a\to b)}$ are either scalar or vector-valued. This allows the case where node a observes multiple sources and node b is interested in reconstructing a subset of them. Let

$$d_{\max} \triangleq \max_{a,b,\in\mathcal{V},\alpha\in\mathcal{U}^{(a)},\beta\in\hat{\mathcal{U}}^{(a\to b)}} d^{(a\to b)}(\alpha,\beta) < \infty.$$

If node b is not interested in reconstructing node a, then $d^{(a\to b)} \equiv 0$.

The distortion matrix $\mathbf{D}$ is said to be achievable at a rate κ in a network $\mathcal{N}$, if for any $\epsilon > 0$, there exists a pair (L, n), $L/n = \kappa$, and block length n coding scheme such that

$$\mathrm{E}\, d_L^{(a\to b)}(U^{(a),L}, \hat{U}^{(a\to b),L}) \le D(a,b) + \epsilon, \tag{3}$$

for any $a, b \in \mathcal{V}$.

IV. Stacked network

For a given network $\mathcal{N}$, the corresponding N-fold stacked network $\underline{\mathcal{N}}$ is defined as N copies of the original network [6]. That is, for each node and each edge in $\mathcal{N}$, there are N copies of the same node or same edge in $\underline{\mathcal{N}}$. At each time instance, each node has access to the data available at nodes which are its copies, and potentially uses this extra information in generating the channel inputs of the future time instances. Likewise, in decoding, all N copies of a node can collaborate in reconstructing the signals. This is made more precise in the following two definitions

$$\underline{X}_t^{(a)} : (\underline{\mathcal{Y}}^{(a)})^{t-1} \times \mathcal{U}^{(a),NL} \to \underline{\mathcal{X}}^{(a)}, \tag{4}$$

and

$$\underline{\hat{U}}^{(a\to b)NL} : (\underline{\mathcal{Y}}^{(b)})^n \times \mathcal{U}^{(b),NL} \to \hat{\mathcal{U}}^{(a\to b),NL}, \tag{5}$$

which correspond to (1) and (2) in the original network. In (4) and (5) all the vectors are of length N.

In an N-layered network, the distortion between the source observed at node a and its reconstruction at node b is defined as

$$D_N(a,b) = \mathrm{E}\left[d_{NL}^{(a\to b)}(U^{(a\to b),NL}, \hat{U}^{(a\to b),NL})\right], \tag{6}$$

for any $a, b \in \{1, \ldots, m\}$.

A distortion matrix $\mathbf{D}$ is said to be achievable in the stacked network at some rate κ if for any given $\epsilon > 0$, there exist N, n and L large enough, such that $D_N(a,b) \le D(a,b) + \epsilon$, for all $a, b \in \{1, \ldots, m\}$. Note that the dimension of the distortion matrices in both single layer and multi-layer networks is $m \times m$. Let $\mathcal{D}(\kappa, \mathcal{N})$ and $\mathcal{D}_s(\kappa, \underline{\mathcal{N}})$ denote the closure of the set of achievable distortion matrices at some rate κ in a network $\mathcal{N}$ and its stacked version $\underline{\mathcal{N}}$ respectively. The following theorem establishes the relationship between the two sets.

Theorem 1: At any rate κ,

$$\mathcal{D}(\kappa, \mathcal{N}) = \mathcal{D}_s(\kappa, \underline{\mathcal{N}}). \tag{7}$$

Proof:

i. Proof of $\mathcal{D}(\kappa, \mathcal{N}) \subseteq \mathcal{D}_s(\kappa, \underline{\mathcal{N}})$. Consider any $\mathbf{D} \in \mathrm{int}(\mathcal{D}(\kappa, \mathcal{N}))$. Then for any $\epsilon > 0$, there exists a coding operating scheme at rate $\kappa = L/n$ on $\mathcal{N}$ such that (3) is satisfied. For any N, a stacked network that uses

