A Theory of Network Equivalence—Part I: Point-to-Point Channels

Ralf Koetter, *Fellow, IEEE*, Michelle Effros, *Fellow, IEEE*, and Muriel Médard, *Fellow, IEEE*

***Abstract*—A family of equivalence tools for bounding network capacities is introduced. Given a network $\mathcal{N}$ with node set $\mathcal{V}$, the capacity of $\mathcal{N}$ is a set of non-negative vectors with elements corresponding to all possible multicast connections in $\mathcal{N}$; a vector $\mathcal{R}$ is in the capacity region for $\mathcal{N}$ if and only if it is possible to simultaneously and reliably establish all multicast connections across $\mathcal{N}$ at the given rates. Any other demand type with independent messages is a special case of this multiple multicast problem, and is therefore included in the given rate region. In Part I, we show that the capacity of a network $\mathcal{N}$ is unchanged if any independent, memoryless, point-to-point channel in $\mathcal{N}$ is replaced by a noiseless bit pipe with throughput equal to the removed channel's capacity. It follows that the capacity of a network comprised entirely of such point-to-point channels equals the capacity of an error-free network that replaces each channel by a noiseless bit pipe of the corresponding capacity. A related separation result was known previously for a single multicast connection over an acyclic network of independent, memoryless, point-to-point channels; our result treats general connections (e.g., a collection of simultaneous unicasts) and allows cyclic or acyclic networks.**

***Index Terms*—Capacity, component models, equivalence, network coding.**

I. Introduction

THE STUDY of network communications has two natural facets reflecting different approaches to thinking about networks. On the one hand, networks are considered in the graph theoretic setup consisting of nodes connected by links. The links are typically not noisy channels, but rather noise-free bit pipes that can be used error free up to a certain capacity. Typical questions concern information flows and routing strategies. On the other hand, multiterminal information theory addresses information transmission through networks by studying noisy channels, or rather the stochastic relationship between input and output signals at devices in a network. Here, the questions typically concern fundamental limits of communication. The capacity regions of broadcast, multiple-access, and interference channels are all examples of questions that are addressed in the context of multiterminal information theory. These questions appear to have no obvious equivalents in networks consisting of error-free bit pipes. Nevertheless, these two views of networking are two natural facets of the same problem, namely communication through networks. This work explores the relationship between these two worlds.

Establishing viable bridges between these two areas proves to be surprisingly fertile. For example, questions about feedback in multiterminal systems are quite nicely expressed in terms of networks of error-free bit pipes. Separation issues—in particular, separation between network coding and channel coding—have natural answers, revealing many network capacity problems as combinatorial rather than statistical, even when communication occurs across networks of noisy channels. Most importantly, bounding general network capacities reduces to solving a central network coding problem described as follows. Given a network of *error-free*, rate-constrained bit pipes, is a given set of connections (e.g., a collection of multicast connections) simultaneously feasible or not. In certain situations, most notably under a single multicast connection, this question has been solved, and the answer is easily characterized [1]. Unfortunately, the general case is wide open, and is suspected to be hard. (Currently, NP hardness is established only for scalar linear network coding [2].) While it appears that fully characterizing the combinatorial network coding problem is out of reach [3], networks of moderate size can be solved quite efficiently, and there are algorithms available that treat precisely this problem with running time that is exponential in the number of nodes [4]–[6]. The observation that, in principle, it is possible to characterize the rate region of a network coding problem will be a cornerstone for our investigations.

The combinatorial nature of the network coding problem creates a situation not unlike that found in complexity theory. In that case, since precise expressions as to how difficult a problem is in absolute terms are difficult to derive, research is devoted instead to showing that one problem is essentially as difficult as another one (even though precise characterizations are not available for either). Inspired by this analogy, we take a similar approach here, demonstrating the relationship between the capacity of a stochastic network and the capacity of a network of noiseless bit pipes, without characterizing the capacity of either network. In fact, this relationship is all we need if we want to address separation issues in networks. It also opens the door to other questions, such as degree-of-freedom or high-signal-to-noise-ratio analyses, which reveal interesting insights.

Manuscript received April 14, 2010; revised October 25, 2010; accepted December 02, 2010. Date of current version January 19, 2011. This work was supported in part by DARPA under the Information Theory for Mobile Ad-Hoc Networks (ITMANET) Program, Grant W911NF-07-1-0029, and by the Lee Center for Advanced Networking at Caltech. The material in this paper was presented in part at the IEEE Information Theory Workshop, Volos, Greece, June 2009, and the 47th Annual Allerton Conference on Communication, Control, and Computing, Monticello, IL, September 2009.

This paper is part of the special issue on "Facets of Coding Theory: From Algorithms to Networks," dedicated to the scientific legacy of Ralf Koetter.

R. Koetter, deceased, was with the Technical University of Munich, Munich Germany.

M. Effros is with the Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125 USA (e-mail: effros@caltech.edu).

M. Médard is with the Department of Electrical Engineering and Computer Science, Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: medard@mit.edu).

Communicated by G. D. Forney, Jr., Associate Editor for the special issue on "Facets of Coding Theory: From Algorithms to Networks."

Digital Object Identifier 10.1109/TIT.2010.2102110

It is interesting to note the variety of new tools generated in recent years for studying network capacities (e.g., [1], [3]–[5], [7]–[12]). The reduction of a network information theoretic question to its combinatorial essence is also at the heart of some of these publications (see, e.g., [12]). Our present work is very different in both technique and results. Our goal is not to derive network capacity regions, but rather to develop equivalence relationships between the capacity regions of distinct networks. In other words, we wish to show that any collection of connections is feasible on one network if and only if it is feasible on another. Since the solution of general network capacities is out of reach, we prove such equivalences without deriving the capacity of either network. While this approach is different from that of other authors, we believe it to be no coincidence that the reduction of a problem to its combinatorial essence plays a central role in a variety of techniques for studying network capacities.

II. Motivation and Summary of Results

Traditionally, the information-theoretic investigation of network capacities has proceeded largely by studying example networks. Shannon's original proof of the capacity of a network described by a single point-to-point channel [13] was followed by Ahlswede's [14] and Liao's [15] capacity derivations for a single multiple-access channel, Cover's early work on a single broadcast channel [16], and so on. While the solution to one network capacity problem may lend insight into future problems, deriving the capacities of new networks is often difficult. As a result, even the capacities for three-node networks remain incompletely solved.

For most large networks, direct derivation of network capacities is out of reach. We therefore seek to divide the network capacity problem into subproblems whose solutions lend insight into the problem at hand. Given a network of independent channels,[1] we seek a simple characterization of each channel's behavior that captures that channel's impact on the broader network. The channel capacity is an obvious candidate characterization. Note, however, that channel capacity captures the rate at which we can reliably communicate across a channel, and reliable communication across a network does not require reliable communication across all channels in that network. In fact, operating each channel at its respective capacity often fails to yield the optimal communication performance across the network as a whole. As a result, channel capacity is not necessarily a relevant characterization of a channel's impact on a larger network.

The following examples illustrate this point. Each establishes a single unicast connection across a network of independent channels. In Example 1, operating point-to-point channels at twice their respective capacities gives a factor of two improvement in the network's error exponent. In Examples 2 and 3, operating a broadcast channel above its capacity increases the rate that can be reliably delivered through the network.

Example 1: Consider the problem of establishing a unicast connection over the two-node network shown in Fig. 1(a). Node 1 transmits a pair of channel inputs $X^{(1)} = (X^{(1,1)}, X^{(1,2)})$. Node 2 receives a pair of channel outputs $Y^{(2)} = (Y^{(2,1)}, Y^{(2,2)})$. The inputs and outputs are stochastically related through a pair of independent but identical channels; i.e.,

$$p(y^{(2,1)}, y^{(2,2)}|x^{(1,1)}, x^{(1,2)}) = p(y^{(2,1)}|x^{(1,1)})p(y^{(2,2)}|x^{(1,2)})$$

for all $(x^{(1,1)}, x^{(1,2)}, y^{(2,1)}, y^{(2,2)})$, while

$$p(y^{(2,1)}|x^{(1,1)}) = p(y^{(2,2)}|x^{(1,2)})$$

when $(x^{(1,1)}, y^{(2,1)}) = (x^{(1,2)}, y^{(2,2)})$. Let

$$C = \max_{p(x^{(1,1)})} I(X^{(1,1)}; Y^{(2,1)}) = \max_{p(x^{(1,2)})} I(X^{(1,2)}; Y^{(2,2)})$$

be the capacity of each channel. For each rate $R < C$ and each blocklength n, we compare two strategies for reliably communicating from node 1 to node 2 at rate $2R$. The first (see Fig. 1(b)) reliably communicates over each link using an optimal $(2^{nR}, n)$ channel code. The second (see Fig. 1(c)) applies a single optimal $(2^{(2n)R}, 2n)$ channel code across the pair of channels, sending the first n symbols of a codeword across the first channel and the remaining n symbols across the second channel, and jointly decoding the channel outputs using its blocklength-$2n$ channel decoder. Using this approach, each channel may have as many as 2^{2nR} possible inputs. Thus when R is close to C, this code operates each channel at up to twice its capacity, making reliable transmission across each individual channel impossible. Since the first strategy operates an n-dimensional code over n time steps while the second strategy operates a $2n$-dimensional code over n time steps, the error probability of the second strategy decays to zero far more quickly than that of the first code. The difference is a factor of two in the error exponent. ■

In Example 1, the penalty for operating a pair of point-to-point channels at their individual capacities is a factor of two in the error exponent. This does not, however, imply that operating channels at their individual capacities fails to achieve the network capacity. In fact, [7] and [17] prove that separation holds for single-source multicast connections across acyclic networks of independent, memoryless point-to-point channels. To prove the result, the authors show that separate network and channel codes achieve the cut-set outer bound on the multicast capacity. Unfortunately, this result is difficult to extend. First, cut-set bounds on the network capacity are not tight in general, and finding good outer bounds and then proving their achievability for all possible connection types on all possible networks is not feasible. Further, if we consider more general network types, then separation between network and channel coding fails even for a single unicast connection, as shown in Example 2.

Example 2: Fig. 2(a) shows a four-node network built from a Gaussian broadcast channel followed by a real additive multiple-access channel. The two channels are independent; i.e.,

$$p(y^{(2)}, y^{(3)}, y^{(4)}|x^{(1)}, x^{(2)}, x^{(3)}) = p(y^{(2)}, y^{(3)}|x^{(1)})p(y^{(4)}|x^{(2)}, x^{(3)}).$$

[1]We say that a network's component channels are independent when the inputs to the channels are distinct and the noise random processes in the channels are independent; a formal definition follows in Section III.

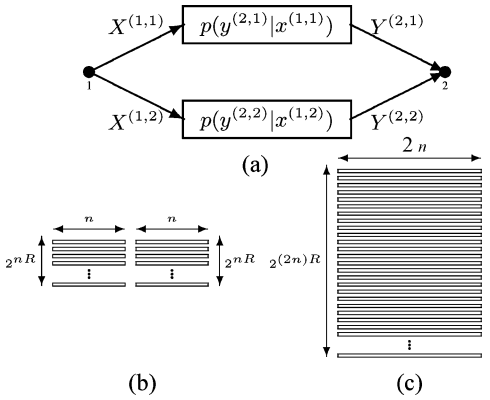

Fig. 1. Example 1 compares separate network and channel coding to joint network and channel coding across the network shown in (a). The separated strategy employs the pair of $(2^{nR}, n)$ channel codes shown in (b); each is used to reliably transmit nR bits over n uses of a single channel. The joint coding strategy employs the single $(2^{(2n)R}, 2n)$ channel code shown in (c); this code is used to reliably transmit information across n uses of the pair of channels. The joint coding strategy achieves twice the error exponent by operating each channel at roughly twice its capacity.

The broadcast channel has power constraint $E[(X^{(1)})^2] \leq P$ and channel outputs $Y^{(2)} = X^{(1)} + Z^{(2)}$ and $Y^{(3)} = X^{(1)} + Z^{(3)}$. Here, $Z^{(2)}$ and $Z^{(3)}$ are statistically dependent mean-0, variance-N random variables with $Z^{(2)} = -Z^{(3)}$, and P and N are positive, real-valued constants. The multiple-access channel has power constraints $E[(X^{(2)})^2], E[(X^{(3)})^2] \leq P + N$ at each transmitter and output $Y^{(4)} = X^{(2)} + X^{(3)}$. We consider a single unicast connection, where node 1 wishes to reliably transmit information to node 4. If we channel code to make each channel reliable and then apply network coding, the achievable rate cannot exceed the broadcast channel's maximal sum rate

$$\max_{\alpha \in [0,1]} \left[\frac{1}{2} \log \left(1 + \frac{\alpha P}{N} \right) + \frac{1}{2} \log \left(1 + \frac{(1-\alpha)P}{\alpha P + N} \right) \right] = \frac{1}{2} \log \left(1 + \frac{P}{N} \right).$$

The network's unicast capacity is infinite since nodes 2 and 3 can simply retransmit their channel outputs uncoded to give output $Y^{(4)} = (X^{(1)} + Z^{(2)}) + (X^{(1)} + Z^{(3)}) = 2X^{(1)}$ at node 4. ■

Example 2 shows that the gap between the capacity of a network and the maximal rate achievable on that network using separate network and channel codes can be very large. It is tempting to believe that the observed gap results from the example's unusual noise characteristics and therefore to hope that the penalty for separate network and channel coding might still be small for real networks since such statistics are unlikely to occur. Unfortunately, the penalty for separate network and channel coding is sometimes high even for networks with independent noise at all receivers, as shown in Example 3.

Example 3: Fig. 2(b) shows an $(m + 2)$-node network constructed from a Gaussian broadcast channel and a real additive multiple-access channel. The broadcast channel has power constraint $E[(X^{(1)})^2] \leq P$ and channel outputs $Y^{(i)} = X^{(1)} + Z^{(i)}$, $i \in \{2, \ldots, m+1\}$, where $Z^{(i)}$ are independent, mean-0, variance-N Gaussian random variables, and P and N are real-valued positive constants. The multiple-access

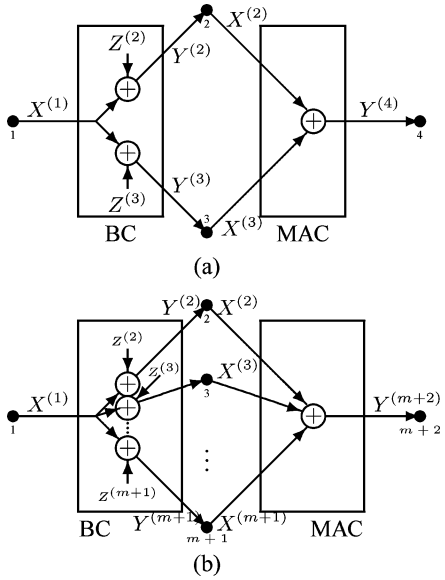

Fig. 2. Separate network and channel coding fails to achieve the unicast capacity of (a) a four-node network with dependent noise at the receivers of the broadcast channel and (b) an $(m+2)$-node network with independent noise at the receivers of the broadcast channel.

channel has power constraint $E[(X^{(i)})^2] \leq P+N$ at each transmitter $i \in \{2, \ldots, m+1\}$ and output $Y^{(m+2)} = \sum_{i=2}^{m+1} X^{(i)}$. We wish to establish a single unicast connection from node 1 to node $(m+2)$. The maximal unicast rate using separate network and channel codes is bounded by the broadcast channel's maximal sum rate

$$\max_{\alpha_2, \ldots, \alpha_{m+1}} \sum_{i=2}^{m+1} \frac{1}{2} \log \left(1 + \frac{\alpha_i P}{\sum_{j=2}^{i-1} \alpha_j P + N} \right) = \frac{1}{2} \log \left(1 + \frac{P}{N} \right);$$

the maximization is taken over all $(\alpha_2, \ldots, \alpha_{m+1})$ with $\alpha_i \geq 0$ for all i and $\sum_{i=2}^{m+1} \alpha_i = 1$. The unicast capacity of the network is at least

$$\frac{1}{2} \log \left(1 + \frac{mP}{N} \right)$$

since nodes 2 through $m+1$ can simply retransmit their channel outputs uncoded to give output

$$Y^{(m+2)} = \sum_{i=2}^{m+1} (X^{(1)} + Z^{(i)}) = mX^{(1)} + \sum_{i=2}^{m+1} Z^{(i)}$$

which is a Gaussian channel with power $E[(mX^{(1)})^2] = m^2 P$ and noise variance $E[(\sum_{i=2}^{m+1} Z^{(i)})^2] = mN$. ■

Examples 2 and 3 show that the capacity of a channel can vastly underestimate the rate at which that channel can operate in a larger network. In a channel with multiple receivers, this phenomenon arises since the channel capacity requires each receiver to decode reliably using only its received channel output; in a larger network, nodes that receive evidence about the channel outputs at multiple receivers may be able to reliably decode at a higher rate. In a channel with multiple transmitters, the capacity requires each transmitter to send an independent transmission; in a larger network, it may be possible to establish dependence at the channel inputs and deliver a higher rate.

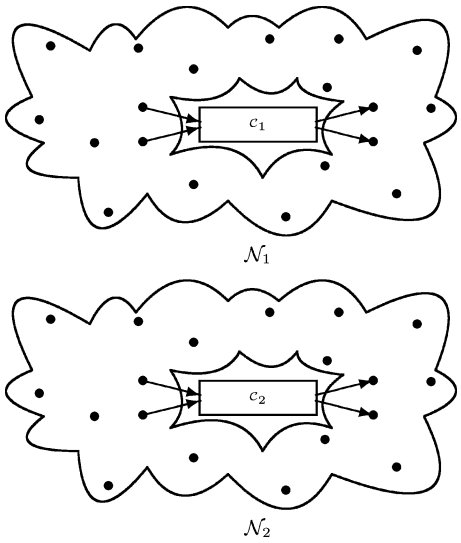

Fig. 3. Networks $\mathcal{N}_1$ and $\mathcal{N}_2$ are identical except that $\mathcal{N}_2$ replaces channel $\mathcal{C}_1$ by channel $\mathcal{C}_2$.

The preceding examples also suggest that a channel's optimal behavior can vary enormously depending on the network in which that channel is employed. To capture the full range of behaviors, we introduce the concept of upper and lower bounding channel models, which is roughly as follows. Consider a pair of channels $\mathcal{C}_1$ and $\mathcal{C}_2$. Let $\mathcal{N}_1$ be an arbitrary network containing independent channel $\mathcal{C}_1$, and let $\mathcal{N}_2$ be another network that is identical to $\mathcal{N}_1$ except that it replaces independent channel $\mathcal{C}_1$ by independent channel $\mathcal{C}_2$. (See Fig. 3 for an illustration; formal definitions follow in Section III.) If the capacity of $\mathcal{N}_1$ is a subset of that for $\mathcal{N}_2$ for all possible $\mathcal{N}_1$ and $\mathcal{N}_2$, then $\mathcal{C}_1$ is a lower bounding model for $\mathcal{C}_2$ (or, equivalently, $\mathcal{C}_2$ is an upper bounding model for $\mathcal{C}_1$). When $\mathcal{C}_1$ is both an upper and a lower bounding model for $\mathcal{C}_2$, we say that $\mathcal{C}_1$ and $\mathcal{C}_2$ are equivalent.

By the given definition, proving that $\mathcal{C}_1$ is a lower bounding model for $\mathcal{C}_2$ requires demonstrating that any connections that can be supported on any network containing $\mathcal{C}_1$ can still be supported if we replace $\mathcal{C}_1$ by $\mathcal{C}_2$. This is challenging both because the bounds must apply for all networks containing the channel and because they must apply to all combinations of connections across each network.

Since sequentially considering all possible networks and all possible connections is infeasible, we propose an alternative strategy for proving bounding relationships between channels. We prove that channel $\mathcal{C}_2$ is an upper bounding model for channel $\mathcal{C}_1$ by proving that channel $\mathcal{C}_2$ can emulate channel $\mathcal{C}_1$ to sufficient accuracy that any code that can be operated across a network $\mathcal{N}_1$ containing $\mathcal{C}_1$ can be operated with similar error probability across the network $\mathcal{N}_2$ that replaces $\mathcal{C}_1$ with $\mathcal{C}_2$. This proves that any rate that is achievable on $\mathcal{N}_1$ is also achievable on $\mathcal{N}_2$, which demonstrates the desired relationship.

Our aim in deriving bounding channel models is to simplify the calculation of capacities. We therefore focus on upper and lower bounding models comprised of noiseless bit pipes. For example, we seek to upper bound an arbitrary point-to-point channel by a bit pipe of the smallest possible capacity. The value of such a result is that it allows us to find new bounds on capacities of networks of noisy point-to-point channels in terms of the network coding capacities of networks of noiseless bit pipes. While the latter problem is not solved in general, a variety of computational tools for bounding these capacities are available. (See, for example, [4]–[6].) This work enables the application of these tools to find capacities for networks of noisy channels.

Section III describes the problem setup. Section IV introduces stacked networks and relates their capacities to standard network capacities. Section V summarizes the main results of Part I, which include identical upper and lower bounding models for point-to-point channels that together prove the equivalence between a point-to-point channel and a noiseless bit pipe of the same capacity. Section VI contains the central proofs. Section VII gives a summary and conclusions for Part I. Part II generalizes the results to derive error-free models for multiterminal channels. These tools are useful for the development of computational tools for bounding network capacities.

III. The Setup

Our notation is similar to that of Cover and Thomas [18, Sec. 15.10]. Network $\mathcal{N}$ has m nodes, described by vertex set $\mathcal{V} = \{1, \ldots, m\}$. Each node transmits an input random variable $X^{(v)} \in \mathcal{X}^{(v)}$ and receives an output random variable $Y^{(v)} \in \mathcal{Y}^{(v)}$ at each time step. The alphabets $\mathcal{X}^{(v)}$ and $\mathcal{Y}^{(v)}$ may be discrete or continuous and scalar or vector. For example, if node v transmits information over k Gaussian channels, then $\mathcal{X}^{(v)} = \mathbb{R}^k$. We use $\mathbf{X} = (X^{(v)} : v \in \mathcal{V})$ and $\mathbf{Y} = (Y^{(v)} : v \in \mathcal{V})$ to denote the collections of all network inputs and outputs. At time t, node v transmits $X_t^{(v)}$ and receives $Y_t^{(v)}$; $\mathbf{X}_t$ and $\mathbf{Y}_t$ denote the full vectors of time-t transmissions and receptions, respectively. By assumption, the network is memoryless and time-invariant, and its behavior is characterized by a conditional probability distribution.[2] Thus

$$p(\mathbf{y}_t|\mathbf{x}^t, \mathbf{y}^{t-1}) = p(\mathbf{y}_t|\mathbf{x}_t)$$

for all t, and any network $\mathcal{N}$ is defined by its corresponding triple

$$\mathcal{N} = \left(\prod_{v=1}^{m} \mathcal{X}^{(v)}, p(\mathbf{y}|\mathbf{x}), \prod_{v=1}^{m} \mathcal{Y}^{(v)} \right)$$

and the causality constraint that $X_t^{(v)}$ is a function only of past network outputs $(Y_1^{(v)}, \ldots, Y_{t-1}^{(v)})$ at node v and the node's outgoing messages, defined next.