this same coding strategy independently in each layer achieves

$$\begin{aligned}\mathrm{E}[d_{NL}^{(a\to b)}(U^{(a\to b),NL},\hat{U}^{(a\to b),NL})] &= \frac{1}{N}\sum_{\ell=1}^{N}\mathrm{E}[d_{L}^{(a\to b)}(U_{(\ell-1)L+1}^{(a\to b),\ell L},\hat{U}_{(\ell-1)L+1}^{(a\to b),\ell L})]\\ &\le \frac{1}{N}\sum_{\ell=1}^{N}D(a,b)+\epsilon\\ &= D(a,b)+\epsilon. \end{aligned} \tag{8}$$

ii. $\mathcal{D}_s(\kappa,\underline{\mathcal{N}}) \subseteq \mathcal{D}(\kappa,\mathcal{N})$. Let $\mathbf{D}\in \mathrm{int}(\mathcal{D}_s(\kappa,\underline{\mathcal{N}}))$. Since $\mathbf{D}\in \mathrm{int}(\mathcal{D}_s(\kappa,\underline{\mathcal{N}}))$, for any $\epsilon>0$, there exists integers N, n, and L such that a stacked network consisting of N layers along with a block length n coding scheme for L source symbols on this stacked network achieves

$$\mathrm{E}\left[d_{NL}^{(a\to b)}(U^{(a\to b),NL},\hat{U}^{(a\to b),NL})\right]\le D(a,b)+\epsilon,$$

for all $a,b\in\mathcal{V}$. The same coding scheme can be used in a single-layer network as follows. Consider a single layer network where each node observes a length-NL block of source symbols and describes the block in the next Nn time steps. At times $t\in\{1,\dots,N\}$, each node a sends what would have been sent at time 1 by node a in layer t of the stacked network. After that, having collected the output of the previous N time steps, at times $t\in\{N+1,\dots,2N\}$, node a sends the outputs of the same node at time 2 in layer $t-N$ (Note that in the first N time steps, node a's output is only a function of its own source, not the channels' outputs. It only collects the channel outputs in order to use them during the next N time steps.). The same strategy is used in n time intervals, each comprising N network uses. During each period, the new channel outputs observed by node a are recorded to be used in the future periods, but do not affect the next inputs generated by that node during that time period. Using this strategy, at the end of nN channel uses, each node's observation has exactly the same distribution as the collection of observations of its N copies in the stacked networks. Therefore, applying the same decoding rule will result in the same performance. Hence, $\mathbf{D}\in\mathcal{D}(\kappa,\mathcal{N})$.

■

V. Replacing a noisy channel with a bit pipe

A. Memoryless sources

In this section we assume all sources are jointly i.i.d., i.e., for any $k\ge 1$, $\mathrm{P}(U^{(1),k},\dots,U^{(m),k}) = \prod_{i=1}^{k}\mathrm{P}(U_i^{(1)},\dots,U_i^{(m)})$, where $\mathrm{P}(U_i^{(1)},\dots,U_i^{(m)})$ does not depend on i. Note that at each time instant the sources might be correlated with each other.

In the described network $\mathcal{N}$, for some $a,b\in\mathcal{V}$ such that $[a,b]\in\mathcal{E}$, consider the noisy channel connecting these two nodes. The channel is described by its transition probabilities $\{p(y|x)\}_{x\in\mathcal{X},y\in\mathcal{Y}}$, and has some finite capacity $C=\max_{p(x)} I(X;Y)$. Now consider a network $\mathcal{N}'$ which is identical to $\mathcal{N}$ except for the noisy channel between a and b, which is replaced by a bit-pipe of capacity C.

Theorem 2: For any $\kappa>0$,

$$\mathcal{D}(\kappa,\mathcal{N})=\mathcal{D}(\kappa,\mathcal{N}'). \tag{9}$$

Proof outline: By Theorem 1, the achievable region of a network is equal to the achievable region of its stacked version. Hence, it suffices to prove that $\mathcal{D}_s(\kappa,\underline{\mathcal{N}})=\mathcal{D}_s(\kappa,\underline{\mathcal{N}'})$.