For any $v \in \mathcal{V}$ and $U \subseteq \mathcal{V} \setminus \{v\}$, let $C_{v,U}(\mathcal{N})$ be the single-source multicast capacity from v to U in $\mathcal{N}$. Let

$$\mathcal{M} \stackrel{\text{def}}{=} \{(v, U) : v \in \mathcal{V}, U \subseteq \mathcal{V} \setminus \{v\}, C_{v,U}(\mathcal{N}) > 0\}$$

denote the set of all possible multicast connections across network $\mathcal{N}$. A code of blocklength n operates the network over n time steps with the goal of communicating, for each $(v, U) \in \mathcal{M}$, message

$$W^{(\{v\}\to U)} \in \mathcal{W}^{(\{v\}\to U)} \stackrel{\text{def}}{=} \{1, \ldots, 2^{nR^{(\{v\}\to U)}}\}$$

[2]The last assumption is restrictive only for channels with continuous output alphabets, where it implies that we consider only channels for which the conditional probability density function exists.

from source node v to all of the sink nodes $u \in U$. The messages $(W^{(\{v\}\to U)} : (v,U) \in \mathcal{M})$ are independent and uniformly distributed by assumption.[3] We use $W^{(\{v\}\to *)} \in \mathcal{W}^{(\{v\}\to *)}$ to denote the vector of messages transmitted from node v and $W \in \mathcal{W}$ to denote the vector of all messages; i.e.,

$$\begin{aligned} W^{(\{v\}\to *)} &\stackrel{\text{def}}{=} (W^{(\{v\}\to U)}: U \subseteq \mathcal{V}\setminus\{v\},\, C_{v,U}(\mathcal{N}) > 0) \\ \mathcal{W}^{(\{v\}\to *)} &\stackrel{\text{def}}{=} \prod_{U: U\subseteq\mathcal{V}\setminus\{v\},\, C_{v,U}(\mathcal{N})>0} \mathcal{W}^{(\{v\}\to U)} \\ W &\stackrel{\text{def}}{=} (W^{(\{v\}\to *)}: v \in \mathcal{V}) \\ \mathcal{W} &\stackrel{\text{def}}{=} \prod_{v\in\mathcal{V}} \mathcal{W}^{(\{v\}\to *)}. \end{aligned}$$

The constant $R^{(\{v\}\to U)}$ is called the multicast rate from v to U, and the at most $m2^{m-1}$-dimensional vector of multicast rates[4] is denoted by $\mathcal{R} \stackrel{\text{def}}{=} (R^{(\{v\}\to U)} : (v,U) \in \mathcal{M})$.

Definition 1: Let a network

$$\mathcal{N} \stackrel{\text{def}}{=} \left(\prod_{v=1}^{m} \mathcal{X}^{(v)}, p(\mathbf{y}|\mathbf{x}), \prod_{v=1}^{m} \mathcal{Y}^{(v)}\right)$$

be given. A blocklength-n solution $\mathcal{S}(\mathcal{N})$ to $\mathcal{N}$ is a set of encoding functions

$$X_t^{(v)} : (\mathcal{Y}^{(v)})^{t-1} \times \mathcal{W}^{(\{v\}\to *)} \to \mathcal{X}^{(v)}$$

mapping $(Y_1^{(v)}, \ldots, Y_{t-1}^{(v)}, W^{(\{v\}\to *)})$ to $X_t^{(v)}$ for each $v \in \mathcal{V}$ and $t \in \{1,\ldots,n\}$ and a set of decoding functions

$$\widehat{W}^{(\{u\}\to V),v} : (\mathcal{Y}^{(v)})^n \times \mathcal{W}^{(\{v\}\to *)} \to \mathcal{W}^{(\{u\}\to V)}$$

mapping $(Y_1^{(v)}, \ldots, Y_n^{(v)}, W^{(\{v\}\to *)})$ to $\widehat{W}^{(\{u\}\to V),v}$ for each (u,V,v) with $(u,V) \in \mathcal{M}$ and $v \in V$. We use $\widehat{W} \neq W$ as notational shorthand for the event that one or more messages is decoded in error at one or more of its intended receivers (i.e., $\widehat{W} \neq W$ if and only if $\widehat{W}^{(\{u\}\to V),v} \neq W^{(\{u\}\to V)}$ for some (u,V,v) with $(u,V) \in \mathcal{M}$ and $v \in V$). The solution $\mathcal{S}(\mathcal{N})$ is called a $(\lambda,\mathcal{R})$-solution, denoted $(\lambda,\mathcal{R})$-$\mathcal{S}(\mathcal{N})$, if $(\log|\mathcal{W}^{(\{v\}\to U)}|)/n = R^{(\{v\}\to U)}$ for all $(v,U) \in \mathcal{M}$ and $\Pr(\widehat{W} \neq W) < \lambda$ using the specified encoding and decoding functions.

Definition 2: The capacity $\mathfrak{R}(\mathcal{N})$ of a network $\mathcal{N}$ is the closure of all rate vectors $\mathcal{R}$ such that for any $\lambda > 0$ and all n sufficiently large, there exists a $(\lambda,\mathcal{R})$-$\mathcal{S}(\mathcal{N})$ solution of blocklength n. We use $\text{int}(\mathfrak{R}(\mathcal{N}))$ to denote the interior of the capacity region $\mathfrak{R}(\mathcal{N})$.

Remark 1: The given definitions are sufficiently general to model a wide variety of memoryless networks, including the usual models for isolated memoryless point-to-point, broadcast, multiple-access, and interference channels. Using vector alphabets makes it possible to model MIMO channels and other systems where a single node transmits multiple network inputs or receives multiple channel outputs. Including a "no transmission" symbol, here denoted by ϕ, is useful both for handling transmitters that receive no channel outputs and receivers that transmit no channel inputs (in which case ϕ is the only symbol in the corresponding alphabet) and for accommodating problems where it is important to be able to embed a node's transmissions in a schedule that may or may not depend on the messages to be sent and the symbols that were received in the network (in which case ϕ is part of a larger alphabet). In all cases we assume that at each time t, random variables $X_t^{(v)}$ and $Y_t^{(v)}$ are given. ■

We say that network

$$\mathcal{N}_1 = \left(\prod_{v=1}^{m} \mathcal{X}^{(v)}, p(\mathbf{y}|\mathbf{x}), \prod_{v=1}^{m} \mathcal{Y}^{(v)}\right)$$

contains independent channel

$$\mathcal{C}_1 = \left(\prod_{v\in\mathcal{V}_1} \mathcal{X}^{(v,1)}, p_1(\mathbf{y}^{(*,1)}|\mathbf{x}^{(*,1)}), \prod_{v\in\mathcal{V}_1} \mathcal{Y}^{(v,1)}\right)$$

if $\mathcal{V}_1 \subseteq \mathcal{V}$,

$$\mathcal{X}^{(v)} = \begin{cases} \mathcal{X}^{(v,0)} \times \mathcal{X}^{(v,1)}, & \text{if } v \in \mathcal{V}_1 \\ \mathcal{X}^{(v,0)}, & \text{if } v \notin \mathcal{V}_1, \end{cases}$$

$$\mathcal{Y}^{(v)} = \begin{cases} \mathcal{Y}^{(v,0)} \times \mathcal{Y}^{(v,1)}, & \text{if } v \in \mathcal{V}_1 \\ \mathcal{Y}^{(v,0)}, & \text{if } v \notin \mathcal{V}_1 \end{cases}$$

and

$$p(\mathbf{y}|\mathbf{x}) = p_0(\mathbf{y}^{(*,0)}|\mathbf{x}^{(*,0)}) p_1(\mathbf{y}^{(*,1)}|\mathbf{x}^{(*,1)})$$

for all $\mathbf{x} = (\mathbf{x}^{(*,0)}, \mathbf{x}^{(*,1)})$ and $\mathbf{y} = (\mathbf{y}^{(*,0)}, \mathbf{y}^{(*,1)})$, where

$$\begin{aligned} \mathbf{x}^{(*,0)} &= (x^{(v,0)} : v \in \mathcal{V}), & \mathbf{x}^{(*,1)} &= (x^{(v,1)} : v \in \mathcal{V}_1) \\ \mathbf{y}^{(*,0)} &= (y^{(v,0)} : v \in \mathcal{V}), & \mathbf{y}^{(*,1)} &= (y^{(v,1)} : v \in \mathcal{V}_1). \end{aligned}$$

The channel

$$\mathcal{C}_0 = \left(\prod_{v=1}^{m} \mathcal{X}^{(v,0)}, p_0(\mathbf{y}^{(*,0)}|\mathbf{x}^{(*,0)}), \prod_{v=1}^{m} \mathcal{Y}^{(v,0)}\right)$$

captures the stochastic behavior of the remainder of the network. We therefore describe the network as

$$\mathcal{N}_1 = \mathcal{C}_0 \times \mathcal{C}_1.$$

Remark 2: As noted above, we set $\mathcal{X}^{(v)} = \{\phi\}$ when node v transmits no network inputs and $\mathcal{Y}^{(v)} = \{\phi\}$ when node v receives no network outputs. The same convention applies in channels. Where notationally convenient, we drop random variables with alphabet $\{\phi\}$ from our channel and network definitions. For example, a binary, memoryless point-to-point channel is defined either as

$$\mathcal{C} = (\mathcal{X}^{(1)} \times \mathcal{X}^{(2)}, p(y^{(1)}, y^{(2)}|x^{(1)}, x^{(2)}), \mathcal{Y}^{(1)} \times \mathcal{Y}^{(2)})$$

with $\mathcal{X}^{(1)} = \mathcal{Y}^{(2)} = \{0,1\}$, $\mathcal{X}^{(2)} = \mathcal{Y}^{(1)} = \{\phi\}$, and $p(y^{(1)}, y^{(2)}|x^{(1)}, x^{(2)}) = p(y^{(2)}|x^{(1)})\delta(y^{(1)} = \phi)$ or, equivalently, as

$$\mathcal{C} = (\mathcal{X}^{(1)}, p(y^{(2)}|x^{(1)}), \mathcal{Y}^{(2)})$$

with $\mathcal{X}^{(1)} = \mathcal{Y}^{(2)} = \{0,1\}$. ■

[3]The proof also goes through if the same message is available at more than one node in the network.

[4]The dimension of $\mathcal{R}$ is really at most $m(2^{m-1}-1)$ since a multicast with receiver set $U = \{\}$ has no meaning.

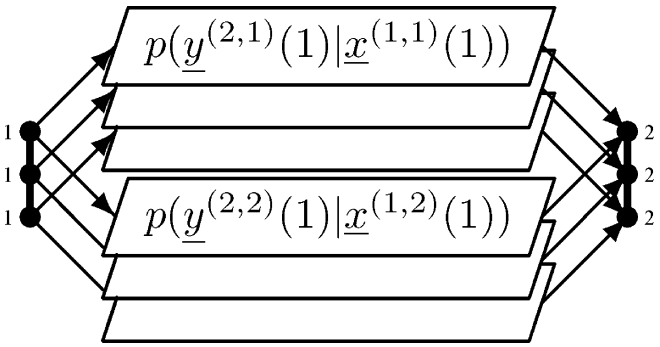

Fig. 4. The 3-fold stacked network $\underline{\mathcal{N}}$ for the network $\mathcal{N}$ in Fig. 1(a).

Definition 3: Let a pair of channels

$$\mathcal{C}_1 = \left(\prod_{v=1}^{k_1} \mathcal{X}^{(v,1)}, p_1(\mathbf{y}^{(*,1)}|\mathbf{x}^{(*,1)}), \prod_{v=1}^{k_1} \mathcal{Y}^{(v,1)} \right)$$

$$\mathcal{C}_2 = \left(\prod_{v=1}^{k_2} \mathcal{X}^{(v,2)}, p_2(\mathbf{y}^{(*,2)}|\mathbf{x}^{(*,2)}), \prod_{v=1}^{k_2} \mathcal{Y}^{(v,2)} \right)$$

be given, where $\mathbf{x}^{(*,i)} = (x^{(v,i)} : v \in \{1, \ldots, k_i\})$, $\mathbf{y}^{(*,i)} = (y^{(v,i)} : v \in \{1, \ldots, k_i\})$. We say that channel $\mathcal{C}_1$ is a lower bounding model for channel $\mathcal{C}_2$, written

$$\mathcal{C}_1 \subseteq \mathcal{C}_2$$

if $\mathfrak{R}(\mathcal{C}_0 \times \mathcal{C}_1) \subseteq \mathfrak{R}(\mathcal{C}_0 \times \mathcal{C}_2)$ for all channels $\mathcal{C}_0$; that is, the capacity region of a network that contains independent channel $\mathcal{C}_1$ is never diminished if we replace $\mathcal{C}_1$ by independent channel $\mathcal{C}_2$. We say that channel $\mathcal{C}_2$ is an upper bounding model for $\mathcal{C}_1$, written

$$\mathcal{C}_2 \supseteq \mathcal{C}_1$$

if $\mathcal{C}_1$ is a lower bounding model for $\mathcal{C}_2$ (i.e., $\mathcal{C}_1 \subseteq \mathcal{C}_2$). We say that channels $\mathcal{C}_1$ and $\mathcal{C}_2$ are equivalent, written

$$\mathcal{C}_1 = \mathcal{C}_2$$

if $\mathcal{C}_1$ is both a lower bounding model and an upper bounding model for $\mathcal{C}_2$ (i.e., $\mathcal{C}_1 \subseteq \mathcal{C}_2$ and $\mathcal{C}_1 \supseteq \mathcal{C}_2$).

IV. Stacked Networks and Stacked Solutions

As noted briefly in Section II, our strategy for showing that a channel $\mathcal{C}_2$ is an upper bounding model for another channel $\mathcal{C}_1$ is to show that $\mathcal{C}_2$ can emulate $\mathcal{C}_1$ to sufficient accuracy that any code built for a network $\mathcal{N}_1 = \mathcal{C}_0 \times \mathcal{C}_1$ can be run across network $\mathcal{N}_2 = \mathcal{C}_0 \times \mathcal{C}_2$ with similar error probability. In the arguments that follow in Section V, we first fix a solution $\mathcal{S}(\mathcal{N}_1)$ on $\mathcal{N}_1$ and then build a code to emulate the typical behavior of channel $\mathcal{C}_1$ under this solution. Since $\mathcal{S}(\mathcal{N}_1)$ may employ memory and establish different distributions across channel $\mathcal{C}_1$ at different times, the channel input and output are not necessarily independent and identically distributed (i.i.d.) across time. As a result, this work applies typicality and emulation arguments not across time but instead across a stack of parallel instances of the network, as shown in Fig. 4.

We introduce the resulting "stacked network" and its solutions below. As we show later in this section, a network and its corresponding stacked network have the same capacities. Further, we show that any solution that can be operated on a stacked network can also be operated on the original network with the same error probability and rate. Finally, the stacked network simplifies later arguments because it establishes the i.i.d. structure used in the emulation arguments employed in Section V. It also avoids the decoding delays associated with block coding across time, which are troublesome both in networks with cycles and in networks with synchronized transmissions.

Given a network

$$\mathcal{N} = \left(\prod_{v=1}^{m} \mathcal{X}^{(v)}, p(\mathbf{y}|\mathbf{x}), \prod_{v=1}^{m} \mathcal{Y}^{(v)} \right)$$

on vertex set $\mathcal{V} = \{1, \ldots, m\}$ and an integer $N \geq 1$, the N-fold stacked network $\underline{\mathcal{N}}$ contains N copies of $\mathcal{N}$. Thus, $\underline{\mathcal{N}}$ has mN nodes described by the multiset $\underline{\mathcal{V}}$ that contains N copies of each $v \in \mathcal{V}$.[5] We visualize $\underline{\mathcal{N}}$ as a stack with N layers, each of which contains one copy of each vertex $v \in \mathcal{V}$. The number of layers (N) in a stacked network $\underline{\mathcal{N}}$ is specified by context; in later typicality and coding arguments, N is allowed to grow without bound.

We carry over notation from network $\mathcal{N}$ to stacked network $\underline{\mathcal{N}}$ by underlining the variable names. Argument ℓ designates the variables in layer ℓ of the stack. Thus for each $v \in \mathcal{V}$, the copy of node v in layer ℓ transmits an input $\underline{X}^{(v)}(\ell)$ and receives an output $\underline{Y}^{(v)}(\ell)$ in each time step; the vectors of inputs and outputs over the N layers of the stack are $\underline{X}^{(v)} \stackrel{\text{def}}{=} (\underline{X}^{(v)}(\ell) : \ell \in \{1, \ldots, N\})$ and $\underline{Y}^{(v)} \stackrel{\text{def}}{=} (\underline{Y}^{(v)}(\ell) : \ell \in \{1, \ldots, N\})$, and their alphabets are $\underline{\mathcal{X}}^{(v)} \stackrel{\text{def}}{=} (\mathcal{X}^{(v)})^N$ and $\underline{\mathcal{Y}}^{(v)} \stackrel{\text{def}}{=} (\mathcal{Y}^{(v)})^N$. Random variables $\underline{\mathbf{X}} \stackrel{\text{def}}{=} (\underline{X}^{(v)} : v \in \mathcal{V})$ and $\underline{\mathbf{Y}} \stackrel{\text{def}}{=} (\underline{Y}^{(v)} : v \in \mathcal{V})$ describe the transmitted and received values for all vertices. The conditional distribution on all outputs from the stacked network given all inputs to the stacked network is given by

$$p(\underline{\mathbf{y}}|\underline{\mathbf{x}}) = \prod_{\ell=1}^{N} p\left(\underline{\mathbf{y}}(\ell) \middle| \underline{\mathbf{x}}(\ell)\right)$$

where $\underline{\mathbf{X}}(\ell) \stackrel{\text{def}}{=} (\underline{X}^{(v)}(\ell) : v \in \mathcal{V})$ and $\underline{\mathbf{Y}}(\ell) \stackrel{\text{def}}{=} (\underline{Y}^{(v)}(\ell) : v \in \mathcal{V})$ are the stacked network inputs and outputs in layer ℓ. The causality constraint on node operations restricts the stacked network input $\underline{X}_t^{(v)}$ to be a function only of past stacked network outputs $(\underline{Y}_1^{(v)}, \ldots, \underline{Y}_{t-1}^{(v)})$ and the messages from the copies of v, as defined below.

A code of blocklength n operates the stacked network over n time steps with the goal of communicating, for each $(v, U) \in \mathcal{M}$, an independent and uniformly distributed message $\underline{W}^{(\{v\} \to U)}(\ell)$ in each layer ℓ. Let

$$\underline{W}^{(\{v\} \to U)} \stackrel{\text{def}}{=} (\underline{W}^{(\{v\} \to U)}(\ell) : \ell \in \{1, \ldots, N\})$$

$$\underline{W}^{(\{v\} \to *)} \stackrel{\text{def}}{=} (\underline{W}^{(\{v\} \to U)} : U \subseteq \mathcal{V} \setminus \{v\},\ C_{v,U}(\mathcal{N}) > 0)$$

$$\underline{\mathcal{W}}^{(\{v\} \to U)} \stackrel{\text{def}}{=} (\mathcal{W}^{(\{v\} \to U)})^N$$

$$\underline{\mathcal{W}}^{(\{v\} \to *)} \stackrel{\text{def}}{=} (\mathcal{W}^{(\{v\} \to *)})^N$$

designate the messages from v to U, all messages from v, and the corresponding alphabets. When

[5] A multiset is a generalization of a set that allows multiple copies of the same element. The distinction between the set $\mathcal{V}$ used to describe the vertices of $\mathcal{N}$ and the multiset $\underline{\mathcal{V}}$ required to describe the vertices of $\underline{\mathcal{N}}$ implies that a stacked network is not a network, and therefore new definitions are required.

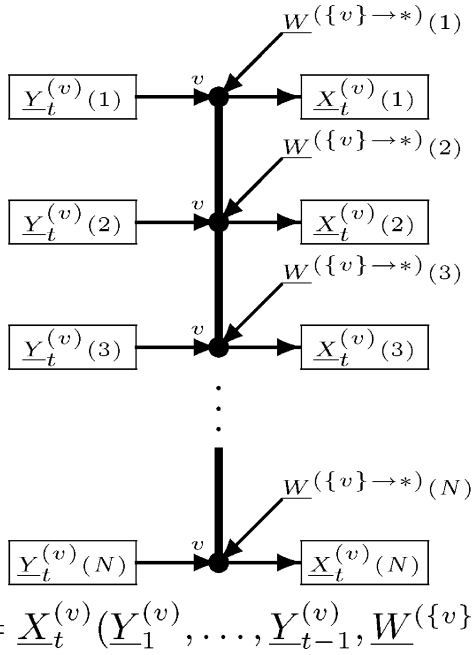

Fig. 5. The operation of $\mathcal{S}(\underline{\mathcal{N}})$ in $\underline{\mathcal{N}}$. The illustration shows the time-t stacked network outputs and inputs at the N copies of node v in $\underline{\mathcal{N}}$; input $\underline{X}_t^{(v)}$ is a function only of prior outputs $(\underline{Y}_1^{(v)}, \ldots, \underline{Y}_{t-1}^{(v)})$ and outgoing messages $\underline{W}^{(\{v\}\to *)}$.