i. $\mathcal{D}_s(\kappa,\underline{\mathcal{N}'})\subseteq\mathcal{D}_s(\kappa,\underline{\mathcal{N}})$: Let $\mathbf{D}\in\mathrm{int}(\mathcal{D}_s(\kappa,\underline{\mathcal{N}'}))$. We need to show that $\mathbf{D}\in\mathcal{D}_s(\kappa,\underline{\mathcal{N}})$ as well. Note that $\mathcal{N}$ and $\mathcal{N}'$ are identical except for the DMC connecting nodes a and b in $\mathcal{N}$ which is replaced by a bit-pipe of capacity C in $\mathcal{N}'$. We next show that any code for $\underline{\mathcal{N}'}$ can be operated on $\underline{\mathcal{N}}$ with a similar expected distortion. Let the number of layers in both networks be N. Given the capacity of the bit-pipes, the number of bits that can be carried from a to b in $\underline{\mathcal{N}'}$ is at most NR, where $R<C$. Hence, if N is large enough, the same information can be transmitted from a to b in $\underline{\mathcal{N}}$ by doing appropriate channel coding across the layers over the noisy channel and its copies connecting a and b in $\underline{\mathcal{N}}$. Let $P_{e,(a\to b)}$ denote the probability of error of the channel code operating over the channel corresponding to the edge $[a,b]$ and its copies in $\underline{\mathcal{N}}$, and let $P_{e,\max}=\max_{[a,b]\in\mathcal{E}}P_{e,a\to b}$. Then the extra expected distortion introduced at each reconstruction point is bounded above by $|\mathcal{E}|P_{e,\max}d_{\max}$ and can be made arbitrarily small.

ii. $\mathcal{D}(\kappa,\underline{\mathcal{N}})\subseteq\mathcal{D}_s(\kappa,\underline{\mathcal{N}'})$: Let $\mathbf{D}\in\mathrm{int}(\mathcal{D}(\kappa,\mathcal{N}))$. We prove that $\mathbf{D}\in\mathcal{D}_s(\kappa,\underline{\mathcal{N}'})$. Consider a code defined on $\mathcal{N}$ that achieves within ϵ of $\mathbf{D}$, and consider the N-fold stacked version of $\mathcal{N}$, $\underline{\mathcal{N}}$. Assume that the same code is applied independently in each layer. We first show that, when all sources are memoryless and uniformly distributed, the performance of the code given the realization of $(\underline{X}_1,\underline{Y}_1)$ only depends on the empirical distribution of $(\underline{X}_1,\underline{Y}_1)$ defined as

$$\hat{p}_{[\underline{X}_1,\underline{Y}_1]}(x,y)=\frac{1}{N}\sum_{\ell=1}^{N}\mathbb{1}_{(\underline{X}_1(\ell),\underline{Y}_1(\ell))=(x,y)}, \tag{10}$$

for all $x\in\mathcal{X}$ and $y\in\mathcal{Y}$. Here the subscript 1 refers to time $t=1$. After establishing this, we use the result proved in [9] and show that at time $t=1$ we can simulate the performance of the noisy link by a bit-pipe of the same capacity. For the rest of the proof, let $U=\{U_i\}$ and $\hat{U}=\{\hat{U}_i\}$ denote some i.i.d. source observed at some node in $\mathcal{V}$ and its reconstruction at some other node in $\mathcal{V}$.

In the original network,

$$\begin{aligned}\mathrm{E}\,d_L(U^L,\hat{U}^L)=\sum_{\substack{x\in\mathcal{X}\\ y\in\mathcal{Y}}}&\mathrm{E}\left[d_L(U^L,\hat{U}^L)\,|(X_1,Y_1)=(x,y)\right]\\ &\times\mathrm{P}\left((X_1,Y_1)=(x,y)\right).\end{aligned} \tag{11}$$