$\mathcal{W}^{(\{v\}\to U)} = \{1, \ldots, 2^{nR^{(\{v\}\to U)}}\}$ in $\mathcal{N}$, $\underline{\mathcal{W}}^{(\{v\}\to U)} = \{1, \ldots, 2^{nR^{(\{v\}\to U)}}\}^N$ in $\underline{\mathcal{N}}$. We therefore define the rate $R^{(\{v\}\to U)}$ for N-fold stacked network $\underline{\mathcal{N}}$ as $(\log|\underline{\mathcal{W}}^{(\{v\}\to U)}|)/(nN)$; this definition makes the rates in $\mathcal{N}$ and $\underline{\mathcal{N}}$ consistent.

While both the transmitted messages and the conditional distribution $p(\underline{\mathbf{y}}|\underline{\mathbf{x}})$ are independent across the layers of the stack, the following definition of a network solution allows both the node encoders and the node decoders for all copies of a node v to work together across the layers. Fig. 5, which shows the operation of a stacked network's encoders for some $v \in \mathcal{V}$, highlights this potential collaboration using vertical lines to connect all copies of node v. The solution definition also specifies the error criterion for coding: a solution is successful only if every node succeeds in decoding all of its incoming messages. This becomes difficult as the number of layers (and thus the number of nodes and messages) grows without bound.

Definition 4: Let a network

$$\mathcal{N} \stackrel{\text{def}}{=} \left(\prod_{v=1}^{m} \mathcal{X}^{(v)}, p(\mathbf{y}|\mathbf{x}), \prod_{v=1}^{m} \mathcal{Y}^{(v)}\right)$$

be given. Let $\underline{\mathcal{N}}$ be the N-fold stacked network for $\mathcal{N}$. A blocklength-n solution $\mathcal{S}(\underline{\mathcal{N}})$ to stacked network $\underline{\mathcal{N}}$ is a set of encoding and decoding functions

$$\underline{X}_t^{(v)} : (\underline{\mathcal{Y}}^{(v)})^{t-1} \times \underline{\mathcal{W}}^{(\{v\}\to *)} \to \underline{\mathcal{X}}^{(v)}$$
$$\widehat{\underline{W}}^{(\{u\}\to V),v} : (\underline{\mathcal{Y}}^{(v)})^{n} \times \underline{\mathcal{W}}^{(\{v\}\to *)} \to \underline{\mathcal{W}}^{(\{u\}\to V)}$$

mapping $(\underline{Y}_1^{(v)}, \ldots, \underline{Y}_{t-1}^{(v)}, \underline{W}^{(\{v\}\to *)})$ to $\underline{X}_t^{(v)}$ for each $t \in \{1, \ldots, n\}$ and $v \in \mathcal{V}$ and mapping $(\underline{Y}_1^{(v)}, \ldots, \underline{Y}_n^{(v)}, \underline{W}^{(\{v\}\to *)})$ to $\widehat{\underline{W}}^{(\{u\}\to V),v}$ for each (u, V, v) with $(u, V) \in \mathcal{M}$ and $v \in V$. We use $\widehat{\underline{W}} \neq \underline{W}$ to designate the event that one or more messages is decoded in error at one or more of its intended receivers (i.e., $\widehat{\underline{W}} \neq \underline{W}$ if and only if $\widehat{\underline{W}}^{(\{u\}\to V),v} \neq \underline{W}^{(\{u\}\to V)}$ for some (u, V, v) with $(u, V) \in \mathcal{M}$ and $v \in V$). The solution $\mathcal{S}(\underline{\mathcal{N}})$ is called a $(\lambda, \mathcal{R})$-solution for $\underline{\mathcal{N}}$, denoted $(\lambda, \mathcal{R})$-$\mathcal{S}(\underline{\mathcal{N}})$, if $(\log|\underline{\mathcal{W}}^{(\{v\}\to U)}|)/(nN) = R^{(\{v\}\to U)}$ for all $(v, U) \in \mathcal{M}$ and $\Pr(\widehat{\underline{W}} \neq \underline{W}) < \lambda$ using the specified encoding and decoding functions.

Definition 5: The capacity $\mathfrak{R}(\underline{\mathcal{N}}) \subset \mathbb{R}_+^{|\mathcal{M}|}$ of stacked networks $\underline{\mathcal{N}}$ is the closure of all rate vectors $\mathcal{R}$ such that a $(\lambda, \mathcal{R})$-$\mathcal{S}(\underline{\mathcal{N}})$ solution exists for any $\lambda > 0$ and all N sufficiently large.

Lemma 1: For any network $\mathcal{N}$, $\mathfrak{R}(\mathcal{N}) = \mathfrak{R}(\underline{\mathcal{N}})$.

Proof:

$\mathfrak{R}(\mathcal{N}) \subseteq \mathfrak{R}(\underline{\mathcal{N}})$*:* Let $\mathcal{R} \in \text{int}(\mathfrak{R}(\mathcal{N}))$. Then for any $\lambda \in (0, 1]$ and any $N \geq 1$, there exists a $(\lambda/N, \mathcal{R})$-$\mathcal{S}(\mathcal{N})$ solution for network $\mathcal{N}$.[6] Running $\mathcal{S}(\mathcal{N})$ independently in each layer of N-fold stacked network $\underline{\mathcal{N}}$ yields a rate-$\mathcal{R}$ solution $\mathcal{S}(\underline{\mathcal{N}})$ for $\underline{\mathcal{N}}$. The error probability for $\mathcal{S}(\underline{\mathcal{N}})$ is less than or equal to λ by the union bound. Thus $\mathcal{R} \in \mathfrak{R}(\underline{\mathcal{N}})$, and the result follows from the closure in the definition of $\mathfrak{R}(\underline{\mathcal{N}})$.

$\mathfrak{R}(\mathcal{N}) \supseteq \mathfrak{R}(\underline{\mathcal{N}})$*:* Let $\mathcal{R} \in \text{int}(\mathfrak{R}(\underline{\mathcal{N}}))$. Then for any $\lambda \in (0, 1]$ and some N sufficiently large, there exists a $(\lambda, \mathcal{R})$-$\mathcal{S}(\underline{\mathcal{N}})$ solution to N-fold stacked network $\underline{\mathcal{N}}$. Let n be the blocklength of $\mathcal{S}(\underline{\mathcal{N}})$. We use $\mathcal{S}(\underline{\mathcal{N}})$ to build a blocklength-nN $(\lambda, \mathcal{R})$-$\mathcal{S}(\mathcal{N})$ solution for $\mathcal{N}$. Fig. 6 shows how this is done using the solution $\mathcal{S}(\underline{\mathcal{N}})$ from Fig. 5. Roughly, $\mathcal{S}(\mathcal{N})$ breaks each message $W^{(\{v\}\to U)} \in \{1, \ldots, 2^{(nN)R^{(\{v\}\to U)}}\}$ into N sub-messages $W^{(\{v\}\to U)}(1), \ldots, W^{(\{v\}\to U)}(N) \in \{1, \ldots, 2^{nR^{(\{v\}\to U)}}\}$ and runs $\mathcal{S}(\underline{\mathcal{N}})$ on these sub-messages as if they were the messages in the N layers of stacked network $\underline{\mathcal{N}}$. The network inputs transmitted at time t by the N copies of node v in $\underline{\mathcal{N}}$ are transmitted at times $(t-1)N+1, \ldots, tN$ by the single copy of node v in $\mathcal{N}$. This gives the desired result since the error probabilities and rates of $\mathcal{S}(\mathcal{N})$ on $\mathcal{N}$ and $\mathcal{S}(\underline{\mathcal{N}})$ on $\underline{\mathcal{N}}$ are equal. We think of this approach as "unraveling" the solution for a stacked network across time.

Formally, for each $(v, U) \in \mathcal{M}$, let

$$f^{(\{v\}\to U)} : \{1, \ldots, 2^{NnR^{(\{v\}\to U)}}\} \to \{1, \ldots, 2^{nR^{(\{v\}\to U)}}\}^N$$

be the natural one-to-one mapping from a single sequence of $NnR^{(\{v\}\to U)}$ bits to N consecutive subsequences each of $nR^{(\{v\}\to U)}$ bits. Let $g^{(\{v\}\to U)}$ be the inverse of $f^{(\{v\}\to U)}$. We use $f^{(\{v\}\to U)}$ to map messages from the message alphabet of the rate-$\mathcal{R}$, blocklength-Nn code $\mathcal{S}(\mathcal{N})$ to the message alphabet for the rate-$\mathcal{R}$, blocklength-n code $\mathcal{S}(\underline{\mathcal{N}})$ for N-fold stacked network $\underline{\mathcal{N}}$. The mapping is one-to-one since in each scenario the total number of bits transmitted from node v

[6]By the definition of capacity, for any $\mathcal{R} \in \text{int}(\mathfrak{R}(\mathcal{N}))$, we can find a rate-$\mathcal{R}$ network solution with arbitrarily small error probability. Since the capacity definition employs a closure, there is no such guarantee for $\mathcal{R} \in \mathfrak{R}(\mathcal{N}) \setminus (\text{int}(\mathfrak{R}(\mathcal{N})))$.

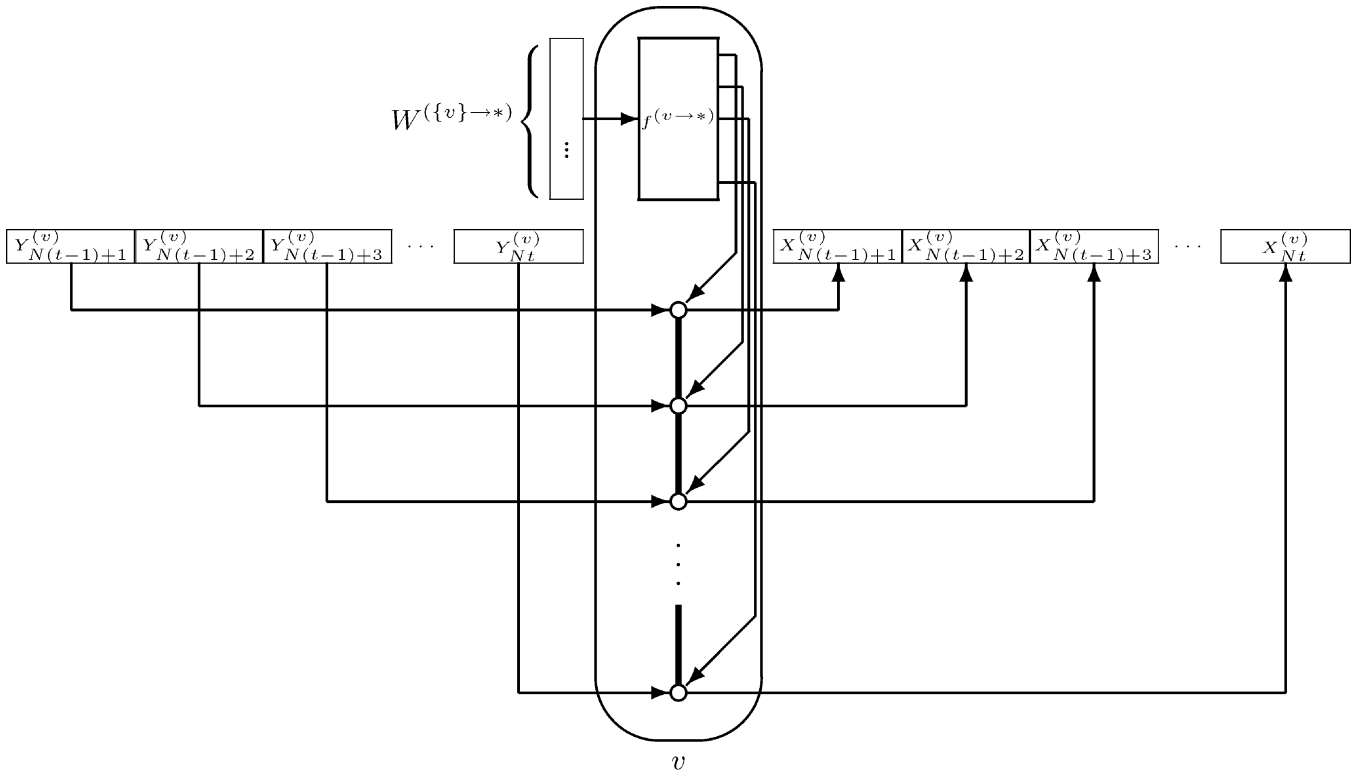

Fig. 6. A blocklength-n solution $\mathcal{S}(\underline{\mathcal{N}})$ for network $\underline{\mathcal{N}}$ can be operated over nN time steps in $\mathcal{N}$ with the same error probability and rate. The illustration shows this operation for the solution $\mathcal{S}(\underline{\mathcal{N}})$ for $\underline{\mathcal{N}}$ from Fig. 5. Each single node v in $\mathcal{N}$ performs the operations of all N copies of v in $\underline{\mathcal{N}}$ and transmits the resulting network inputs over N time steps. Vectors $(X^{(v)}_{N(t-1)+1},\ldots,X^{(v)}_{Nt})$ and $(Y^{(v)}_{N(t-1)+1},\ldots,Y^{(v)}_{Nt})$ in $\mathcal{N}$ play the roles of vectors $\underline{X}^{(v)}_t$ and $\underline{Y}^{(v)}_t$ in $\underline{\mathcal{N}}$.

to the nodes in U is $NnR^{(\{v\}\to U)}$.[7] For each $v \in \mathcal{V}$, let $f^{(\{v\}\to *)}(W^{(\{v\}\to *)}) = (f^{(\{v\}\to U)}(W^{(\{v\}\to U)}) : U \subseteq \mathcal{V}\setminus\{v\}, C_{v,U}(\mathcal{N}) > 0)$. For each $t \in \{1\ldots,n\}$, let

$$X^{(v)}(t) = (X^{(v)}_{(t-1)N+1},\ldots,X^{(v)}_{tN})$$
$$Y^{(v)}(t) = (Y^{(v)}_{(t-1)N+1},\ldots,Y^{(v)}_{tN})$$

denote the network inputs and outputs at node v for N consecutive time steps beginning at time $(t-1)N+1$. We define the solution $\mathcal{S}(\mathcal{N})$ as

$$X^{(v)}(t) = \underline{X}^{(v)}_t(Y^{(v)}(1),\ldots,Y^{(v)}(t-1), f^{(\{v\}\to *)}(W^{(\{v\}\to *)}))$$
$$\widehat{W}^{(\{u\}\to V),v} = g^{(\{u\}\to V)}(\underline{\widehat{W}}^{(\{u\}\to V),v}(Y^{(v)}(1),\ldots, Y^{(v)}(n), f^{(\{v\}\to *)}(W^{(\{v\}\to *)}))).$$

Solution $\mathcal{S}(\mathcal{N})$ satisfies the causality constraints and performs precisely the same mappings as $\mathcal{S}(\underline{\mathcal{N}})$. In addition, the conditional distribution on network outputs given network inputs for N consecutive transmissions on network $\mathcal{N}$ equals the conditional distribution on the N-dimensional vector of time-t outputs given the N-dimensional vector of time-t inputs for N-fold stacked network $\underline{\mathcal{N}}$. Thus, the solution $\mathcal{S}(\mathcal{N})$ achieves the same rate and error probability on $\mathcal{N}$ as $\mathcal{S}(\underline{\mathcal{N}})$ achieves on $\underline{\mathcal{N}}$. ■

[7]We here neglect rounding issues, which are asymptotically negligible. When $nR^{(\{v\}\to U)}$ is not an integer, the given strategy yields a solution $\mathcal{S}(\mathcal{N})$ of rate $N\lfloor nR^{(\{v\}\to U)}\rfloor/(Nn) = \lfloor nR^{(\{v\}\to U)}\rfloor/n$, which approaches $R^{(\{v\}\to U)}$ as n grows without bound.

The preceding theorem shows not only that $\mathfrak{R}(\mathcal{N})$ and $\mathfrak{R}(\underline{\mathcal{N}})$ are equal, but also that any point in $\mathfrak{R}(\underline{\mathcal{N}})$ can be achieved using the same single-layer solution independently in each layer of the stack. Such a solution is attractive because it establishes, for each time t, an i.i.d. distribution on the network inputs and outputs in the layers of the stack (i.e., $(\underline{\mathbf{X}}_t(1),\underline{\mathbf{Y}}_t(1)),\ldots,(\underline{\mathbf{X}}_t(N),\underline{\mathbf{Y}}_t(N))$ are i.i.d. for each time t). Unfortunately, this i.i.d. structure is not sufficient for the typicality arguments that follow. The problem is that the solution $\mathcal{S}(\mathcal{N})$ used to build a solution $\mathcal{S}(\underline{\mathcal{N}})$ in the proof of Lemma 1 varies both with the number of layers N and the desired error probability λ. Thus we cannot let N grow without bound for a *fixed* distribution $p_t(\mathbf{x},\mathbf{y})$ on $(\underline{\mathbf{X}}_t(\ell),\underline{\mathbf{Y}}_t(\ell))$.

Stacked solutions, illustrated in Fig. 7 and defined formally below, are structured solutions for stacked networks that are designed to achieve capacity while establishing the desired i.i.d. structure on network inputs and outputs across the layers of the stack. Like the codes used in the proof of Lemma 1, stacked solutions use the same solution $\mathcal{S}(\mathcal{N})$ in each layer of the stack. The difference is that they do not allow $\mathcal{S}(\mathcal{N})$ to vary with λ and N; instead, they fix $\mathcal{S}(\mathcal{N})$ and use channel coding to make the code reliable. For each $(v,U) \in \mathcal{M}$ a channel code maps $\underline{W}^{(\{v\}\to U)} \in \underline{\mathcal{W}}^{(\{v\}\to U)} \stackrel{\text{def}}{=} \{1,\ldots,2^{nR^{(\{v\}\to U)}}\}^N$ to $\underline{\tilde{W}}^{(\{v\}\to U)} \in \underline{\tilde{\mathcal{W}}}^{(\{v\}\to U)} \stackrel{\text{def}}{=} \{1,\ldots,2^{n\tilde{R}^{(\{v\}\to U)}}\}^N$ for some $\tilde{R}^{(\{v\}\to U)} > R^{(\{v\}\to U)}$. The network then delivers $\underline{\tilde{W}}^{(\{v\}\to U)}(1),\ldots,\underline{\tilde{W}}^{(\{v\}\to U)}(N)$ by running rate-$\tilde{\mathcal{R}}$ solution $\mathcal{S}(\mathcal{N})$ independently in each layer of the stack. For each (v,U,u) with $(v,U)\in\mathcal{M}$ and $u\in U$, the channel decoder for

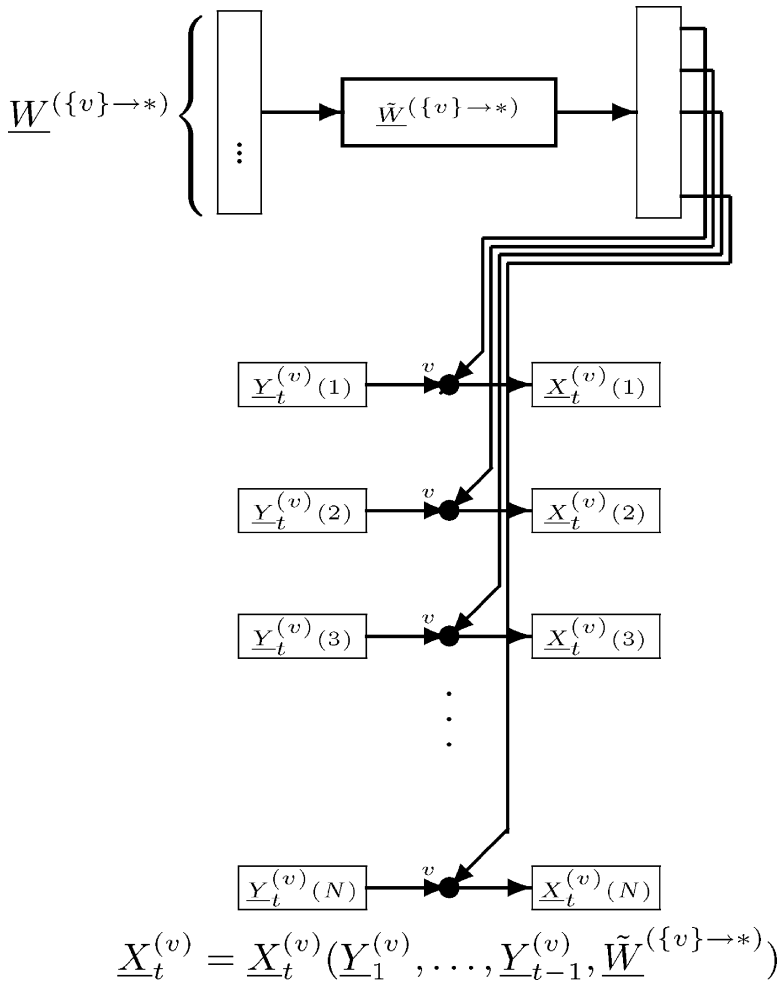

Fig. 7. A stacked solution $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ first channel codes each message and then applies the same solution $\mathcal{S}(\mathcal{N})$ independently in each layer of the stack. Solution $\mathcal{S}(\mathcal{N})$ operates at a rate $\tilde{\mathcal{R}}$ exceeding the rate $\mathcal{R}$ of solution $\underline{\mathcal{S}}(\underline{\mathcal{N}})$. The only collaboration between the copies of node v is in the channel coding operation; we highlight this fact by omitting the vertical lines that previously connected those nodes.

(v,U) at u maps the reproduction $\underline{\widehat{\tilde{W}}}^{(\{v\}\to U),u}$ of codeword $\underline{\tilde{W}}^{(\{v\}\to U)}$ to reconstruction $\underline{\widehat{W}}^{(\{v\}\to U),u}$ of $\underline{W}^{(\{v\}\to U)}$.

Definition 6: Let a network

$$\mathcal{N} \stackrel{\text{def}}{=} \left(\prod_{v=1}^{m} \mathcal{X}^{(v)}, p(\mathbf{y}|\mathbf{x}), \prod_{v=1}^{m} \mathcal{Y}^{(v)}\right)$$

be given. Let $\underline{\mathcal{N}}$ be the N-fold stacked network for $\mathcal{N}$. A blocklength-n stacked solution $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ to network $\underline{\mathcal{N}}$ is defined by channel codes

$$\underline{\tilde{W}}^{(\{v\}\to U)} : \underline{\mathcal{W}}^{(\{v\}\to U)} \to \underline{\tilde{\mathcal{W}}}^{(\{v\}\to U)}$$
$$\underline{\widehat{W}}^{(\{v\}\to U),u} : \underline{\tilde{\mathcal{W}}}^{(\{v\}\to U)} \to \underline{\mathcal{W}}^{(\{v\}\to U)}$$

for each $(v,U) \in \mathcal{V}$ and $u \in U$ and a single-layer solution $\mathcal{S}(\mathcal{N})$ with node encoders and decoders

$$X_t^{(v)} : (\mathcal{Y}^{(v)})^{t-1} \times \tilde{\mathcal{W}}^{(\{v\}\to *)} \to \mathcal{X}^{(v)}$$
$$\widehat{\tilde{W}}^{(\{u\}\to V),v} : (\mathcal{Y}^{(v)})^{n} \times \tilde{\mathcal{W}}^{(\{v\}\to *)} \to \tilde{\mathcal{W}}^{(\{u\}\to V)}.$$

The stacked solution channel codes each message as

$$\underline{\tilde{W}}^{(\{v\}\to U)} = \underline{\tilde{W}}^{(\{v\}\to U)}(\underline{W}^{(\{v\}\to U)})$$

and then applies $\mathcal{S}(\mathcal{N})$ independently in each layer of the stack to give

$$\underline{X}_t^{(v)}(\ell) = X_t^{(v)}(\underline{Y}_1^{(v)}(\ell), \ldots, \underline{Y}_{t-1}^{(v)}(\ell), \underline{\tilde{W}}^{(\{v\}\to *)}(\ell))$$
$$\underline{\widehat{\tilde{W}}}^{(\{u\}\to V),v}(\ell) = \widehat{\tilde{W}}^{(\{u\}\to V),v}(\underline{Y}_1^{(v)}(\ell), \ldots, \underline{Y}_n^{(v)}(\ell), \underline{\tilde{W}}^{(\{v\}\to *)}(\ell)).$$

The channel decoders reconstruct the messages as

$$\underline{\widehat{W}}^{(\{u\}\to V),v} = \underline{\widehat{W}}^{(\{u\}\to V),v}\left(\underline{\widehat{\tilde{W}}}^{(\{u\}\to V),v}\right).$$

The solution $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ is called a stacked $(\lambda,\mathcal{R})$-solution, denoted $(\lambda,\mathcal{R})$-$\underline{\mathcal{S}}(\underline{\mathcal{N}})$, if $(\log|\underline{\mathcal{W}}^{(\{v\}\to U)}|)/(nN) = R^{(\{v\}\to U)}$ for all $(v,U) \in \mathcal{M}$ and the specified mappings imply $\Pr\left(\underline{\widehat{W}} \neq \underline{W}\right) < \lambda$.

Theorem 2: Given any $\mathcal{R} \in \text{int}(\mathfrak{R}(\mathcal{N}))$, there exists a sequence of blocklength-n $(2^{-N\delta}, \mathcal{R})$-$\underline{\mathcal{S}}(\underline{\mathcal{N}})$ stacked solutions for some fixed $n \geq 1$ and $\delta > 0$ and all N sufficiently large.

Proof: Given any target rate $\mathcal{R} \in \text{int}(\mathfrak{R}(\mathcal{N}))$, fix some $\tilde{\mathcal{R}} \in \text{int}(\mathfrak{R}(\mathcal{N}))$ for which $\tilde{R}^{(\{v\}\to U)} > R^{(\{v\}\to U)}$ for all

$(v,U) \in \mathcal{M}$. We begin by choosing the rate-$\tilde{\mathcal{R}}$ solution $\mathcal{S}(\mathcal{N})$ to be used in each layer of the stack. Set

$$\rho = \min_{(v,U)\in\mathcal{M}} (\tilde{R}^{(\{v\}\to U)} - R^{(\{v\}\to U)}).$$

For any $p \in [0,1]$, let $h(p) \stackrel{\text{def}}{=} -p\log p - (1-p)\log(1-p)$ be the binary entropy function. For reasons that will become clear later, we next find constants λ and n satisfying

$$\max_{(v,U)\in\mathcal{M}} \tilde{R}^{(\{v\}\to U)}\lambda + h(\lambda)/n < \rho$$

such that there exists a $(\lambda, \tilde{\mathcal{R}})$-$\mathcal{S}(\mathcal{N})$ solution of blocklength n. This is possible because $\tilde{\mathcal{R}} \in \text{int}(\mathfrak{R}(\mathcal{N}))$ implies that for any $\lambda \in (0,1]$ and all n sufficiently large there exists a blocklength-n $(\lambda, \tilde{\mathcal{R}})$-$\mathcal{S}(\mathcal{N})$ solution. Thus, we meet the desired constraint by choosing λ to be small (e.g., $\lambda = \rho/(2\max_{(v,U)} \tilde{R}^{(\{v\}\to U)})$) and then choosing n sufficiently large; the chosen n will be the blocklength of code $\mathcal{S}(\underline{\mathcal{N}})$ for all N.

Fix a $(\lambda, \tilde{\mathcal{R}})$-$\mathcal{S}(\mathcal{N})$ solution of blocklength n, denoting the solution's node encoders and decoders by

$$X_t^{(v)} : (\mathcal{Y}^{(v)})^{t-1} \times \tilde{\mathcal{W}}^{(\{v\}\to *)} \to \mathcal{X}^{(v)}$$
$$\widehat{\tilde{W}}^{(\{u\}\to V),v} : (\mathcal{Y}^{(v)})^{n} \times \tilde{\mathcal{W}}^{(\{v\}\to *)} \to \tilde{\mathcal{W}}^{(\{u\}\to V)}$$

where $\tilde{\mathcal{W}}^{(\{v\}\to U)} = \{1,\ldots,2^{n\tilde{R}^{(\{v\}\to U)}}\}$ and $\tilde{\mathcal{W}}^{(\{v\}\to *)} = \prod_{U:U\subseteq\mathcal{V}\setminus\{v\},\, C_{v,U}(\mathcal{N})>0} \tilde{\mathcal{W}}^{(\{v\}\to U)}$.

If we apply $\mathcal{S}(\mathcal{N})$ independently in each layer of stacked network $\underline{\mathcal{N}}$, then for each $(v,U) \in \mathcal{M}$ and each receiver $u \in U$ the N layers of the stack behave like N independent uses of a channel

$$\left(\tilde{\mathcal{W}}^{(\{v\}\to U)}, p^{(\{v\}\to U),u}(\hat{\tilde{w}}^{(\{v\}\to U)} \Big| \tilde{w}^{(\{v\}\to U)}), \tilde{\mathcal{W}}^{(\{v\}\to U)}\right)$$

where

$$p^{(\{v\}\to U),u}\left(\hat{\tilde{w}}^{(\{v\}\to U)} \Big| \tilde{w}^{(\{v\}\to U)}\right) \stackrel{\text{def}}{=}$$
$$\Pr\left(\widehat{\tilde{W}}^{(\{v\}\to U),u} = \hat{\tilde{w}}^{(\{v\}\to U)} \Big| \tilde{W}^{(\{v\}\to U)} = \tilde{w}^{(\{v\}\to U)}\right)$$

is the probability that solution $\mathcal{S}(\mathcal{N})$ reconstructs transmitted message $\tilde{w}^{(\{v\}\to U)}$ as $\hat{\tilde{w}}^{(\{v\}\to U)}$ at node u. We therefore design, for each $(v,U) \in \mathcal{M}$, a $(2^{N(nR^{(\{v\}\to U)})}, N)$ channel code with encoder

$$\underline{\tilde{W}}^{(\{v\}\to U)} : \underline{\mathcal{W}}^{(\{v\}\to U)} \to \underline{\tilde{\mathcal{W}}}^{(\{v\}\to U)}$$

and $|U|$ decoders

$$\widehat{\underline{W}}^{(\{v\}\to U),u} : \underline{\tilde{\mathcal{W}}}^{(\{v\}\to U)} \to \underline{\mathcal{W}}^{(\{v\}\to U)}.$$

The channel code's codewords

$$\left(\underline{\tilde{W}}^{(\{v\}\to U)}(\underline{w}^{(\{v\}\to U)}) : \underline{w}^{(\{v\}\to U)} \in \underline{\mathcal{W}}^{(\{v\}\to U)}\right)$$

are chosen independently and uniformly at random from $\underline{\tilde{\mathcal{W}}}^{(\{v\}\to U)} \stackrel{\text{def}}{=} (\tilde{\mathcal{W}}^{(\{v\}\to U)})^N$. For each $u \in U$, channel decoder $\widehat{\underline{W}}^{(\{v\}\to U),u}(\cdot)$ is the maximum likelihood decoder for discrete memoryless channel $p^{(\{v\}\to U),u}(\hat{\tilde{w}}^{(\{v\}\to U)} | \tilde{w}^{(\{v\}\to U)})$. The mutual information for this channel is

$$I\left(\tilde{W}^{(\{v\}\to U)}; \widehat{\tilde{W}}^{(\{v\}\to U),u}\right)$$
$$\stackrel{(a)}{=} n\tilde{R}^{(\{v\}\to U)} - H\left(\tilde{W}^{(\{v\}\to U)} \Big| \widehat{\tilde{W}}^{(\{v\}\to U),u}\right)$$
$$\stackrel{(b)}{>} n\tilde{R}^{(\{v\}\to U)} - (\lambda n\tilde{R}^{(\{v\}\to U)} + h(\lambda))$$

for each $u \in U$, where (a) follows since $\tilde{W}^{(\{v\}\to U)}$ is uniformly distributed on $\tilde{\mathcal{W}}^{(\{v\}\to U)}$, and (b) follows from Fano's inequality. The desired rate per channel use is $nR^{(\{v\}\to U)}$, which must be reliably decoded by all receivers in U. This rate is strictly less than the mutual information to each receiver since

$$I\left(\tilde{W}^{(\{v\}\to U)}; \widehat{\tilde{W}}^{(\{v\}\to U),u}\right) - nR^{(\{v\}\to U)}$$
$$> n\rho - (\lambda n\tilde{R}^{(\{v\}\to U)} + h(\lambda)) > 0$$

owing to our earlier choice of λ and n. The strong coding theorem for discrete memoryless channels bounds the expected error probability of each randomly drawn code at each of its decoders as

$$E\left[\Pr\left(\widehat{\underline{W}}^{(\{v\}\to U),u} \neq \underline{W}^{(\{v\}\to U)}\right)\right] \leq 2^{-N\delta_0}$$

for some constant $\delta_0 > 0$ and all N sufficiently large [19, Th. 5.6.2]. Since $|\mathcal{V}| = m$, the number of channel decoders is at most

$$m\sum_{i=1}^{m-1}\binom{m-1}{i} i$$
$$= m\sum_{i=1}^{m-1}\frac{(m-1)!i}{i!(m-1-i)!}$$
$$= m(m-1)\sum_{i=1}^{m-1}\binom{m-2}{i-1}$$
$$= m(m-1)2^{m-2}.$$

Therefore, for any $\delta \in (0, \delta_0)$, the union bound gives

$$E[\Pr(\widehat{\underline{W}} \neq \underline{W})] \leq m(m-1)2^{m-2}2^{-N\delta_0} < 2^{-N\delta}$$

for all N sufficiently large. This is the expected error probability with respect to the given distribution over the collection of channel codes for all messages. There must exist a single instance of the channel codes that does at least as well, giving a $(2^{-N\delta}, \mathcal{R})$-$\underline{\mathcal{S}}(\underline{\mathcal{N}})$ solution for each such N. ■

The random code design used in the proof of Theorem 2 chooses a collection of channel codewords uniformly and independently at random and then independently describes each layer of the channel coded messages using the same single-layer solution $\mathcal{S}(\mathcal{N})$. Given this construction, it is not surprising that the resulting network inputs and outputs $(\underline{\mathbf{X}}_t(1), \underline{\mathbf{Y}}_t(1)), \ldots, (\underline{\mathbf{X}}_t(N), \underline{\mathbf{Y}}_t(N))$ for each time t are i.i.d. Lemma 7 in Appendix I gives the formal proof.

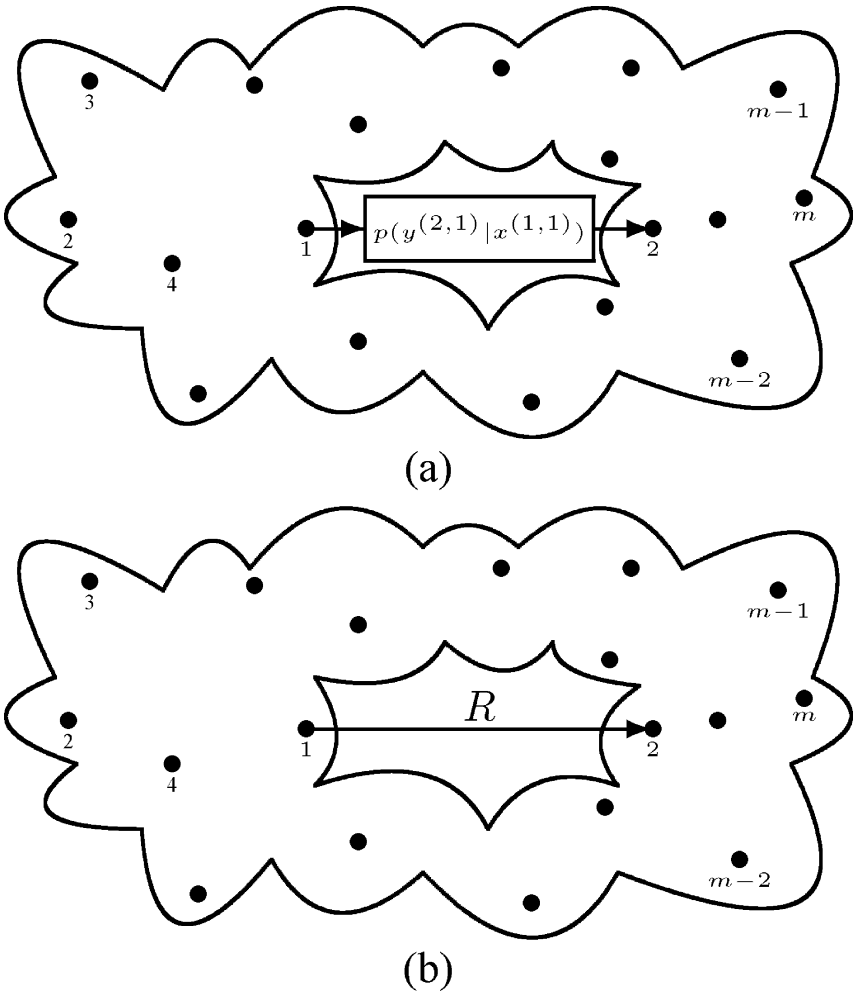

Fig. 8. (a) An m-node network $\mathcal{N}$ containing a channel $\mathcal{C} = (\mathcal{X}^{(1,1)}, p(y^{(2,1)}|x^{(1,1)}), \mathcal{Y}^{(2,1)})$ from node 1 to node 2. The distribution $p(\mathbf{y}^{(*,0)}|\mathbf{x}^{(*,0)})$ on the remaining channel outputs given the remaining channel inputs is arbitrary. (b) The corresponding network $\mathcal{N}^R$ that replaces channel $\mathcal{C}$ by a capacity-R noiseless bit pipe $(\{0,1\}^R, \delta(\tilde{y}^{(2,1)} - \tilde{x}^{(1,1)}), \{0,1\}^R)$.

V. Point-to-Point Channels

As described in Section III, a pair of lower and upper bounding models for some channel $\mathcal{C}$ bounds the range of that channel's behaviors in all networks in which the channel can appear. If $\mathcal{C}$ is a point-to-point channel, then these networks take the form shown in Fig. 8(a). Here, $\mathcal{N} = \mathcal{C}_0 \times \mathcal{C}$ contains independent point-to-point channel

$$\mathcal{C} = (\mathcal{X}^{(1,1)}, p(y^{(2,1)}|x^{(1,1)}), \mathcal{Y}^{(2,1)})$$

from node 1 to node 2 and an arbitrary channel

$$\mathcal{C}_0 = \left(\prod_{v=1}^{m} \mathcal{X}^{(v,0)}, p(\mathbf{y}^{(*,0)}|\mathbf{x}^{(*,0)}), \prod_{v=1}^{m} \mathcal{Y}^{(v,0)} \right)$$

describing the stochastic behavior of the rest of the network.[8] Let C be the capacity of channel $\mathcal{C}$, and let $p^*(x^{(1,1)})$ be an input distribution to channel $\mathcal{C}$ that achieves the channel's capacity.[9]

To bound the behavior of point-to-point channel $\mathcal{C}$, we investigate the implications of replacing that channel by a distinct independent channel

$$\tilde{\mathcal{C}} = (\tilde{\mathcal{X}}^{(1,1)}, \tilde{p}(\tilde{y}^{(2,1)}|\tilde{x}^{(1,1)}), \tilde{\mathcal{Y}}^{(2,1)})$$

giving network $\tilde{\mathcal{N}} = \mathcal{C}_0 \times \tilde{\mathcal{C}}$. Channel $\tilde{\mathcal{C}}$ is a lower bounding model for $\mathcal{C}$ ($\tilde{\mathcal{C}} \subseteq \mathcal{C}$ in the notation of Definition 3) if the existence of a $(\tilde{\lambda}, \mathcal{R})$-$\mathcal{S}(\tilde{\mathcal{N}})$ solution implies the existence of a $(\lambda, \mathcal{R})$-$\mathcal{S}(\mathcal{N})$ solution, where λ can be made arbitrarily small if $\tilde{\lambda}$ can. Channel $\tilde{\mathcal{C}}$ is an upper bounding model for $\mathcal{C}$ ($\tilde{\mathcal{C}} \supseteq \mathcal{C}$ in the notation of Definition 3) if the existence of a $(\lambda, \mathcal{R})$-$\mathcal{S}(\mathcal{N})$ solution implies the existence of a $(\tilde{\lambda}, \mathcal{R})$-$\mathcal{S}(\tilde{\mathcal{N}})$ solution, where $\tilde{\lambda}$ can be made arbitrarily small if λ can. Channel $\tilde{\mathcal{C}}$ is equivalent to channel $\mathcal{C}$ ($\tilde{\mathcal{C}} = \mathcal{C}$ in the notation of Definition 3) if both statements hold. We intend to show that any channel $\tilde{\mathcal{C}}$ of capacity $\tilde{C} \leq C$ is a lower bounding model for $\mathcal{C}$, and any channel $\tilde{\mathcal{C}}$ of capacity $\tilde{C} \geq C$ is an upper bounding model for $\mathcal{C}$. We first treat the case where $\tilde{\mathcal{C}}$ is a noiseless bit pipe, as shown in Fig. 8(b).

For any $R \in \mathbb{R}_+$, a noiseless bit pipe $\mathcal{C}^R$ of capacity R is a channel that delivers R bits per channel use from a single

[8]Node indices are arbitrary. In particular, there is no assumption about a partial ordering on vertices, so there is no loss of generality in assuming that the point-to-point channel $\mathcal{C}$ has transmitter node 1 and receiver node 2.

[9]The implicit assumption of the existence of such a probability distribution is restrictive for continuous-alphabet channels; when $\mathcal{C}$ is continuous, we assume that its capacity C can be achieved (to arbitrary accuracy) by an input distribution that has a probability density function. This includes most of the continuous channels studied in the literature.

transmitter to a single receiver error free. In order to specify both error-free information delivery and the units (bits) by which we quantify information, we denote the channel by

$$\mathcal{C}^R = (\{0,1\}^R, \delta(\tilde{y}^{(2,1)} - \tilde{x}^{(1,1)}), \{0,1\}^R).$$

We employ this notation whether or not $R \in \mathbb{R}^+$ is an integer.

While noninteger capacities are in no way exceptional for noisy channels, their use in noise-free models is less common in the literature. While thinking of information in integer units of bits per channel use (or integer multiples of $m \log_2 q$ bits per channel use in a channel carrying symbols from alphabet q^m) gives us tools for reasoning about network capacities, it is important to remember that even with integer capacities, bit pipes are a purely theoretical construct. Real-world information flow arguably has no more natural quanta than does space or time.

We define noiseless bit pipe $\mathcal{C}^R$ of capacity R to be a mechanism that delivers $\lfloor N_0 R \rfloor$ bits over each block of N_0 channel uses; the mechanism is error free and can be operated at any integer $N_0 \geq 1$. Surprisingly, the precise schedule of information delivery over the N_0 channel uses is of little consequence even in networks with cycles. The capacity of a network $\mathcal{N} = \mathcal{C}_0 \times \mathcal{C}^R$ is unchanged whether bit pipe $\mathcal{C}^R$ delivers its $\lfloor N_0 R \rfloor$ bits evenly over time or entirely in step N_0.[10] Theorem 3 proves the equivalence of a point-to-point channel $\mathcal{C}$ and a noiseless bit pipe of the same capacity.

Theorem 3: Let a point-to-point channel

$$\mathcal{C} = (\mathcal{X}^{(1,1)}, p(y^{(2,1)}|x^{(1,1)}), \mathcal{Y}^{(2,1)})$$

of capacity $C = \max_{p(x^{(1,1)})} I(X^{(1,1)}; Y^{(2,1)}) > 0$ be given, and let

$$\mathcal{C}^C = (\{0,1\}^C, \delta(\tilde{y}^{(2,1)} - \tilde{x}^{(1,1)}), \{0,1\}^C)$$

be a noiseless bit pipe of the same capacity. Then

$$\mathcal{C} = \mathcal{C}^C.$$

Proof: Fix an arbitrary channel $\mathcal{C}_0$. Let $\mathcal{N} = \mathcal{C}_0 \times \mathcal{C}$ and $\mathcal{N}^R = \mathcal{C}_0 \times \mathcal{C}^R$. The proof is accomplished in three steps. Each step appears as an intermediate result, formally stated and proved in Section VI.

Lemma 4 proves that $\mathfrak{R}(\mathcal{N}^R)$ is continuous in R for $R > 0$. Roughly, this involves showing that a solution for N-fold stacked network $\underline{\mathcal{N}}^{R+\delta}$ can be run across N'-fold stacked network $\underline{\mathcal{N}}^{R-\delta}$ provided $N'(R-\delta) \geq N(R+\delta)$; the rate for the latter code approaches that of the former code as δ approaches 0 and the number of layers in both stacks grows without bound.