On the other hand, in the N-fold stacked network,

$$\begin{aligned}
&\mathrm{E}\left[d_{NL}(U^{NL},\hat{U}^{NL})\right]\\
&=\mathrm{E}\left[\sum_{\ell=1}^{N}\frac{d_L\left(U_{(\ell-1)L+1}^{\ell L},\hat{U}_{(\ell-1)L+1}^{\ell L}\right)}{N}\times\right.\\
&\qquad\left.\sum_{\substack{x\in\mathcal{X}\\ y\in\mathcal{Y}}}\mathbb{1}_{(\underline{X}_1(\ell),\underline{Y}_1(\ell))=(x,y)}\right]\\
&=\mathrm{E}\left[\sum_{\substack{x\in\mathcal{X}\\ y\in\mathcal{Y}}}\sum_{\ell=1}^{N}\right.\\
&\left.\frac{d_L\left(U_{(\ell-1)L+1}^{\ell L},\hat{U}_{(\ell-1)L+1}^{\ell L}\right)\mathbb{1}_{(\underline{X}_1(\ell),\underline{Y}_1(\ell))=(x,y)}}{N}\right]\\
&=\sum_{\substack{x\in\mathcal{X}\\ y\in\mathcal{Y}}}\mathrm{E}\left[d_L(U^L,\hat{U}^L)\,|(X_1,Y_1)=(x,y)\right]\times\\
&\qquad\mathrm{E}[\hat{p}_{[\underline{X}_1,\underline{Y}_1]}(x,y)]. \qquad (12)
\end{aligned}$$

Comparing (11) and (12) reveals that the desired result will follow if we can find a coding scheme for which,

$$\left|\mathrm{P}\left((X_1,Y_1)=(x,y)\right)-\mathrm{E}[\hat{p}_{[\underline{X}_1,\underline{Y}_1]}(x,y)]\right|, \qquad (13)$$

can be made arbitrary small.

To prove this, consider a channel with input drawn i.i.d. from some distribution $p(x)$. The encoder observes N source symbols and sends a message of NR bits to the decoder. The decoder converts these NR bits into a reconstruction block $\underline{Y}=(Y_1,\ldots,Y_N)$. The empirical joint distribution between the channel input and channel output induced by the bit pipe is defined in the classical sense as follows

$$\hat{p}_{[\underline{X},\underline{Y}]}(x,y)=\frac{1}{N}\sum_{\ell=1}^{N}\mathbb{1}_{(\underline{X}(\ell),\underline{Y}(\ell))=(x,y)}.$$

Consider a DMC described by transition probabilities $\{p(y|x)\}_{x\in\mathcal{X},y\in\mathcal{Y}}$ whose input is an i.i.d. process distributed according to some distribution $p(x)$. In [9], it is shown that, as long as $R>I(X;Y)$, any such channel can be simulated by a bit pipe of rate at most R such that the total variation between $\hat{p}_{[\underline{X},\underline{Y}]}(x,y)$ and $p(x,y)=p(x)p(y|x)$ can be made arbitrarily small for large enough block lengths. In other words, there exists a sequence of coding schemes over the bit-pipe such that

$$\left\|\hat{p}_{[\underline{X},\underline{Y}]}-p\right\|_1\overset{n\to\infty}{\longrightarrow}0 \text{ a.s.} \qquad (14)$$

(where $\hat{p}_{[\underline{X},\underline{Y}]}$ and p are vectors describing distributions $(\hat{p}_{[\underline{X},\underline{Y}]}(x,y):x,y\in\mathcal{X},\mathcal{Y})$ and $(p(x,y):x,y\in\mathcal{X},\mathcal{Y})$ respectively.)

Combining this result with our initial claim yields the desired result, i.e., at time $t=1$, we can replace the noisy link by a bit-pipe. To extend this result to the next $n-1$ time steps, we use induction. Note that in the original network

$$\begin{aligned}
&\mathrm{E}\,d_L(U^L,\hat{U}^L)=\\
&\sum_{\substack{x_t\in\mathcal{X}\\ y_t\in\mathcal{Y}\\ t=1,\ldots,n}}\mathrm{E}\left[d_L(U^L,\hat{U}^L)\left|\bigcap_{t=1}^{n}\{(X_t,Y_t)=(x_t,y_t)\}\right.\right]\\
&\qquad\times\mathrm{P}\left((X^n,Y^n)=(x^n,y^n)\right). \qquad (15)
\end{aligned}$$