Lemma 5 proves that $\mathfrak{R}(\mathcal{N}^R) \subseteq \mathfrak{R}(\mathcal{N})$ for all $R < C$. The proof uses a channel code to make noisy channel $\mathcal{C}$ emulate noiseless bit pipe $\mathcal{C}^R$ in a stacked network; using this emulator, a solution for network $\underline{\mathcal{N}}^R$ can be run on network $\underline{\mathcal{N}}$ with similar error probability. It follows from Lemma 5 that $\cup_{R<C} \mathfrak{R}(\mathcal{N}^R) \subseteq \mathfrak{R}(\mathcal{N})$. The closure of $\cup_{R<C} \mathfrak{R}(\mathcal{N}^R)$ equals $\mathfrak{R}(\mathcal{N}^C)$ by the continuity of $\mathfrak{R}(\mathcal{N}^R)$ in R. This implies $\mathfrak{R}(\mathcal{N}^C) \subseteq \mathfrak{R}(\mathcal{N})$ by the closure in the definition of $\mathfrak{R}(\mathcal{N})$.

Theorem 6 proves that $\mathfrak{R}(\mathcal{N}) \subseteq \mathfrak{R}(\mathcal{N}^R)$ for all $R > C$. The proof employs channel emulation to make a noiseless bit pipe $\mathcal{C}^R$ with $R > C$ emulate channel $\mathcal{C}$ in a stacked network; using this emulator, a solution for network $\underline{\mathcal{N}}$ can be run on $\underline{\mathcal{N}}^R$ with similar error probability.[11] Theorem 6 implies $\mathfrak{R}(\mathcal{N}) \subseteq \cap_{R>C} \mathfrak{R}(\mathcal{N}^R)$. The desired result then follows from the continuity of $\mathfrak{R}(\mathcal{N}^R)$ in R. ■

Theorem 3 has a number of interesting implications. Sequentially applying the result to each channel in a network of point-to-point channels proves that the capacity of a network of independent, memoryless, point-to-point channels equals the capacity of another network in which each channel is replaced by a bit pipe of the same capacity. Thus Shannon's channel coding theorem tells us everything that we need to know about the noise in independent, point-to-point channels. What remains is a purely combinatorial question about how much information can be reliably delivered in networks of noiseless bit pipes.

Theorem 3 further demonstrates that there is no loss in capacity associated with first independently channel coding on every point-to-point channel and then network coding across the resulting asymptotically lossless links. This separation result is useful for practical code design since it allows us to leverage the rich literature on channel code design and the more recent literature on network code design to build good codes for networks of noisy point-to-point channels. The mechanism for operating such separated channel and network codes is the one achieved by unraveling a stacked solution as described in Lemma 1; applying the argument in the more obvious way across time is problematic when the network code cannot wait for the channel code to be decoded before beginning its operations.

Theorem 3 also generalizes a variety of analytical and computational tools for finding network capacities from networks of noiseless point-to-point channels to networks of noisy point-to-point channels. For example, the classical result that feedback does not increase the capacity of a point-to-point channel can now be proven in two ways. The first is the classical information theoretic argument that shows that the receiver has no information that is useful to the transmitter that the transmitter does not already know. The second observes that the min-cut between the transmitter and the receiver in the equivalent network is the same with or without feedback; therefore feedback does not increase capacity. While both proofs lead to the same well-known result, the latter is easier to generalize. For example, since capacities are known for a variety of network coding problems, we can immediately determine whether feedback increases the achievable rate regions for a variety of connection types (e.g., multisource multicast, single-source nonoverlapping demands, and single-source nonoverlapping plus multicast demands) in networks of noisy point-to-point channels.

[10] This observation follows from the unraveling argument in the proof of Lemma 1. That proof shows that for any $\mathcal{N}$ and any $\mathcal{R} \in \text{int}(\mathfrak{R}(\mathcal{N}))$, we can build a $(\lambda, \mathcal{R})$-$\mathcal{S}(\mathcal{N})$ solution by unraveling a $(\lambda, \mathcal{R})$-$\mathcal{S}(\underline{\mathcal{N}})$ solution for N-fold stacked network $\underline{\mathcal{N}}$; such a solution is guaranteed to exist for all N sufficiently large. Using this approach, none of the network outputs from time steps $N(t-1)+1, \ldots, Nt$ are required until time $Nt+1$, and thus the precise schedule of their arrival over that period has no impact on the channel capacity.

[11] While it is known that a noiseless bit pipe of a given throughput can emulate any discrete memoryless channel of lesser capacity [20], it is not clear how to apply these results in our context. First, they treat only finite-alphabet channels. Further, direct application would require proving continuity of capacity in the channel statistics. We prove the result directly, without application of [20].

Theorem 3 also implies that the capacity of a network of independent, memoryless, point-to-point channels equals the capacity of any other network of independent, memoryless, point-to-point channels that can be created by replacing each channel by another channel of the same capacity. From the perspective of capacity, a network of Gaussian channels is no different from a corresponding network of binary erasure channels.

VI. Central Proofs

To prove Theorem 3, we first prove the continuity of $\mathfrak{R}(\mathcal{N}^R)$ in R for all $R > 0$.[12] We begin with a definition of continuity. For any $R > 0$ and $\delta \in (0, R)$, let

$$\epsilon(\delta) \stackrel{\text{def}}{=} \max_{\mathcal{R} \in \mathfrak{R}(\underline{\mathcal{N}}^{R+\delta})} \min_{\mathcal{R}' \in \mathfrak{R}(\underline{\mathcal{N}}^{R-\delta})} \|\mathcal{R} - \mathcal{R}'\|_\infty$$

be the worst-case ℓ_∞-norm between a point $\mathcal{R} \in \mathfrak{R}(\underline{\mathcal{N}}^{R+\delta})$ and its closest point $\mathcal{R}' \in \mathfrak{R}(\underline{\mathcal{N}}^{R-\delta})$. We say that $\mathfrak{R}(\underline{\mathcal{N}}^R)$ is continuous in R for all $R > 0$ if for any $\epsilon > 0$, there exists a $\delta \in (0, R)$ for which $\epsilon(\delta) \leq \epsilon$.

Lemma 4: Given a channel $\mathcal{C}_0$, let $\mathcal{N}^R = \mathcal{C}_0 \times \mathcal{C}^R$. Capacity $\mathfrak{R}(\mathcal{N}^R)$ is continuous in R for all $R > 0$.

Proof: By Lemma 1 it suffices to prove that $\mathfrak{R}(\underline{\mathcal{N}}^R)$ is continuous in R. Fix any $\delta \in (0, R)$ and $\mathcal{R} \in \text{int}(\mathfrak{R}(\underline{\mathcal{N}}^{R+\delta}))$. For any $\lambda > 0$ and all N sufficiently large there exists a $(\lambda, \mathcal{R})$-$\mathcal{S}(\underline{\mathcal{N}}^{R+\delta})$ solution for the N-fold stacked network $\underline{\mathcal{N}}^{R+\delta}$. This solution can be run with the same error probability on N'-fold stacked network $\underline{\mathcal{N}}^{R-\delta}$ provided

$$N'(R - \delta) \geq N(R + \delta).$$

This is accomplished by operating solution $\mathcal{S}(\underline{\mathcal{N}}^{R+\delta})$ unchanged across the first N copies of channel $\mathcal{C}_0$ in $\underline{\mathcal{N}}^{R-\delta}$ and sending the $\lfloor N(R+\delta) \rfloor$ bits intended for transmission across N bit pipes of rate $R + \delta$ in $\underline{\mathcal{N}}^{R+\delta}$ across the N' bit pipes of rate $R - \delta$ in $\underline{\mathcal{N}}^{R-\delta}$; here $N'(R - \delta) \geq N(R + \delta)$ implies $\lfloor N'(R - \delta) \rfloor \geq \lfloor N(R + \delta) \rfloor$, so the full transmission can be delivered. If $N' = \lceil N(R + \delta)/(R - \delta) \rceil$, then the rate of the resulting code is

$$\mathcal{R}' = \frac{\mathcal{R}N}{N'} > \mathcal{R} \frac{N}{N(R+\delta)/(R-\delta) + 1}.$$

Since $\mathcal{R}$ and R are fixed, the difference

$$\mathcal{R} - \mathcal{R}' < \mathcal{R} \frac{2N\delta + R - \delta}{N(R+\delta) + R - \delta}$$

approaches 0 as N grows and δ approaches 0. Since $\mathcal{R}$ is arbitrary, the desired result follows. ∎

Lemma 5 and Theorem 6, below, show that $R < C$ implies $\mathcal{C}^R \subseteq \mathcal{C}$ and $R > C$ implies $\mathcal{C} \subseteq \mathcal{C}^R$. Both results are proven by showing that codes designed for one network can be operated across the other network with asymptotically negligible error probability. To operate a code designed for $\mathcal{N}^R = \mathcal{C}_0 \times \mathcal{C}^R$ across network $\mathcal{N} = \mathcal{C}_0 \times \mathcal{C}$, we employ channel coding to make the noisy channel $\mathcal{C}$ in $\mathcal{N}$ emulate the noiseless bit pipe $\mathcal{C}^R$ in $\mathcal{N}^R$. To operate a code designed for $\mathcal{N} = \mathcal{C}_0 \times \mathcal{C}$ across network $\mathcal{N}^R = \mathcal{C}_0 \times \mathcal{C}^R$, we employ a channel emulator to make the noiseless bit pipe $\mathcal{C}^R$ in $\mathcal{N}^R$ emulate the noisy channel $\mathcal{C}$ in $\mathcal{N}$. In both cases, the emulators are run across the layers of a stacked network and not across time.

Lemma 5: Consider a pair of channels

$$\mathcal{C} = (\mathcal{X}^{(1,1)}, p(y^{(2,1)}|x^{(1,1)}), \mathcal{Y}^{(2,1)})$$
$$\mathcal{C}^R = (\{0,1\}^R, \delta(\tilde{y}^{(2,1)} - \tilde{x}^{(1,1)}), \{0,1\}^R)$$

where $C = \max_{p(x^{(1,1)})} I(X^{(1,1)}; Y^{(2,1)}) > 0$ is the capacity of $\mathcal{C}$. If $R < C$, then $\mathcal{C}^R \subseteq \mathcal{C}$.

Proof: To prove that $\mathcal{C}^R$ is a lower bounding model for $\mathcal{C}$, we must show that for any networks $\mathcal{N} = \mathcal{C}_0 \times \mathcal{C}$ and $\mathcal{N}^R = \mathcal{C}_0 \times \mathcal{C}^R$, $\mathfrak{R}(\mathcal{N}^R) \subseteq \mathfrak{R}(\mathcal{N})$. Applying Lemma 1, we prove this result by proving that $\mathfrak{R}(\underline{\mathcal{N}}^R) \subseteq \mathfrak{R}(\underline{\mathcal{N}})$. Fix $\mathcal{R} \in \text{int}(\mathfrak{R}(\underline{\mathcal{N}}^R))$. The argument that follows builds a sequence of rate-$\mathcal{R}$ solutions for stacked network $\underline{\mathcal{N}}$ and shows that the error probability can be made arbitrarily small.

Step 1 – Choose stacked solution $\underline{\mathcal{S}}(\underline{\mathcal{N}}^R)$ for $\underline{\mathcal{N}}^R$: By Theorem 2, there exists a sequence of stacked solutions $\underline{\mathcal{S}}(\underline{\mathcal{N}}^R)$, each of which uses the same single-layer solution $\mathcal{S}(\mathcal{N}^R)$ independently in each layer of the stack, and for which $\Pr(\widehat{\underline{W}} \neq \underline{W}) \leq 2^{-N\delta}$ for all N sufficiently large. Fix such a sequence of codes. Let n be the blocklength of $\mathcal{S}(\mathcal{N}^R)$ (and therefore the blocklength of $\underline{\mathcal{S}}(\underline{\mathcal{N}}^R)$ for all N). Rather than separately specifying the operations of the message channel codes and the single-layer solution that together define the stacked solution, we here use $\underline{\tilde{X}}_t^{(v)}(\cdot)$ to denote their combined action as node encoders and $\widehat{\underline{W}}^{(\{u\} \to V), v}(\cdot)$ to denote their combined action as node decoders for the stacked solution.

Step 2 – Choose channel code (α_N, β_N): Since $R < C$, there exists a sequence of $(2^{NR}, N)$ channel codes $\{(\alpha_N, \beta_N)\}_{N=1}^\infty$ for channel $\mathcal{C}$ with encoders α_N, decoders β_N, and maximal error probability

$$\lambda^{(N)} \stackrel{\text{def}}{=} \max_{\underline{w}} \Pr(\beta_N(\underline{Y}^{(2,1)}) \neq \underline{w} | \underline{X}^{(1,1)} = \alpha_N(\underline{w}))$$

approaching 0 as N grows without bound. Fix such a sequence of channel codes.

Step 3 – Build solution $\mathcal{S}(\underline{\mathcal{N}})$ for $\underline{\mathcal{N}}$: The solution $\mathcal{S}(\underline{\mathcal{N}})$ operates $\underline{\mathcal{S}}(\underline{\mathcal{N}}^R)$ across N-fold stacked network $\underline{\mathcal{N}}$ with the aid of channel code (α_N, β_N), as shown in Fig. 9. At each time $t \in \{1, \ldots, n\}$, the channel code emulates N uses of bit pipe $\mathcal{C}^R$ across the stacked network's N copies of channel $\mathcal{C}$. Let $\underline{X}_t^{(v)}$ and $\underline{Y}_t^{(v)}$ be the time-t input and output of $\underline{\mathcal{N}}$ at node v. The node encoders for $\mathcal{S}(\underline{\mathcal{N}})$ operate the node encoders from $\underline{\mathcal{S}}(\underline{\mathcal{N}}^R)$ as

$$\underline{\tilde{X}}_t^{(v)} = \underline{\tilde{X}}_t^{(v)}(\underline{\tilde{Y}}_1^{(v)}, \ldots, \underline{\tilde{Y}}_{t-1}^{(v)}, \underline{W}^{(\{v\} \to *)})$$

where $\underline{\tilde{Y}}_1^{(v)}, \ldots, \underline{\tilde{Y}}_{t-1}^{(v)}$ are the node's prior stacked network outputs, channel decoded (if necessary) as

$$\underline{\tilde{Y}}_t^{(v)} = \begin{cases} (\underline{Y}_t^{(2,0)}, \beta_N(\underline{Y}_t^{(2,1)})), & v = 2 \\ \underline{Y}_t^{(v)}, & v \neq 2. \end{cases}$$

[12] Continuity of the rate region at $R = 0$ remains an open problem for most networks [21], [22]. The subtle underlying question here is whether a number of bits that grows sublinearly in the coding dimension can change the network capacity.

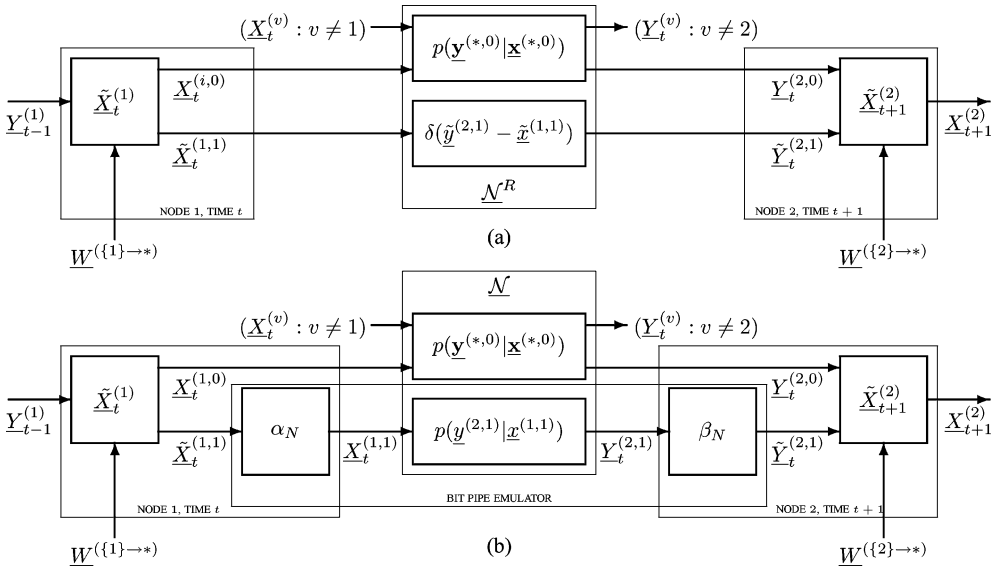

Fig. 9. Operation of node 1 at time t and node 2 at time $t+1$ in solutions (a) $\underline{\mathcal{S}}(\underline{\mathcal{N}}^R)$ and (b) $\mathcal{S}(\underline{\mathcal{N}})$. We show the nodes at different times since the output $\tilde{\underline{X}}_t^{(1,1)}$ from node 1 at time t cannot influence the encoder at node 2 until time $t+1$ (due to the causality constraint).

The resulting channel inputs are then channel encoded (if necessary) as

$$\underline{X}_t^{(v)} = \begin{cases} (\tilde{\underline{X}}_t^{(1,0)}, \alpha_N(\tilde{\underline{X}}_t^{(1,1)})), & \text{if } v = 1 \\ \tilde{\underline{X}}_t^{(v)}, & \text{if } v \neq 1 \end{cases}$$

before transmission. The node decoders for $\mathcal{S}(\underline{\mathcal{N}})$ likewise operate the node decoders from $\underline{\mathcal{S}}(\underline{\mathcal{N}}^R)$ as

$$\widehat{\underline{W}}^{(\{u\}\to V),v} = \widehat{\underline{W}}^{(\{u\}\to V),v}(\tilde{\underline{Y}}_1^{(v)}, \ldots, \tilde{\underline{Y}}_n^{(v)}, \underline{W}^{(\{v\}\to *)}).$$

Step 4 – Bound the error probability for $\mathcal{S}(\underline{\mathcal{N}})$: An error can occur if either the channel code decodes in error at one or more time steps or the channel code decodes correctly in all n time steps but the code $\underline{\mathcal{S}}(\underline{\mathcal{N}}^R)$ fails. If the channel code (α_N, β_N) decodes correctly at all times $t \in \{1, \ldots, n\}$, then the conditional probability of an error given $\underline{W} = \underline{w}$ is precisely what it would have been for the original code. Let E_t denote the event that the channel code fails at time t. We bound the error probability as

$$\begin{aligned} &\Pr(\widehat{\underline{W}} \neq \underline{W}) \\ &\quad \overset{(a)}{\leq} \sum_{t=1}^{n} \Pr(E_t) + \sum_{\underline{w}} \Pr(\{\underline{W} = \underline{w}\} \cap \cap_{t=1}^{n} E_t^c) \\ &\qquad \cdot \Pr(\widehat{\underline{W}} \neq \underline{W} | \{\underline{W} = \underline{w}\} \cap \cap_{t=1}^{n} E_t^c) \\ &\quad \overset{(b)}{\leq} n\lambda^{(N)} + 2^{-N\delta}. \end{aligned}$$

Inequality (a) follows from the union bound; (b) follows from the error probability bound for the channel code and from the observation that $\Pr(\{\underline{W} = \underline{w}\} \cap \cap_{t=1}^{n} E_t^c) \leq \Pr(\underline{W} = \underline{w})$ for all $\underline{w}$. Since n and δ are positive, finite constants, and $\lambda^{(N)}$ decays to zero with increasing N, this bound approaches 0 as N grows without bound. ■

Just as Lemma 5 shows that we can run a solution for $\underline{\mathcal{N}}^R$ across $\underline{\mathcal{N}}$ with the help of a channel code, Theorem 6, below, shows that we can run a solution for $\underline{\mathcal{N}}$ across $\underline{\mathcal{N}}^R$ using a channel emulator to emulate channel $\mathcal{C}$ across noiseless bit pipe $\mathcal{C}^R$. The emulator resembles a lossy source code. Its encoder maps each vector $\underline{x}^{(1,1)}$ of channel inputs to a rate-R binary description, which is transmitted across the noiseless bit pipe. The emulator decoder then maps each binary description to a vector $\underline{y}^{(2,1)}$ of channel outputs. Together, the emulator encoder and decoder map channel inputs to channel outputs.

While each instance of the emulator is deterministic, we design the emulation code at random and take the expectation over the random ensemble of codes. Showing that the resulting expected error probability is small proves the existence of a good instance of the emulation code. The probability that a randomly chosen code maps channel input vector $\underline{x}^{(1,1)}$ to channel output vector $\underline{y}^{(2,1)}$ is designed to approximate the probability $p(\underline{y}^{(2,1)}|\underline{x}^{(1,1)}) = \prod_{\ell=1}^{N} p(\underline{y}^{(2,1)}(\ell)|\underline{x}^{(1,1)}(\ell))$ that $\underline{x}^{(1,1)}$ is mapped to $\underline{y}^{(2,1)}$ in N independent uses of the channel. Since achieving good emulation performance for all possible $(\underline{x}^{(1,1)}, \underline{y}^{(2,1)})$ is difficult, our emulator design focuses on approximating $p(\underline{y}^{(2,1)}|\underline{x}^{(1,1)})$ on the set of jointly typical channel input–output pairs.

Theorem 6: Consider a pair of channels

$$\begin{aligned} \mathcal{C} &= (\mathcal{X}^{(1,1)}, p(y^{(2,1)}|x^{(1,1)}), \mathcal{Y}^{(2,1)}) \\ \mathcal{C}^R &= (\{0,1\}^R, \delta(\tilde{y}^{(2,1)} - \tilde{x}^{(1,1)}), \{0,1\}^R) \end{aligned}$$

where $C = \max_{p(x^{(1,1)})} I(X^{(1,1)}; Y^{(2,1)})$ is the capacity of $\mathcal{C}$. If $R > C$, then $\mathcal{C} \subseteq \mathcal{C}^R$.