On the other hand, using the same analysis used in deriving (12), in the N-fold stacked network,

$$\begin{aligned}
&\mathrm{E}\left[d_{NL}(U^{NL},\hat{U}^{NL})\right]\times\\
&=\sum_{\substack{x_t\in\mathcal{X}\\ y_t\in\mathcal{Y}\\ t=1,\ldots,n}}\mathrm{E}\left[d_L(U^L,\hat{U}^L)\,|(X^n,Y^n)=(x^n,y^n)\right]\\
&\qquad\times\mathrm{E}\left[\frac{\left|\left\{\ell:(\underline{X}^t(\ell),\underline{Y}^t(\ell))=(x^t,y^t)\right\}\right|}{L}\right]. \qquad (16)
\end{aligned}$$

Therefore, we need to show that by appropriate coding over the bit-pipes,

$$\begin{aligned}
&\left|\mathrm{P}\left((X^n,Y^n)=(x^n,y^n)\right)\right.\\
&\qquad\left.-\mathrm{E}\left[\frac{\left|\left\{\ell:(\underline{X}^t(\ell),\underline{Y}^t(\ell))=(x^t,y^t)\right\}\right|}{L}\right]\right| \qquad (17)
\end{aligned}$$

can be made arbitrary small. Note that

$$\begin{aligned}
&\mathrm{P}\left((X^n,Y^n)=(x^n,y^n)\right)=\prod_{t=1}^{n}\\
&\mathrm{P}\left((X_t,Y_t)=(x_t,y_t)\,|(X^{t-1},Y^{t-1})=(x^{t-1},y^{t-1})\right), \qquad (18)
\end{aligned}$$

and

$$\begin{aligned}
&\frac{\left|\left\{\ell:(\underline{X}^n(\ell),\underline{Y}^n(\ell))=(x^n,y^n)\right\}\right|}{L}\\
&=\prod_{t=1}^{n}\frac{\left|\left\{\ell:(\underline{X}^t(\ell),\underline{Y}^t(\ell))=(x^t,y^t)\right\}\right|}{\left|\left\{\ell:(\underline{X}^{t-1}(\ell),\underline{Y}^{t-1}(\ell))=(x^{t-1},y^{t-1})\right\}\right|}, \qquad (19)
\end{aligned}$$

where for $t=1$

$$\left|\left\{\ell:(\underline{X}^{t-1}(\ell),\underline{Y}^{t-1}(\ell))=(x^{t-1},y^{t-1})\right\}\right|=L.$$

We have already proved that by appropriate coding, we can make the first term in (19) converge to the first term in (18) with probability one. By induction, we can prove that the same result is true for any other term in (19) and its corresponding term in (18). After proving this, since all the terms in (19) and as a result their product are positive and upper-bounded by 1, we can use the Dominated Convergence Theorem (see, for example, [10]) to show that (17) can be made arbitrary small.

To apply induction, assume there exist some coding schemes by which we make the first $t-1$ terms in (19)

each converge to the corresponding term in (18) almost surely. Using this assumption, we prove that the same thing is true for the t^{th} term as well.

Note that when the first $t-1$ terms are very close, the frequency of occurrence of each pattern $\{(\underline{X}^{t-1}(\ell), \underline{Y}^{t-1}(\ell)) = (x^{t-1}, y^{t-1})\}$ across the layers in $\underline{\mathcal{N}}$ is very close to the pattern's probability. Since the two networks perform the same except for link $[a,b]$, the network guarantees that the frequency of $\{(\underline{X}^{t}(\ell), \underline{Y}^{t-1}(\ell)) = (x^{t}, y^{t-1})\}$ is also close to its probability in $\underline{\mathcal{N}}$. In order to finish the proof, we use Lemma 1 proved in Appendix 1.

Lemma 1: If we choose the random codes used at times $t-1$ and t independently, then

$$\begin{aligned}&\mathrm{E}[\mathbb{1}_{\underline{Y}_t(1)=y_t}|(\underline{X}_{t-1}(1), \underline{Y}_{t-1}(1)) = (x_{t-1}, y_{t-1}),\\ &\quad \underline{X}_t(1) = x_t] = \mathrm{P}\left(\underline{Y}_t(1) = y_t | \underline{X}_t(1) = x_t\right), \end{aligned} \quad (20)$$

where the expectation is both with respect to the network and the code selections.

■

B. Sources with memory

Assume that the sources are no longer memoryless but mixing. That is for any integers k and T

$$\begin{aligned}&\Big|\mathrm{P}\left((U^{(1),k}, \ldots, U^{(m),k}, U_T^{(1),T+k}, \ldots, U_T^{(m),T+k}) = \right.\\ &\qquad \left.(u^{(1),k}, \ldots, u^{(m),k}, u_T^{(1),T+k}, \ldots, u_T^{(m),T+k})\right) -\\ &\mathrm{P}\left((U^{(1),k}, \ldots, U^{(m),k}) = (u^{(1),k}, \ldots, u^{(m),k})\right) \times\\ &\mathrm{P}\left((U_T^{(1),T+k}, \ldots, U_T^{(m),T+k}) = (u_T^{(1),T+k}, \ldots, u_T^{(m),T+k})\right)\Big|\end{aligned}$$

goes to 0 as T approaches ∞. In the proof of Theorem 2, we used the fact that the sources are correlated and jointly i.i.d. to conclude that the inputs to the copies of a channel in the stacked network are i.i.d. If the sources have memory, this does not hold any more. But, if we assume that the sources are mixing, then for block length L large enough, the two sets $\{U^L, U_{2L+1}^{3L}, \ldots\}$ and $\{U_{L+1}^{2L}, U_{3L+1}^{4L}, \ldots\}$ look like two i.i.d. sequences. Therefore, in the stacked network, if we code the even-numbered layers together and the odd-numbered ones together, such that each one is done separate from the other one, we get back to the i.i.d. regime and can prove a similar result.

Appendix A: Proof of Lemma 1

Note that

$$\begin{aligned}&\mathrm{E}\left[\mathbb{1}_{\underline{Y}_t(1)=y_t}|\underline{X}_{t-1}(1) = x_{t-1}, \underline{Y}_{t-1}(1) = y_{t-1}, \underline{X}_t(1) = x_t\right]\\ &= \sum_{\underline{s}_1, \hat{\underline{s}}_1, \underline{s}_2} \mathrm{P}\left(\underline{Y}_t(1) = y_t, \underline{X}_{t-1}(2:N) = \underline{s}_1,\right.\\ &\qquad \underline{Y}_{t-1}(2:N) = \hat{\underline{s}}_1, \underline{X}_t(2:N) = \underline{s}_2 | \underline{X}_{t-1}(1) = x_{t-1},\\ &\qquad \left.\underline{Y}_{t-1}(1) = y_{t-1}, \underline{X}_t(1) = x_t\right)\\ &= \sum_{\underline{s}_2} \mathrm{P}(\underline{Y}_t(1) = y_t | \underline{X}_t = [x_t, \underline{s}_2])\, \mathrm{P}(\underline{X}_t(2:N) = \underline{s}_2 |\\ &\quad \underline{X}_{t-1}(1) = x_{t-1}, \underline{Y}_{t-1}(1) = y_{t-1}, \underline{X}_t(1) = x_t). \end{aligned} \quad (A\text{-}1)$$

But

$$\begin{aligned}&\mathrm{P}(\underline{Y}_t(1) = b | \underline{X}_t = \underline{x}_t)\\ &= \sum_{\underline{y}_t : \underline{y}_t(1) = b} \frac{\mathrm{P}(\underline{X}_t = \underline{x}_t, \underline{Y}_t = \underline{y}_t)}{\mathrm{P}(\underline{X}_t = \underline{x}_t)} \end{aligned} \quad (A\text{-}2)$$