Proof: To prove that $\mathcal{C}^R$ is an upper bounding model for $\mathcal{C}$, we must show that for any $\mathcal{N} = \mathcal{C}_0 \times \mathcal{C}$ and $\mathcal{N}^R = \mathcal{C}_0 \times \mathcal{C}^R$, $\mathfrak{R}(\mathcal{N}) \subseteq \mathfrak{R}(\mathcal{N}^R)$. By Lemma 1, it suffices to show that $\mathfrak{R}(\underline{\mathcal{N}}) \subseteq \mathfrak{R}(\underline{\mathcal{N}}^R)$. Fix $\mathcal{R} \in \text{int}(\mathfrak{R}(\underline{\mathcal{N}}))$ and $\lambda > 0$. The argument that follows shows that for any N sufficiently large there exists a $(\lambda, \mathcal{R})$ solution $\underline{\mathcal{S}}(\underline{\mathcal{N}}^R)$ for N-fold stacked network $\underline{\mathcal{N}}^R$. We first design a stacked solution for $\underline{\mathcal{N}}$ and a channel emulator; each is designed independently at random, and then

the two are combined. Good instances of both codes are chosen jointly once both are in place.

Step 1 – Randomly design $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ for $\underline{\mathcal{N}}$: Apply the random stacked solution design algorithm from the proof of Theorem 2 to design a sequence of rate-$\mathcal{R}$ stacked solutions $\underline{\mathcal{S}}(\underline{\mathcal{N}})$. By Theorem 2, the expected probability of the union of all error events is at most $2^{-N\delta}$ for all N sufficiently large. Let n be the blocklength of $\underline{\mathcal{S}}(\underline{\mathcal{N}})$, which is fixed for all N. By Lemma 7, for each $t \in \{1,\ldots,n\}$, the randomly designed stacked solution establishes an i.i.d. distribution $E[p_t(\underline{\mathbf{x}},\underline{\mathbf{y}})] = \prod_{\ell=1}^{N} p_t(\underline{\mathbf{x}}(\ell),\underline{\mathbf{y}}(\ell))$ across the layers of the stack, where the distribution $p_t(\mathbf{x},\mathbf{y})$ in each layer is the time-t distribution on single-layer network inputs and outputs for a single-layer solution $\mathcal{S}(\mathcal{N})$ used in each layer of stacked solution $\underline{\mathcal{S}}(\underline{\mathcal{N}})$; both $\mathcal{S}(\mathcal{N})$ and $p_t(\mathbf{x},\mathbf{y})$ are independent of N. Let

$$p_t(x^{(1,1)}, y^{(2,1)}) = p_t(x^{(1,1)}) p(y^{(2,1)}|x^{(1,1)})$$

be that distribution's marginal on $(X^{(1,1)}, Y^{(2,1)})$.

Under the random stacked solution design,

$$(\underline{X}_t^{(1,1)}(1), \underline{Y}_t^{(2,1)}(1)), (\underline{X}_t^{(1,1)}(2), \underline{Y}_t^{(2,1)}(2)), \ldots$$

are i.i.d., and the probability that $(\underline{X}_t^{(1,1)}, \underline{Y}_t^{(2,1)})$ falls in the jointly typical set $\widehat{A}_{\epsilon,t}^{(N)}$ for $p_t(x^{(1,1)}, y^{(2,1)})$ approaches 1 as N grows without bound. The definition of $\widehat{A}_{\epsilon,t}^{(N)}$ appears in Appendix II. The parameter $\epsilon(t)$ used in that definition varies with t. This is useful both because $p_t(x^{(1,1)}, y^{(2,1)})$ may vary with t and because emulation at one time t affects the distribution at future times. We denote the full vector of parameters by $\epsilon = (\epsilon(1),\ldots,\epsilon(n))$. Lemma 8, which also appears in Appendix II, shows that

$$p_t((\widehat{A}_{\epsilon,t}^{(N)})^c) < 2^{-Nc(\epsilon,t)} \tag{1}$$

for some constant $c(\epsilon,t)$ that approaches zero as $\epsilon(t)$ approaches zero.

Step 2 – Randomly design n channel emulators: For each $t \in \{1,\ldots,n\}$, we design a code $(\alpha_{N,t}, \beta_{N,t})$ to emulate N independent uses of channel $\mathcal{C}$ across N copies of bit pipe $\mathcal{C}^R$. Code $(\alpha_{N,t}, \beta_{N,t})$ emulates $\mathcal{C}$ under input distribution $p_t(x^{(1,1)})$. Mappings

$$\alpha_{N,t} : \underline{\mathcal{X}}^{(1,1)} \to \{0,1\}^{NR}$$
$$\beta_{N,t} : \{0,1\}^{NR} \to \underline{\mathcal{Y}}^{(2,1)}$$

are the emulator encoder and decoder, respectively. We design the emulator decoder at random, drawing codewords

$$\beta_{N,t}(1), \ldots, \beta_{N,t}(2^{NR}) \tag{2}$$

i.i.d. according to the marginal $p_t(\underline{y}^{(2,1)})$ of distribution $p_t(\underline{x}^{(1,1)}, \underline{y}^{(2,1)}) = \prod_{\ell=1}^{N} p_t(\underline{x}^{(1,1)}(\ell), \underline{y}^{(2,1)}(\ell))$. The encoder $\alpha_{N,t}$ is defined as

$$\alpha_{N,t}(\underline{x}^{(1,1)}) = \begin{cases} k, & \text{if } (\underline{x}^{(1,1)}, \beta_{N,t}(k)) \in \widehat{A}_{\epsilon,t}^{(N)} \\ 1, & \text{if } \nexists k \text{ s.t. } (\underline{x}^{(1,1)}, \beta_{N,t}(k)) \in \widehat{A}_{\epsilon,t}^{(N)}. \end{cases} \tag{3}$$

When there is more than one index k for which $(\underline{x}^{(1,1)}, \beta_{N,t}(k)) \in \widehat{A}_{\epsilon,t}^{(N)}$, the encoder design chooses uniformly at random among them.

Any fixed instance of the emulator is a deterministic mapping; the conditional distribution on the emulator output given the emulator input is an indicator function

$$\widehat{p}_t(\underline{y}^{(2,1)}|\underline{x}^{(1,1)}) \stackrel{\text{def}}{=} 1(\underline{y}^{(2,1)} - \beta_{N,t}(\alpha_{N,t}(\underline{x}^{(1,1)}))).$$

The expected value of this distribution with respect to the distribution on emulators imposed by our random design algorithm is

$$E[\widehat{p}_t(\underline{y}^{(2,1)}|\underline{x}^{(1,1)})] \stackrel{\text{def}}{=} \Pr(\beta_{N,t}(\alpha_{N,t}(\underline{x}^{(1,1)})) = \underline{y}^{(2,1)}). \tag{4}$$

Lemma 11 in Appendix II bounds the difference between the channel distribution $p(\underline{y}^{(2,1)}|\underline{x}^{(1,1)})$ and the expected distribution $E[\widehat{p}_t(\underline{y}^{(2,1)}|\underline{x}^{(1,1)})]$ as

$$E[\widehat{p}_t(\underline{y}^{(2,1)}|\underline{x}^{(1,1)})] \le p(\underline{y}^{(2,1)}|\underline{x}^{(1,1)}) 2^{N(4a(\epsilon,t)+2\epsilon(t)+1/N)} \tag{5}$$

for all $(\underline{x}^{(1,1)}, \underline{y}^{(2,1)}) \in \widehat{A}_{\epsilon,t}^{(N)}$. This result bounds the accuracy with which the random ensemble of emulators approximates the desired channel distribution when $(\underline{X}_t^{(1,1)}, \underline{Y}_t^{(2,1)}) \in \widehat{A}_{\epsilon,t}^{(N)}$. Here, $a(\epsilon,t)$, defined along with typical set $\widehat{A}_{\epsilon,t}^{(N)}$ in Appendix II, approaches zero as $\epsilon(t)$ approaches zero.

To bound the probability that $(\underline{X}_t^{(1,1)}, \underline{Y}_t^{(2,1)}) \notin \widehat{A}_{\epsilon,t}^{(N)}$ under operation of the randomly designed emulator $(\alpha_{N,t}, \beta_{N,t})$, let

$$E[\widehat{p}_t((\widehat{A}_{\epsilon,t}^{(N)})^c|\underline{x}^{(1,1)})] \stackrel{\text{def}}{=} \sum_{\underline{y}^{(2,1)}:(\underline{x}^{(1,1)},\underline{y}^{(2,1)})\notin\widehat{A}_{\epsilon,t}^{(N)}} E[\widehat{p}_t(\underline{y}^{(2,1)}|\underline{x}^{(1,1)})].$$

Using a proof similar to the proof of the rate-distortion theorem, Lemma 12 in Appendix II shows

$$E\left[\widehat{p}_t((\widehat{A}_{\epsilon,t}^{(N)})^c|\underline{x}^{(1,1)})\right] \le p((\widehat{A}_{\epsilon,t}^{(N)})^c|\underline{x}^{(1,1)}) + e^{-2^{N(R-I(X_t^{(1,1)};Y_t^{(2,1)})-2a(\epsilon,t)-\epsilon(t))}} \tag{6}$$

for all $\underline{x}^{(1,1)}$.

Note that the definitions of $\widehat{A}_{\epsilon,t}^{(N)}$ and $p_t(\underline{x}^{(1,1)})$ depend only on the single-layer solution $\mathcal{S}(\mathcal{N})$ in stacked solution $\underline{\mathcal{S}}(\underline{\mathcal{N}})$, and recall that that single-layer solution is not randomly designed but deterministically chosen. Thus the distribution $p_t(\underline{x}^{(1,1)}, \underline{y}^{(2,1)})$, and the typical set defined for it, are fixed over all random codes. As a result, $E[p_t(\underline{x}^{(1,1)})] = p_t(\underline{x}^{(1,1)})$ and $E[p_t(\widehat{A}_{\epsilon,t}^{(N)}|\underline{x}^{(1,1)})] = p_t(\widehat{A}_{\epsilon,t}^{(N)}|\underline{x}^{(1,1)})$. This explains the absence of expectations around $\widehat{A}_{\epsilon,t}^{(N)}$ and $p_t(\underline{x}^{(1,1)}, \underline{y}^{(2,1)})$ throughout. (See, for example, the right hand side of (6).)

Step 3 – Build solution $\mathcal{S}(\underline{\mathcal{N}}^R)$ for $\underline{\mathcal{N}}^R$: Solution $\mathcal{S}(\underline{\mathcal{N}}^R)$ operates $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ across network $\underline{\mathcal{N}}^R$ with the aid of the n emulation codes $\{(\alpha_{N,t}, \beta_{N,t})\}_{t=1}^{n}$; $\mathcal{S}(\underline{\mathcal{N}}^R)$ is not a stacked solution since the emulation codes code across the layers of the stack. We begin with an informal code description. For each node $v \notin \{1,2\}$, the operation of node v in $\mathcal{S}(\underline{\mathcal{N}}^R)$ is identical to its operation in $\underline{\mathcal{S}}(\underline{\mathcal{N}})$. At node 1, solution $\mathcal{S}(\underline{\mathcal{N}}^R)$ applies the node encoder from $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ followed by the emulator

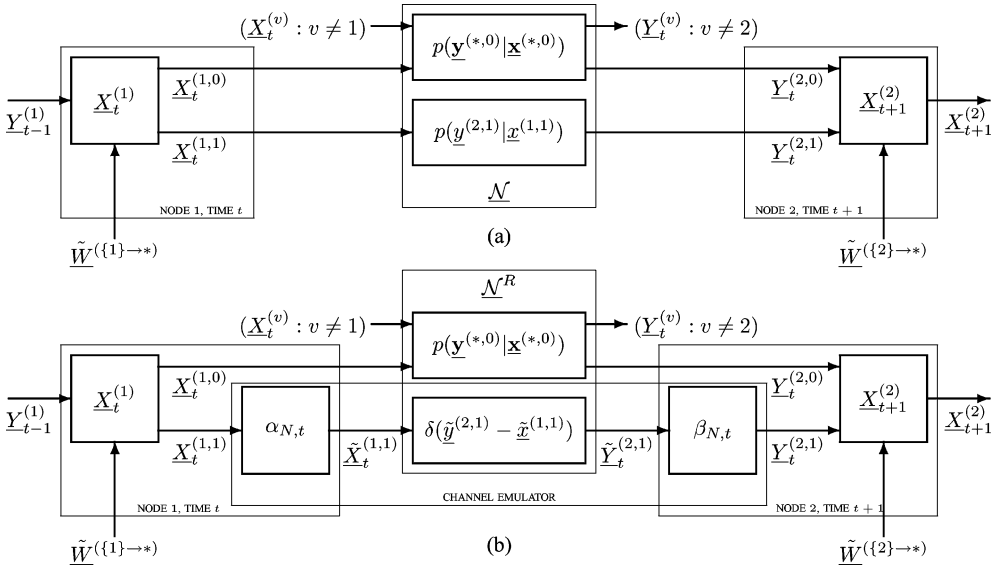

Fig. 10. Operation of node 1 at time t and node 2 at time $t+1$ in solutions (a) $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ and (b) $\mathcal{S}(\underline{\mathcal{N}}^R)$. We show the nodes at different times since the stacked network input $\underline{X}_t^{(1,1)}$ from node 1 at time t cannot influence the encoder at node 2 until time $t+1$ (due to the causality constraint).

encoder, which maps $\underline{X}_t^{(1,1)}$ to a binary description $\underline{\tilde{X}}_t^{(1,1)}$ to be sent across the bit pipe. The node decoder at node 1 is unchanged. At node 2, $\mathcal{S}(\underline{\mathcal{N}}^R)$ applies the emulator decoder to map the bit-pipe output $\underline{\tilde{Y}}_t^{(2,1)}$ to a channel output $\underline{Y}_t^{(2,1)}$ before applying the encoder and decoder from $\underline{\mathcal{S}}(\underline{\mathcal{N}})$. Fig. 10 illustrates these operations, which are defined formally below.

For each $v \in \mathcal{V}$, node v first channel codes its outgoing messages $\underline{W}^{(\{v\}\to U)}$, $U \subseteq \mathcal{V}\setminus\{v\}$, $C_{v,U}(\mathcal{N}) > 0$, as

$$\underline{\tilde{W}}^{(\{v\}\to U)} = \underline{\tilde{W}}^{(\{v\}\to U)}(\underline{W}^{(\{v\}\to U)})$$

using the channel code chosen in the random code design of Step 1. Let $(\underline{\tilde{X}}_t^{(v)}, \underline{\tilde{Y}}_t^{(v)})$ be the time-t input and output of $\underline{\mathcal{N}}_R$ at node v. At each time $t \in \{1,\dots,n\}$, node v applies its node encoder as

$$\underline{X}_t^{(v)}(\ell) = \underline{X}_t^{(v)}(\underline{Y}_1^{(v)}(\ell),\dots,\underline{Y}_{t-1}^{(v)}(\ell), \underline{\tilde{W}}^{(\{v\}\to *)}(\ell)).$$

Here, $\underline{Y}_t^{(v)}$ is the stacked network output decoded (if necessary) using the randomly designed emulation decoder, giving

$$\underline{Y}_t^{(v)} = \begin{cases} (\underline{\tilde{Y}}_t^{(2,0)}, \beta_{N,t}(\underline{\tilde{Y}}_t^{(2,1)})), & \text{if } v = 2 \\ \underline{\tilde{Y}}_t^{(v)}, & \text{if } v \neq 2. \end{cases}$$

Node v applies (if necessary) the channel emulation encoder before each transmission, giving

$$\underline{\tilde{X}}_t^{(v)} = \begin{cases} (\underline{X}_t^{(1,0)}, \alpha_{N,t}(\underline{X}_t^{(1,1)})), & \text{if } v = 1 \\ \underline{X}_t^{(v)}, & \text{if } v \neq 1. \end{cases}$$

At time n, node v applies the node-v decoders from $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ as

$$\underline{\widehat{\tilde{W}}}^{(\{u\}\to V),v}(\ell) = \widehat{\tilde{W}}^{(\{u\}\to V),v}(\underline{Y}_1^{(v)}(\ell),\dots,\underline{Y}_n^{(v)}(\ell), \underline{\tilde{W}}^{(\{v\}\to *)}(\ell))$$

and then applies the channel decoders from $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ to give

$$\underline{\widehat{W}}^{(\{u\}\to V),v} = \underline{\widehat{W}}^{(\{u\}\to V),v}\left(\underline{\widehat{\tilde{W}}}^{(\{u\}\to V),v}\right).$$

Step 4 – Bound the error probability for $\mathcal{S}(\underline{\mathcal{N}}^R)$: The error analysis begins with a characterization of error events. The definitions of error events rely on both expected probabilities resulting from the operation of random solution $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ on $\underline{\mathcal{N}}$ and expected probabilities resulting from the operation of random solution $\mathcal{S}(\underline{\mathcal{N}}^R)$ on $\underline{\mathcal{N}}^R$. To avoid confusion, we use $E[\Pr(\cdot)]$ for the former and $E[\widehat{\Pr}(\cdot)]$ for the latter.

For any fixed instance of code $\underline{\mathcal{S}}(\underline{\mathcal{N}})$, define

$$B_t^{(N)} \stackrel{\text{def}}{=} \Big\{(\underline{x}^{(1,1)}, \underline{y}^{(2,1)}) : \Pr\left(\underline{\widehat{W}} \neq \underline{W} \,\middle|\, (\underline{X}_t^{(1,1)}, \underline{Y}_t^{(2,1)}) = (\underline{x}^{(1,1)}, \underline{y}^{(2,1)})\right) \geq 2^{-N\delta/2}\Big\} \tag{7}$$

to be the set of input–output pairs for channel $\mathcal{C}$ at time t for which the conditional probability of an error in the operation of $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ on $\underline{\mathcal{N}}$ exceeds threshold $2^{-N\delta/2}$; we think of $B_t^{(N)}$ as the "bad" set since it contains channel input–output pairs for which the conditional probability of an error decays to zero significantly more slowly than the expected rate of decay $2^{-N\delta}$ derived for our random code design in Theorem 2.

We treat channel input–output pairs that are atypical or fall in the "bad" set as error events. To bound the probability that $(\underline{X}_t^{(1,1)}, \underline{Y}_t^{(2,1)}) \notin (\widehat{A}_{\epsilon,t}^{(N)} \setminus B_t^{(N)})$ for some time t, let $G_0 \stackrel{\text{def}}{=} (\underline{\mathcal{X}}^{(1,1)})^n \times (\underline{\mathcal{Y}}^{(2,1)})^n$, and for $t \in \{1,\dots,n\}$ let $G_t \subseteq G_0$ be the set of all channel input–output pairs that do not experience such an error in the first t time steps; then

$$G_t \stackrel{\text{def}}{=} \bigcap_{t'=1}^{t} \Big\{\Big((\underline{x}^{(1,1)})^n, (\underline{y}^{(2,1)})^n\Big) : (\underline{x}_{t'}^{(1,1)}, \underline{y}_{t'}^{(2,1)}) \in \widehat{A}_{\epsilon,t'}^{(N)} \setminus B_{t'}^{(N)}\Big\}$$

and $((\underline{X}^{(1,1)})^n, (\underline{Y}^{(2,1)})^n) \notin G_n$ captures all error events due to atypicality or "bad" channel input–output pairs. Note that

$$\begin{aligned}(G_n)^c &= \bigcup_{t=1}^{n} \left(G_{t-1} \cap (\widehat{A}_{\epsilon,t}^{(N)} \setminus B_t^{(N)})^c \right) \\ &= \bigcup_{t=1}^{n} \left(\left(G_{t-1} \cap (\widehat{A}_{\epsilon,t}^{(N)})^c \right) \right. \\ &\quad \left. \cup \left(G_{t-1} \cap \widehat{A}_{\epsilon,t}^{(N)} \cap B_t^{(N)} \right) \right).\end{aligned}$$

By the union bound, the expected probability of $(G_n)^c$ satisfies

$$\begin{aligned}E\left[\widehat{\Pr}((G_n)^c)\right] &\le \sum_{t=1}^{n} E\left[\widehat{\Pr}\left(G_{t-1} \cap (\widehat{A}_{\epsilon,t}^{(N)})^c\right)\right] \\ &\quad + \sum_{t=1}^{n} E\left[\widehat{\Pr}\left(G_{t-1} \cap \widehat{A}_{\epsilon,t}^{(N)} \cap B_t^{(N)}\right)\right]\end{aligned}$$

and we bound the error probability of our solution as

$$\begin{aligned}&E\left[\widehat{\Pr}\left(\underline{\widehat{W}} \neq \underline{W}\right)\right] \\ &\le E\left[\widehat{\Pr}\left((G_n)^c \cup \left(G_n \cap \{\underline{\widehat{W}} \neq \underline{W}\}\right)\right)\right] \\ &\le \sum_{t=1}^{n} E\left[\widehat{\Pr}\left(G_{t-1} \cap (\widehat{A}_{\epsilon,t}^{(N)})^c\right)\right] \qquad (8) \\ &\quad + \sum_{t=1}^{n} E\left[\widehat{\Pr}\left(G_{t-1} \cap \widehat{A}_{\epsilon,t}^{(N)} \cap B_t^{(N)}\right)\right] \qquad (9) \\ &\quad + E\left[\widehat{\Pr}\left(G_n \cap \{\underline{\widehat{W}} \neq \underline{W}\}\right)\right]. \qquad (10)\end{aligned}$$

The expectation captures the random code design, while $\widehat{\Pr}$ captures the random message choice and random action of channel $\mathcal{C}_0$.

Bounding each of the terms in this sum requires a characterization of the expected distribution achieved by randomly designed solution $\mathcal{S}(\underline{\mathcal{N}}^R)$. Recall from (18) in Appendix I that the expected behavior of solution $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ is characterized by distribution

$$\begin{aligned}&E[p(\underline{w}, \underline{\tilde{w}}, \underline{\mathbf{x}}^n, \underline{\mathbf{y}}^n, \underline{\widehat{\tilde{w}}}, \underline{\widehat{w}})] \\ &= p(\underline{w}) E[p(\underline{\tilde{w}}|\underline{w}) p(\underline{\widehat{w}}|\underline{\widehat{\tilde{w}}})] p(\underline{\widehat{\tilde{w}}}|\underline{\mathbf{y}}^n, \underline{\tilde{w}}) \\ &\quad \cdot \prod_{t=1}^{n} \left[p(\underline{\mathbf{x}}_t|\underline{\mathbf{y}}^{t-1}, \underline{\tilde{w}}) p(\underline{\mathbf{y}}_t|\underline{\mathbf{x}}_t) \right]. \qquad (11)\end{aligned}$$

Solution $\mathcal{S}(\underline{\mathcal{N}}^R)$ can be similarly characterized. In particular, since the messages are again uniformly distributed and we employ the same stacked solution and channel codes, distributions $p(\underline{w})$, $p(\underline{\tilde{w}}|\underline{w})$, $p(\underline{\mathbf{x}}_t|\underline{\mathbf{y}}^{t-1}, \underline{\tilde{w}})$, $p(\underline{\widehat{\tilde{w}}}|\underline{\mathbf{y}}^n, \underline{\tilde{w}})$, and $p(\underline{\widehat{w}}|\underline{\widehat{\tilde{w}}})$ remain unchanged. The difference between $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ and $\mathcal{S}(\underline{\mathcal{N}}^R)$ is the replacement of channel $\mathcal{C}$ at time t by the operation of the channel emulator $(\alpha_{N,t}, \beta_{N,t})$ across noiseless bit-pipe $\mathcal{C}^R$. Thus at time t, solution $\mathcal{S}(\underline{\mathcal{N}}^R)$ replaces the channel distribution $p(\underline{y}^{(2,1)}|\underline{x}^{(1,1)})$ by the emulator distribution $\widehat{p}_t(\underline{y}^{(2,1)}|\underline{x}^{(1,1)})$. Since $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ and $(\alpha_{N,t}, \beta_{N,t})$ for each t are all chosen independently at random, the expected distribution imposed by the randomly designed solution $\mathcal{S}(\underline{\mathcal{N}}^R)$ is

$$\begin{aligned}&E[\widehat{p}(\underline{w}, \underline{\tilde{w}}, \underline{\mathbf{x}}^n, \underline{\mathbf{y}}^n, \underline{\widehat{\tilde{w}}}, \underline{\widehat{w}})] \\ &= p(\underline{w}) E[p(\underline{\tilde{w}}|\underline{w}) p(\underline{\widehat{w}}|\underline{\widehat{\tilde{w}}})] p(\underline{\widehat{\tilde{w}}}|\underline{\mathbf{y}}^n, \underline{\tilde{w}}) \\ &\quad \cdot \prod_{t=1}^{n} \left[p(\underline{\mathbf{x}}_t|\underline{\mathbf{y}}^{t-1}, \underline{\tilde{w}}) E[\widehat{p}_t(\underline{\mathbf{y}}_t|\underline{\mathbf{x}}_t)] \right] \qquad (12)\end{aligned}$$

where

$$\begin{aligned}E[\widehat{p}_t(\underline{\mathbf{y}}_t|\underline{\mathbf{x}}_t)] &\stackrel{\text{def}}{=} E[\widehat{p}_t(\underline{y}_t^{(2,1)}|\underline{x}_t^{(1,1)}) p(\underline{\mathbf{y}}_t^{(*,0)}|\underline{\mathbf{x}}_t^{(*,0)})] \\ &= E[\widehat{p}_t(\underline{y}_t^{(2,1)}|\underline{x}_t^{(1,1)})] p(\underline{\mathbf{y}}_t^{(*,0)}|\underline{\mathbf{x}}_t^{(*,0)}).\end{aligned}$$

To bound (8) and (9), we first bound

$$E[\widehat{\Pr}(G_{t-1}, \underline{x}_t^{(1,1)})] \stackrel{\text{def}}{=} E[\widehat{\Pr}(G_{t-1} \cap \{\underline{X}_t^{(1,1)} = \underline{x}_t^{(1,1)}\})]$$

by summing expected probability $E[\widehat{p}(\underline{w}, \underline{\tilde{w}}, \underline{\mathbf{x}}^t, \underline{\mathbf{y}}^{t-1})]$ over all vectors $(\underline{w}, \underline{\tilde{w}}, \underline{\mathbf{x}}^{t-1}, \underline{\mathbf{y}}^{t-1}, \underline{\mathbf{x}}_t^{(*,0)})$ that satisfy $(\underline{x}_{t'}^{(1,1)}, \underline{y}_{t'}^{(2,1)}) \in \widehat{A}_{\epsilon,t'}^{(N)} \setminus B_{t'}^{(N)}$ for $t' < t$. In taking this sum, we include the full expression inside the expectation since the definition of $B_t^{(N)}$ for each t (and therefore the definition of G_t for each t) depends on the randomly chosen instance of the solution $\underline{\mathcal{S}}(\underline{\mathcal{N}})$. Thus

$$\begin{aligned}&E\left[\widehat{\Pr}(G_{t-1}, \underline{x}_t^{(1,1)})\right] \\ &\stackrel{(a)}{=} E\left[\sum p(\underline{w}) p(\underline{\tilde{w}}|\underline{w}) \left[\prod_{t'=1}^{t} p(\underline{\mathbf{x}}_{t'}|\underline{\mathbf{y}}^{t'-1}, \underline{\tilde{w}})\right]\right. \\ &\quad \left. \cdot \left[\prod_{t'=1}^{t-1} \widehat{p}_t(\underline{y}_{t'}^{(2,1)}|\underline{x}_{t'}^{(1,1)}) p(\underline{\mathbf{y}}_{t'}^{(*,0)}|\underline{\mathbf{x}}_{t'}^{(*,0)})\right]\right] \\ &\stackrel{(b)}{\le} E\left[\sum 2^{N \sum_{t'=1}^{t-1} (4a(\epsilon,t') + 2\epsilon(t') + 1/N)}\right. \\ &\quad \left. \cdot\, p(\underline{w}, \underline{\tilde{w}}, \underline{\mathbf{x}}^t, \underline{\mathbf{y}}^{t-1})\right] \\ &\stackrel{(c)}{\le} 2^{N \sum_{t'=1}^{t-1} (4a(\epsilon,t') + 2\epsilon(t') + 1/N)} p_t(\underline{x}_t^{(1,1)}) \qquad (13)\end{aligned}$$

for each $\underline{x}_t^{(1,1)} \in \underline{\mathcal{X}}^{(1,1)}$. Here, (a) follows from (12), (b) employs the bound on the accuracy of our channel emulator from (5), and the inequality in (c) follows from summing over all $(\underline{x}_{t'}^{(1,1)}, \underline{y}_{t'}^{(2,1)})$ for $t' < t$ rather than just $(\underline{x}_{t'}^{(1,1)}, \underline{y}_{t'}^{(2,1)}) \in \widehat{A}_{\epsilon,t'}^{(N)} \setminus B_{t'}^{(N)}$, $t' < t$. No expectation is required in (c) since $p_t(\underline{x}_t^{(1,1)})$ is fixed for all random codes, as discussed above. The given bound captures how the input distribution to node 1 at time t is affected by the replacement of the channel by its emulator in all previous time steps.

We apply this result to bound (8) as

$$\begin{aligned}&E\left[\widehat{\Pr}(G_{t-1} \cap (\widehat{A}_{\epsilon,t}^{(N)})^c)\right] \\ &\stackrel{(a)}{=} \sum_{\underline{x}_t^{(1,1)}} E\left[\widehat{\Pr}(G_{t-1}, \underline{x}_t^{(1,1)}) \widehat{p}_t((\widehat{A}_{\epsilon,t}^{(N)})^c|\underline{x}_t^{(1,1)})\right]\end{aligned}$$

$$
\begin{aligned}
&\overset{(b)}{=} \sum_{\underline{x}_t^{(1,1)}} E[\widehat{\Pr}(G_{t-1}, \underline{x}_t^{(1,1)})] E[\widehat{p}_t((\widehat{A}_{\epsilon,t}^{(N)})^c | \underline{x}_t^{(1,1)})] \\
&\overset{(c)}{\leq} \sum_{\underline{x}_t^{(1,1)}} E\left[\widehat{\Pr}(G_{t-1}, \underline{x}_t^{(1,1)})\right] \Big[p((\widehat{A}_{\epsilon,t}^{(N)})^c | \underline{x}_t^{(1,1)}) \\
&\quad + e^{-2^{N(R-I(X_t^{(1,1)};Y_t^{(2,1)})-2a(\epsilon,t)-\epsilon(t))}}\Big] \\
&= \sum_{\underline{x}_t^{(1,1)}} E\left[\widehat{\Pr}(G_{t-1}, \underline{x}_t^{(1,1)})\right] p((\widehat{A}_{\epsilon,t}^{(N)})^c | \underline{x}_t^{(1,1)}) \\
&\quad + \sum_{\underline{x}_t^{(1,1)}} E\left[\widehat{\Pr}(G_{t-1}, \underline{x}_t^{(1,1)})\right] \\
&\quad \cdot e^{-2^{N(R-I(X_t^{(1,1)};Y_t^{(2,1)})-2a(\epsilon,t)-\epsilon(t))}} \\
&\overset{(d)}{\leq} \sum_{\underline{x}_t^{(1,1)}} E\left[\widehat{\Pr}(G_{t-1}, \underline{x}_t^{(1,1)})\right] p((\widehat{A}_{\epsilon,t}^{(N)})^c | \underline{x}_t^{(1,1)}) \\
&\quad + e^{-2^{N(R-I(X_t^{(1,1)};Y_t^{(2,1)})-2a(\epsilon,t)-\epsilon(t))}} \\
&\overset{(e)}{\leq} \sum_{\underline{x}_t^{(1,1)}} 2^{N\sum_{t'=1}^{t-1}(4a(\epsilon,t')+2\epsilon(t')+1/N)} p_t(\underline{x}_t^{(1,1)}) \\
&\quad \cdot p((\widehat{A}_{\epsilon,t}^{(N)})^c | \underline{x}_t^{(1,1)}) \\
&\quad + e^{-2^{N(R-I(X_t^{(1,1)};Y_t^{(2,1)})-2a(\epsilon,t)-\epsilon(t))}} \\
&\overset{(f)}{\leq} 2^{-N(c(\epsilon,t)-\sum_{t'=1}^{t-1}(4a(\epsilon,t')+2\epsilon(t')+1/N))} \\
&\quad + e^{-2^{N(R-I(X_t^{(1,1)};Y_t^{(2,1)})-2a(\epsilon,t)-\epsilon(t))}}. \qquad (14)
\end{aligned}
$$

Here, (a) follows since the emulator $(\alpha_{N,t}, \beta_{N,t})$ observes only the channel input $\underline{X}_t^{(1,1)}$; (b) follows from the independence of the random solution and emulator designs; (c) follows from (6); (d) follows since

$$\sum_{\underline{x}_t^{(1,1)}} E[\widehat{\Pr}(G_{t-1}, \underline{x}_t^{(1,1)})] = E[\widehat{\Pr}(G_{t-1})] \leq 1$$

(e) follows from (13); and (f) follows from (1).

To bound (9), recall from Theorem 2 that for all N sufficiently large the expected probability that solution $\mathcal{S}(\underline{\mathcal{N}})$ decodes any messages in error on network $\underline{\mathcal{N}}$ is bounded by $2^{-N\delta}$. Thus

$$
\begin{aligned}
2^{-N\delta} &\geq E\left[\Pr(\underline{\widehat{W}} \neq \underline{W})\right] \\
&\geq E\Bigg[\sum_{(\underline{x}_t^{(1,1)}, \underline{y}_t^{(2,1)}) \in B_t^{(N)}} p_t(\underline{x}_t^{(1,1)}, \underline{y}_t^{(2,1)}) \\
&\qquad \Pr(\underline{\widehat{W}} \neq \underline{W} | (\underline{x}_t^{(1,1)}, \underline{y}_t^{(2,1)}))\Bigg] \\
&\overset{(a)}{\geq} 2^{-N\delta/2} E\left[p_t(B_t^{(N)})\right]
\end{aligned}
$$

where (a) follows from the definition of $B_t^{(N)}$ in (7). Thus the expected probability of set $B_t^{(N)}$ on $\underline{\mathcal{N}}$ is

$$E\left[p_t(B_t^{(N)})\right] \leq 2^{-N\delta/2}. \qquad (15)$$

For solution $\mathcal{S}(\underline{\mathcal{N}}^R)$ on $\underline{\mathcal{N}}^R$

$$
\begin{aligned}
&E\left[\widehat{\Pr}(G_{t-1} \cap \widehat{A}_{\epsilon,t}^{(N)} \cap B_t^{(N)})\right] \\
&= \sum_{\underline{x}_t^{(1,1)}} E\left[\widehat{\Pr}(G_{t-1}, \underline{x}_t^{(1,1)}) \widehat{p}_t(\widehat{A}_{\epsilon,t}^{(N)} \cap B_t^{(N)} | \underline{x}_t^{(1,1)})\right] \\
&\overset{(a)}{\leq} 2^{N\sum_{t'=1}^{t-1}(4a(\epsilon,t')+2\epsilon(t')+1/N)} \sum_{\underline{x}_t^{(1,1)}} p_t(\underline{x}_t^{(1,1)}) \\
&\quad \cdot E[\widehat{p}_t(\widehat{A}_{\epsilon,t}^{(N)} \cap B_t^{(N)} | \underline{x}_t^{(1,1)})] \\
&\overset{(b)}{\leq} 2^{N\sum_{t'=1}^{t}(4a(\epsilon,t')+2\epsilon(t')+1/N)} \sum_{\underline{x}_t^{(1,1)}} p_t(\underline{x}_t^{(1,1)}) \\
&\quad \cdot E[p_t(\widehat{A}_{\epsilon,t}^{(N)} \cap B_t^{(N)} | \underline{x}_t^{(1,1)})] \\
&\overset{(c)}{\leq} 2^{-N(\delta/2-\sum_{t'=1}^{t}(4a(\epsilon,t')+2\epsilon(t')+1/N))}. \qquad (16)
\end{aligned}
$$

Here, (a) follows from (13); (b) follows from (5); and (c) follows from (15). An expectation is required in inequality (b) since the definition of $B_t^{(N)}$ relies on the randomly chosen solution.

To bound (10), we sum $E[\widehat{p}(\underline{w}, \underline{\tilde{w}}, \underline{\mathbf{x}}^n, \underline{\mathbf{y}}^n, \underline{\hat{\tilde{w}}}, \underline{\hat{w}})]$ over all $(\underline{w}, \underline{\tilde{w}}, \underline{\mathbf{x}}^n, \underline{\mathbf{y}}^n, \underline{\hat{\tilde{w}}}, \underline{\hat{w}})$, for which $\underline{w} \neq \underline{\hat{w}}$ and $(\underline{x}_t^{(1,1)}, \underline{y}_t^{(2,1)}) \in \widehat{A}_{\epsilon,t}^{(N)} \setminus B_t^{(N)}$ for all $t \in \{1, \ldots, n\}$. The resulting bound is

$$
\begin{aligned}
&E\left[\widehat{\Pr}\left(G_n \cap \{\underline{\widehat{W}} \neq \underline{W}\}\right)\right] \\
&\overset{(a)}{=} E\Bigg[\sum p(\underline{w}) p(\underline{\tilde{w}}|\underline{w}) p(\underline{\hat{\tilde{w}}}|\underline{\mathbf{y}}^n, \underline{\tilde{w}}) p(\underline{\hat{w}}|\underline{\hat{\tilde{w}}}) \\
&\qquad \cdot \left[\prod_{t=1}^{n} p(\underline{\mathbf{x}}_t | \underline{\mathbf{y}}^{t-1}, \underline{\tilde{w}}) \widehat{p}_t(\underline{\mathbf{y}}_t | \underline{\mathbf{x}}_t)\right]\Bigg] \\
&\overset{(b)}{\leq} 2^{N\sum_{t=1}^{n}(4a(\epsilon,t)+2\epsilon(t)+1/N)} \\
&\qquad \cdot E\left[\sum p(\underline{w}, \underline{\tilde{w}}, \underline{\mathbf{x}}^n, \underline{\mathbf{y}}^n, \underline{\hat{\tilde{w}}}, \underline{\hat{w}})\right] \\
&\overset{(c)}{\leq} 2^{N\sum_{t=1}^{n}(4a(\epsilon,t)+2\epsilon(t)+1/N)} \\
&\qquad \cdot E\left[\sum p(\underline{w}, \underline{x}_1^{(1,1)}, \underline{y}_1^{(2,1)}, \underline{\hat{w}})\right] \\
&= 2^{N\sum_{t=1}^{n}(4a(\epsilon,t)+2\epsilon(t)+1/N)} \\
&\qquad \cdot E\Bigg[\sum_{(\underline{x}_1^{(1,1)}, \underline{y}_1^{(2,1)}) \in \widehat{A}_{\epsilon,1}^{(N)} \setminus B_1^{(N)}} p_1(\underline{x}_1^{(1,1)}, \underline{y}_1^{(2,1)}) \\
&\qquad \cdot \Pr(\underline{\widehat{W}} \neq \underline{W} | (\underline{x}_1^{(1,1)}, \underline{y}_1^{(2,1)}))\Bigg] \\
&\overset{(d)}{<} 2^{N\sum_{t=1}^{n}(4a(\epsilon,t)+2\epsilon(t)+1/N)} 2^{-N\delta/2}. \qquad (17)
\end{aligned}
$$

Here, (a) follows from (12); (b) follows from (5) and (11). In (c), we remove the restriction $(\underline{x}_t^{(1,1)}, \underline{y}_t^{(2,1)}) \in \widehat{A}_{\epsilon,t}^{(N)} \setminus B_t^{(N)}$ for $t > 1$, and we sum over all $(\underline{\mathbf{x}}_t, \underline{\mathbf{y}}_t)$ for $t > 1$, over all $(\underline{\mathbf{x}}_1^{(*,0)}, \underline{\mathbf{y}}_1^{(*,0)})$, and over all $(\underline{\tilde{w}}, \underline{\hat{\tilde{w}}})$; the remaining sum is over all $(\underline{w}, \underline{x}_1^{(1,1)}, \underline{y}_1^{(2,1)}, \underline{\hat{w}})$ for which $\underline{w} \neq \underline{\hat{w}}$ and $(\underline{x}_1^{(1,1)}, \underline{y}_1^{(2,1)}) \in \widehat{A}_{\epsilon,1}^{(N)} \setminus B_1^{(N)}$. Then (d) follows from (7) and the bound $p_1(\widehat{A}_{\epsilon,1}^{(N)} \setminus B_1^{(N)}) \leq 1$.

Step 5 – Choose parameters: The final step is to show that we can choose typical set parameters $\epsilon = (\epsilon(1), \ldots, \epsilon(n))$ such that $\widehat{\Pr}(\underline{\widehat{W}} \neq \underline{W}) < \lambda$ for all N sufficiently large. Since n is

fixed and finite, plugging (14), (16), and (17) into (8), (9), and (10) shows that the expected error probability of $\mathcal{S}(\underline{\mathcal{N}}^R)$ goes to zero provided

$$\sum_{t'=1}^{t-1}(4a(\epsilon,t')+2\epsilon(t')+1/N)<c(\epsilon,t)$$

$$2a(\epsilon,t)+\epsilon(t)<R-I(X_t^{(1,1)};Y_t^{(2,1)})$$

$$\sum_{t'=1}^{n}(4a(\epsilon,t')+2\epsilon(t')+1/N)<\delta/2$$

for all $t\in\{1,\dots,n\}$. Recall that constants $a(\epsilon,t)$ and $c(\epsilon,t)$, which are defined in (19) and Lemma 8 in Appendix II, depend only on distribution $p_t(x^{(1,1)},y^{(2,1)})$ and the value $\epsilon(t)$. Each goes to 0 as $\epsilon(t)$ approaches 0. The following sequential choice of $\epsilon(n),\dots,\epsilon(1)$ yields the desired result. Set $\epsilon(n)$ such that $2a(\epsilon,n)+\epsilon(n)\le\min\{\delta/(8n),(R-I(X_n^{(1,1)};Y_n^{(2,1)}))/2\}$. Given $\epsilon(t+1),\dots,\epsilon(n)$, set $\epsilon(t)$ such that

$$2a(\epsilon,t)+\epsilon(t)\le\min\left\{\frac{\delta}{8n},\frac{R-I(X_t^{(1,1)};Y_t^{(2,1)})}{2},\frac{c(\epsilon,t+1)}{t+1},\dots,\frac{c(\epsilon,n)}{n}\right\}.$$

The resulting bound on the expected probability of the union of all error events goes to zero as N grows without bound since $R>C\ge I(X_t^{(1,1)};Y_t^{(2,1)})$ (by the theorem assumption and definition of capacity) and $\delta>0$. Since the expected probability of the union of all error events goes to zero as N grows without bound, there must exist a single instance of the code $\mathcal{S}(\underline{\mathcal{N}}^R)$ that does at least as well. ■

Remark 3: It is interesting to specify the choice of parameters in Theorems 2 and 6 required to guarantee the existence of a $(\tilde{\lambda},\mathcal{R})$-$\mathcal{S}(\mathcal{N}^R)$ solution for an arbitrary $\tilde{\lambda}>0$ and $\mathcal{R}\in\text{int}(\mathfrak{R}(\mathcal{N}))$. Since we have $\mathcal{R}\in\text{int}(\mathfrak{R}(\mathcal{N}))$ there exists an $\tilde{\mathcal{R}}\in\text{int}(\mathfrak{R}(\mathcal{N}))$ with $\tilde{\mathcal{R}}>\mathcal{R}$. We choose ρ in Theorem 2 accordingly as $\min_{u,V}\{\tilde{R}^{(\{u\}\to V)}-R^{(\{u\}\to V)}\}$. Once ρ is chosen, we choose λ and n so that the condition $\rho>\max_{u,V}\{\tilde{R}^{(\{u\}\to V)}\}\lambda+h(\lambda)/n$ is satisfied for a $(\lambda,\mathcal{R})$-$\mathcal{S}(\mathcal{N})$ solution of blocklength n. Note that for each $(u,V)\in\mathcal{M}$ and each $v\in V$, $R^{(\{u\}\to V)}$ is less than the capacity of the channel $p(\widehat{\tilde{w}}^{(\{u\}\to V),v}|\tilde{w}^{(\{u\}\to V)})$ imposed by this solution, so $\delta>0$. Fixing $\mathcal{S}(\mathcal{N})$ fixes distributions $p_t(x^{(1,1)})$. We next choose ϵ as specified above and design channel emulator $(\alpha_{N,t},\beta_{N,t})$ for N sufficiently large. The given blocklength-n solution for the N-fold stacked network $\underline{\mathcal{N}}^R$ can be unraveled and run as a blocklength-nN solution on the single-layer network $\mathcal{N}^R$ as described in the first half of the proof of Lemma 1. ■

VII. Conclusions

The equivalence tools introduced in this work suggest a new path towards the construction of computational tools for bounding the capacities of large networks. Unlike cut-set strategies, which investigate networks in their entirety, the approach here is to bound capacities of networks by bounding the behaviors of component channels. We here demonstrate the strategy when the network components are point-to-point channels. In that case, we present the first proof of the separation between network and channel coding for general connection types, demonstrating the equivalence between the capacity of a network of noisy point-to-point channels and the capacity of another network where each noisy channel is replaced by a noiseless bit pipe of the same capacity. Part II applies the same strategy to multiterminal channels, first deriving standards of emulation accuracy under which a network of bit pipes is an upper or lower bounding model for a multiterminal channel and then applying those standards to derive bit-pipe models for a variety of multiterminal channels. Using bounding networks constructed from noiseless bit pipes allows us to apply available computational tools for bounding network coding capacities to networks constructed from noisy component channels.