$$\begin{aligned}&= \frac{1}{\mathrm{P}(\underline{X}_t = \underline{x}_t)} \sum_{\underline{Y}_t : \underline{Y}_t(1) = b} \mathrm{P}(\underline{X}_t(1) = \underline{x}_t(1)) \times\\ &\mathrm{P}(\underline{Y}_t(1) = y_t | \underline{X}_t(1) = \underline{x}_t(1)) \times\\ &\mathrm{P}(\underline{X}_t(2:N) = \underline{x}_t(2:N) | \underline{X}_t(1) = \underline{x}_t(1), \underline{Y}_t(1) = b) \times\\ &\mathrm{P}(\underline{Y}_t(2:N) = \underline{y}_t(2:N) | \underline{X}_t = \underline{x}_t, \underline{Y}_t(1) = \underline{y}_t(1)) \end{aligned} \quad (A\text{-}3)$$

$$\begin{aligned}&= \frac{1}{\mathrm{P}(\underline{X}_t = \underline{x}_t)} \sum_{\underline{y}_t : \underline{y}_t(1) = b} \mathrm{P}(\underline{X}_t(1) = \underline{x}_t(1)) \times\\ &\mathrm{P}(\underline{Y}_t(1) = y_t | \underline{X}_t(1) = \underline{x}_t(1)) \times\\ &\mathrm{P}(\underline{X}_t(2:N) = \underline{x}_t(2:N) | \underline{X}_t(1) = \underline{x}_t(1)) \times\\ &\mathrm{P}(\underline{Y}_t(2:N) = \underline{y}_t(2:N) | \underline{X}_t = \underline{x}_t, \underline{y}_t(1) = b) \end{aligned} \quad (A\text{-}4)$$

$$\begin{aligned}&= \mathrm{P}(\underline{Y}_t(1) = b | \underline{X}_t(1) = \underline{x}_t(1)) \times\\ &\quad \sum_{\underline{y}_t(1) = y_t} \mathrm{P}(\underline{Y}_t(2:N) = \underline{y}_t(2:N) | \underline{X}_t = \underline{x}_t, \underline{Y}_t(1) = b)\\ &= \mathrm{P}(\underline{Y}_t(1) = b | \underline{X}_t(1) = \underline{x}_t(1)). \end{aligned} \quad (A\text{-}5)$$

Combining (A-1) and (A-5) yields the desired result.

Acknowledgments

SJ is supported by the Center for Mathematics of Information at Caltech, and ME is supported by the DARPA ITMANET program under grant number W911NF-07-1-0029.

References

[1] C. E. Shannon,"A mathematical theory of communications: Parts I and II," *Bell Syst. Tech. J.*, vol. 27, pp. 379423, 623656, 1948.

[2] S. Vembu, S. Verdu, and Y. Steinberg, "The source-channel separation theorem revisited," *IEEE Trans. Info. Theory*, vol. 41, no. 1, pp. 44-54, Jan. 1995.

[3] A. El Gamal and T. M. Cover, "Multiple user information theory," *Proc. IEEE*, vol. 68, pp. 14661483, Dec. 1980.

[4] M. Effros, M. Médard, T. Ho, S. Ray, D. Karger and R. Koetter, "Linear network codes: a unified framework for source, channel, and network coding," *Proc. of the DIMACS Workshop on Network Info. Theory*, Piscataway, NJ, March 2003.

[5] A. Ramamoorthy, K. Jain, P. A. Chou, and M. Effros, "Separating distributed source coding from network coding," *IEEE Transactions on Information Theory*, vol. 52, pp. 27852795, June 2006.

[6] R. Koetter, M. Effros, and M. Médard. "On the theory of network equivalence," *IEEE Inform. Theory Workshop (ITW)*, 2009.

[7] S. Borade, "Network Information Flow: Limits and Achievability," In *Proc. IEEE Int. Symp. Inform. Theory (ISIT)*, Lausanne, Switzerland, 2002.

[8] L. Song, R. W. Yeung, and N. Cai, "A separation theorem for single-source network coding," *IEEE Transactions on Information Theory*, vol. 52, pp. 1861-1871, May 2006.

[9] P. Cuff, H. Permuter, T.M. Cover. "Coordination Capacity," submitted to *IEEE Trans. on Info.Theory*, Aug. 2009 (available at arxiv.org/abs/0909.2408).

[10] R. Durrett, *Probability. Theory and examples,* Wadsworth & Brooks/Cole, Pacific Grove, CA, 1991.