Appendix I
Lemma 7

Lemma 7: Under the random code design in the proof of Theorem 2, for each time $t\in\{1,\dots,n\}$ there exists a distribution $p_t(\mathbf{x},\mathbf{y})$ independent of N such that

$$E[\Pr\left((\underline{\mathbf{X}}_t,\underline{\mathbf{Y}}_t)=(\underline{\mathbf{x}},\underline{\mathbf{y}})\right)]=\prod_{\ell=1}^{N}p_t(\underline{\mathbf{x}}(\ell),\underline{\mathbf{y}}(\ell));$$

the expectation is over the random channel code design.

Proof: The stacked solution design combines a single-layer solution $\mathcal{S}(\mathcal{N})$ with a collection of channel codes. The solution $\mathcal{S}(\mathcal{N})$ is fixed for all N. The channel codes are independently and randomly designed for each N. Fix N and the instance of the channel codes. Then $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ establishes distribution

$$\begin{aligned}&p(\underline{w},\underline{\tilde{w}},\underline{\mathbf{x}}^n,\underline{\mathbf{y}}^n,\underline{\widehat{\tilde{w}}},\underline{\widehat{w}})\\&=p(\underline{w})p(\underline{\tilde{w}}|\underline{w})\left[\prod_{t=1}^{n}p(\underline{\mathbf{x}}_t|\underline{\mathbf{y}}^{t-1},\underline{\tilde{w}})p(\underline{\mathbf{y}}_t|\underline{\mathbf{x}}_t)\right]\\&\quad\cdot p(\underline{\widehat{\tilde{w}}}|\underline{\mathbf{y}}^n,\underline{\tilde{w}})p(\underline{\widehat{w}}|\underline{\widehat{\tilde{w}}})\end{aligned}\tag{18}$$

where $p(\underline{w})$ is the uniform distribution on messages, $p(\underline{\tilde{w}}|\underline{w})$ captures the operation of the message channel encoders, $p(\underline{\mathbf{x}}_t|\underline{\mathbf{y}}^{t-1},\underline{\tilde{w}})$ describes the operation of the time-t node encoders, $p(\underline{\mathbf{y}}_t|\underline{\mathbf{x}}_t)$ is the network distribution, $p(\underline{\widehat{\tilde{w}}}|\underline{\mathbf{y}}^n,\underline{\tilde{w}})$ results from the node decoders, and $p(\underline{\widehat{w}}|\underline{\widehat{\tilde{w}}})$ describes the operation of the message channel decoders.[13]

Many of these distributions factor across the layers of the stack. The messages are drawn independently and uniformly at random, giving $p(\underline{w})=\prod_{\ell=1}^{N}p(\underline{w}(\ell))$. Stacked solution $\underline{\mathcal{S}}(\underline{\mathcal{N}})$ applies solution $\mathcal{S}(\mathcal{N})$ independently in each layer of the stack, giving

$$p(\underline{\mathbf{x}}_t|\underline{\mathbf{y}}^{t-1},\underline{\tilde{w}})=\prod_{\ell=1}^{N}p(\underline{\mathbf{x}}_t(\ell)|\underline{\mathbf{y}}^{t-1}(\ell),\underline{\tilde{w}}(\ell))$$

$$p(\underline{\widehat{\tilde{w}}}|\underline{\mathbf{y}}^n,\underline{\tilde{w}})=\prod_{\ell=1}^{N}p(\underline{\widehat{\tilde{w}}}(\ell)|\underline{\mathbf{y}}^n(\ell),\underline{\tilde{w}}(\ell))$$

[13]Any fixed code has deterministic encoders and decoders; thus $p(\underline{\tilde{w}}|\underline{w}),p(\underline{\mathbf{x}}_t|\underline{\mathbf{y}}^{t-1},\underline{\tilde{w}}),p(\underline{\widehat{\tilde{w}}}|\underline{\mathbf{y}}^n,\underline{\tilde{w}}),p(\underline{\widehat{w}}|\underline{\widehat{\tilde{w}}})\in\{0,1\}$. For example, $p(\underline{\tilde{w}}|\underline{w})$ equals 1 if $\underline{\tilde{W}}(\underline{w})=\underline{\tilde{w}}$ and 0 otherwise.

where $p(\underline{\mathbf{x}}_t(\ell)|\underline{\mathbf{y}}^{t-1}(\ell), \underline{\tilde{w}}(\ell))$ and $p(\underline{\widehat{\tilde{w}}}(\ell)|\underline{\mathbf{y}}^{n}(\ell), \underline{\tilde{w}}(\ell))$ characterize the node encoders and decoders for $\mathcal{S}(\mathcal{N})$. The stacked network definition sets $p(\underline{\mathbf{y}}_t|\underline{\mathbf{x}}_t) = \prod_{\ell=1}^{N} p(\underline{\mathbf{y}}_t(\ell)|\underline{\mathbf{x}}_t(\ell))$. Thus

$$p(\underline{w},\underline{\tilde{w}},\underline{\mathbf{x}}^n,\underline{\mathbf{y}}^n,\underline{\widehat{\tilde{w}}},\underline{\widehat{w}}) = p(\underline{\tilde{w}}|\underline{w})p(\underline{\widehat{w}}|\underline{\widehat{\tilde{w}}}) \cdot \prod_{\ell=1}^{N}\left[p(\underline{w}(\ell))p(\underline{\widehat{\tilde{w}}}(\ell)|\underline{\mathbf{y}}^n(\ell),\underline{\tilde{w}}(\ell)) \cdot \prod_{t=1}^{n}\left[p(\underline{\mathbf{x}}_t(\ell)|\underline{\mathbf{y}}^{t-1}(\ell),\underline{\tilde{w}}(\ell))p(\underline{\mathbf{y}}_t(\ell)|\underline{\mathbf{x}}_t(\ell))\right]\right].$$

To calculate $\Pr((\underline{\mathbf{X}}_t, \underline{\mathbf{Y}}_t) = (\underline{\mathbf{x}}, \underline{\mathbf{y}}))$ for each t, we focus on the marginal

$$p(\underline{w},\underline{\tilde{w}},\underline{\mathbf{x}}^n,\underline{\mathbf{y}}^n) = p(\underline{\tilde{w}}|\underline{w})\prod_{\ell=1}^{N}\left[p(\underline{w}(\ell)) \cdot \prod_{t=1}^{n}\left[p(\underline{\mathbf{x}}_t(\ell)|\underline{\mathbf{y}}^{t-1}(\ell),\underline{\tilde{w}}(\ell))p(\underline{\mathbf{y}}_t(\ell)|\underline{\mathbf{x}}_t(\ell))\right]\right].$$

Taking the expectation over the random channel code design gives

$$E[p(\underline{w},\underline{\tilde{w}},\underline{\mathbf{x}}^n,\underline{\mathbf{y}}^n)] = E[p(\underline{\tilde{w}}|\underline{w})]\prod_{\ell=1}^{N}\left[p(\underline{w}(\ell)) \cdot \prod_{t=1}^{n}\left[p(\underline{\mathbf{x}}_t(\ell)|\underline{\mathbf{y}}^{t-1}(\ell),\underline{\tilde{w}}(\ell))p(\underline{\mathbf{y}}_t(\ell)|\underline{\mathbf{x}}_t(\ell))\right]\right].$$

The channel codeword for each $\underline{w}$ is chosen independently and uniformly at random, so $E[p(\underline{\tilde{w}}|\underline{w})] = \prod_{\ell=1}^{N} p(\underline{\tilde{w}}(\ell))$ is the uniform distribution on $\underline{\tilde{\mathcal{W}}}$, and

$$E[p(\underline{w},\underline{\tilde{w}},\underline{\mathbf{x}}^n,\underline{\mathbf{y}}^n)] = \prod_{\ell=1}^{N}\left[p(\underline{w}(\ell))p(\underline{\tilde{w}}(\ell)) \cdot \prod_{t=1}^{n}\left[p(\underline{\mathbf{x}}_t(\ell)|\underline{\mathbf{y}}^{t-1}(\ell),\underline{\tilde{w}}(\ell))p(\underline{\mathbf{y}}_t(\ell)|\underline{\mathbf{x}}_t(\ell))\right]\right].$$

The resulting marginals are

$$E[p(\underline{\mathbf{x}}_t, \underline{\mathbf{y}}_t)] = \prod_{\ell=1}^{N} p_t(\underline{\mathbf{x}}_t(\ell), \underline{\mathbf{y}}_t(\ell)),$$

where $p_t(\mathbf{x}, \mathbf{y})$ is the distribution on $(\mathbf{X}_t, \mathbf{Y}_t)$ established by solution $\mathcal{S}(\mathcal{N})$ when operated on messages uniformly distributed on $\tilde{\mathcal{W}}$. This distribution is independent of N, which gives the desired result. ■

APPENDIX II
SUPPORTING MATERIALS FOR THE PROOF OF THEOREM 6

Let $\epsilon = (\epsilon(1), \ldots, \epsilon(n))$ be a vector of positive constants, and for each t define

$$f_t(\underline{x}^{(1,1)}) \stackrel{\text{def}}{=} \left|-\frac{1}{N}\log p_t(\underline{x}^{(1,1)}) - H(X_t^{(1,1)})\right|$$

$$f_t(\underline{y}^{(2,1)}) \stackrel{\text{def}}{=} \left|-\frac{1}{N}\log p_t(\underline{y}^{(2,1)}) - H(Y_t^{(2,1)})\right|$$

and

$$f_t(\underline{x}^{(1,1)}, \underline{y}^{(2,1)}) \stackrel{\text{def}}{=} \left|-\frac{1}{N}\log p_t(\underline{x}^{(1,1)}, \underline{y}^{(2,1)}) - H(X_t^{(1,1)}, Y_t^{(2,1)})\right|$$

where $H(X_t^{(1,1)})$ is the entropy of $X_t^{(1,1)}$, $H(Y_t^{(2,1)})$ is the entropy of $Y_t^{(2,1)}$, and $H(X_t^{(1,1)}, Y_t^{(2,1)})$ is the entropy of $(X_t^{(1,1)}, Y_t^{(2,1)})$ under $p_t(x^{(1,1)}, y^{(2,1)})$.[14] For each $t \in \{1, \ldots, n\}$, let

$$A_{\epsilon,t}^{(N)} \stackrel{\text{def}}{=} \Big\{(\underline{x}^{(1,1)}, \underline{y}^{(2,1)}) \in \underline{\mathcal{X}}^{(1,1)} \times \underline{\mathcal{Y}}^{(2,1)} : f_t(\underline{x}^{(1,1)}) \le \epsilon(t),\ \ f_t(\underline{y}^{(2,1)}) \le a(\epsilon,t), \ f_t(\underline{x}^{(1,1)}, \underline{y}^{(2,1)}) \le a(\epsilon,t)\Big\}$$

where

$$a(\epsilon,t) \stackrel{\text{def}}{=} (1+\epsilon(t)) \cdot \inf\{\epsilon' > 0 : \Pr\left(f_t(\underline{Y}_t^{(2,1)}) > \epsilon' \vee f_t(\underline{X}_t^{(1,1)}, \underline{Y}_t^{(2,1)}) > \epsilon'\right) \le 2^{-6N\epsilon(t)} \text{ for all } N \text{ sufficiently large}\}. \quad (19)$$

This infimum is shown to be well defined in the proof of Lemma 8. Define set

$$\widehat{A}_{\epsilon,t}^{(N)} \stackrel{\text{def}}{=} \Big\{(\underline{x}^{(1,1)}, \underline{y}^{(2,1)}) \in A_{\epsilon,t}^{(N)} : p\left((A_{\epsilon,t}^{(N)})^c \Big| \underline{x}^{(1,1)}\right) \le 2^{-3N\epsilon(t)}\Big\}$$

where

$$p((A_{\epsilon,t}^{(N)})^c|\underline{x}^{(1,1)}) \stackrel{\text{def}}{=} \sum_{\underline{y}^{(2,1)}:(\underline{x}^{(1,1)},\underline{y}^{(2,1)}) \notin A_{\epsilon,t}^{(N)}} p(\underline{y}^{(2,1)}|\underline{x}^{(1,1)}).$$

We henceforth call $\widehat{A}_{\epsilon,t}^{(N)}$ the typical set. This definition restricts attention to those typical channel inputs $\underline{x}^{(1,1)}$ that are most likely to yield jointly typical channel outputs. This restriction is later useful for showing that the number of jointly typical channel outputs for each typical channel input is roughly the same. Such a result could be obtained more directly for finite-alphabet channels using strong typicality; we here treat the general case. Lemma 8, below, proves that $p_t((\widehat{A}_{\epsilon,t}^{(N)})^c)$ approaches zero as N grows without bound.

Lemma 8: Let $(\underline{X}(1), \underline{Y}(1)), \ldots, (\underline{X}(N), \underline{Y}(N))$ be drawn i.i.d. according to distribution $p_t(x, y)$ on alphabet $\mathcal{X}^{(1,1)} \times \mathcal{Y}^{(2,1)}$. Then there exists a constant $c(\epsilon, t) > 0$ for which

$$p_t((\widehat{A}_{\epsilon,t}^{(N)})^c) < 2^{-Nc(\epsilon,t)}$$

for all N sufficiently large. Constant $c(\epsilon, t)$ approaches 0 as $\epsilon(t) > 0$ approaches 0.

[14] We use notation $H(\cdot)$ for both discrete and differential entropy and assume that $H(X_t^{(1,1)}, Y_t^{(2,1)}) < \infty$.

Proof: The result follows from Chernoff's bound, which we apply to averages of i.i.d. random variables. Chernoff's bound states that for any i.i.d. random variables $A(1), A(2), \ldots, A(N)$,

$$\Pr\left(\frac{1}{N}\sum_{\ell=1}^{N} A(\ell) > a\right) \leq e^{N \min_{s>0}[M(s)-sa]}$$

where $M(s) \stackrel{\text{def}}{=} \ln E[e^{sA}]$ and $\min_{s>0}[M(s) - sa] \leq 0$ for all $a \geq E[A]$ with equality if and only if $a = E[A]$ (see, for example, [23, pp. 482–484]). Note that $|\min_{s>0}[M(s) - sa]|$ approaches 0 as a approaches $E[A]$.

We begin by applying the Chernoff bound to the following sequence of random variables

$$-\log p_t(\underline{X}(1)), \ldots, -\log p_t(\underline{X}(N)).$$

We then negate the sequence and apply the Chernoff bound again. Combining these results with the union bound gives

$$\Pr\left(f_t(\underline{X}) > \epsilon(t)\right) \leq 2^{-Nb_0}$$

for some $b_0 > 0$ and all N sufficiently large. Likewise, for any $\epsilon' > 0$,

$$\begin{aligned}\Pr\left(f_t(\underline{Y}) > \epsilon'\right) &\leq 2^{-Nb_1}\\ \Pr\left(f_t(\underline{X},\underline{Y}) > \epsilon'\right) &\leq 2^{-Nb_2}\end{aligned}$$

for some $b_1, b_2 > 0$ and all N sufficiently large. Since $|\min_{s>0}[M(s) - sa]|$ grows without bound as a increases, constants b_1 and b_2 can be made arbitrarily large by choosing ϵ' large enough. This implies that the infimum in the definition of $a(\epsilon, t)$ ((19), above) is well-defined.

Applying these bounds gives

$$\begin{aligned}&p_t((A_{\epsilon,t}^{(N)})^c)\\ &\quad\stackrel{(a)}{\leq} \Pr\left(f_t(\underline{X}) > \epsilon(t)\right)\\ &\qquad + \Pr\left(f_t(\underline{Y}) > a(\epsilon,t) \vee f_t(\underline{X},\underline{Y}) > a(\epsilon,t)\right)\\ &\quad\stackrel{(b)}{\leq} 2^{-Nb_0} + 2^{-6N\epsilon(t)}\end{aligned}$$

where (a) applies the union bound and the definition of $A_{\epsilon,t}^{(N)}$, and (b) follows from our first Chernoff bound and the definition of $a(\epsilon, t)$ in (19). Let

$$C_t^{(N)} \stackrel{\text{def}}{=} \left\{\underline{x} \in \underline{\mathcal{X}}^{(1,1)} : f_t(\underline{x}) \leq \epsilon(t), \; p\left((A_{\epsilon,t}^{(N)})^c \middle| \underline{x}\right) > 2^{-3N\epsilon(t)}\right\}.$$

Then, by the definition of (restricted) typical set $\widehat{A}_{\epsilon,t}^{(N)}$,

$$\begin{aligned}&p_t\left((\widehat{A}_{\epsilon,t}^{(N)})^c\right) - p_t\left((A_{\epsilon,t}^{(N)})^c\right)\\ &= p_t\left(\{(\underline{x},\underline{y}) \in A_{\epsilon,t}^{(N)} : \underline{x} \in C_t^{(N)}\}\right)\\ &= \sum_{\underline{x}\in C_t^{(N)}} p_t(\underline{x}) p\left(A_{\epsilon,t}^{(N)} \middle| \underline{x}\right)\\ &\stackrel{(a)}{\leq} p_t\left(C_t^{(N)}\right)\end{aligned}$$

where (a) follows since $p(A_{\epsilon,t}^{(N)}|\underline{X}_t = \underline{x}) \leq 1$. To bound $p_t(C_t^{(N)})$, note that

$$\begin{aligned}2^{-6N\epsilon(t)} &\stackrel{(a)}{\geq} \sum_{\underline{x}} p_t(\underline{x}) \Pr\left(f_t(\underline{Y}_t) > a(\epsilon,t)\right.\\ &\qquad \left. \vee\, f_t(\underline{X}_t,\underline{Y}_t) > a(\epsilon,t) \,|\underline{X}_t = \underline{x}\right)\\ &\stackrel{(b)}{\geq} \sum_{\underline{x}\in C_t^{(N)}} p_t(\underline{x}) \Pr\left(f_t(\underline{Y}_t) > a(\epsilon,t)\right.\\ &\qquad \left. \vee\, f_t(\underline{X}_t,\underline{Y}_t) > a(\epsilon,t) \,|\underline{X}_t = \underline{x}\right)\\ &\stackrel{(c)}{>} p_t(C_t^{(N)}) 2^{-3N\epsilon(t)}\end{aligned}$$

where (a) follows from the definition of $a(\epsilon, t)$, (b) follows from the non-negativity of probability since we restrict the terms of the summation, and (c) applies the definition of $C_t^{(N)}$. Thus $p_t(C_t^{(N)}) < 2^{-3N\epsilon(t)}$, which gives the desired result. ■

Lemma 11, which follows, bounds the expected emulation distribution $E[\widehat{p}_t(\underline{y}^{(2,1)}|\underline{x}^{(1,1)})]$ resulting from the random ensemble of emulators $(\alpha_{N,t}, \beta_{N,t})$. (Recall that $E[\widehat{p}_t(\underline{y}^{(2,1)}|\underline{x}^{(1,1)})]$ is defined in (4) in Section VI.) Lemmas 9 and 10 are intermediate steps used in the proof of Lemma 11.

For any $(\underline{x},\underline{y}) \in \underline{\mathcal{X}}^{(1,1)} \times \underline{\mathcal{Y}}^{(2,1)}$, define functions $K_t(\underline{x},\underline{y})$ and $q_t(\underline{x})$ as

$$\begin{aligned}K_t(\underline{x},\underline{y}) &\stackrel{\text{def}}{=} \begin{cases}1, & \text{if } (\underline{x},\underline{y}) \in \widehat{A}_{\epsilon,t}^{(N)}\\ 0, & \text{otherwise}\end{cases}\\ q_t(\underline{x}) &\stackrel{\text{def}}{=} \sum_{\underline{y}\in\underline{\mathcal{Y}}^{(2,1)}} K_t(\underline{x},\underline{y}) p_t(\underline{y}).\end{aligned} \tag{20}$$

Lemma 9 characterizes $E[\widehat{p}_t(\underline{y}|\underline{x})]$ as a function of the probability $q_t(\underline{x})$ that a single codeword drawn at random is jointly typical with $\underline{x}$; specifically, the lemma shows that $p_t(\underline{y})/q_t(\underline{x})$ is the expected probability that $\underline{x}$ is mapped to $\underline{y}$ given that there is at least one codeword in the codebook that is typical with $\underline{x}$. Lemma 10 then bounds $q_t(\underline{x})$ for all $\underline{x}$ satisfying the conditions of $\widehat{A}_{\epsilon,t}^{(N)}$. Our restriction in the typical set definition is useful here.

Lemma 9: Let $(\alpha_{N,t}, \beta_{N,t})$ be the random emulator defined in (2) and (3). Then for any $(\underline{x},\underline{y}) \in \widehat{A}_{\epsilon,t}^{(N)}$,

$$E[\widehat{p}_t(\underline{y}|\underline{x})] = \frac{p_t(\underline{y})}{q_t(\underline{x})}\left(1 - (1 - q_t(\underline{x}))^{2^{NR}}\right).$$

Proof: Recall that $q_t(\underline{x})$ is the probability that a single randomly drawn codeword $\underline{Y} \in \underline{\mathcal{Y}}^{(2,1)}$ satisfies $(\underline{x},\underline{Y}) \in \widehat{A}_{\epsilon,t}^{(N)}$. Using the given random code design, for any $(\underline{x},\underline{y}) \in \widehat{A}_{\epsilon,t}^{(N)}$,

$$\begin{aligned}E[\widehat{p}_t(\underline{y}|\underline{x})] &= \sum_{j=1}^{2^{NR}}\sum_{k=1}^{j}\binom{2^{NR}}{j}\binom{j}{k}(1 - q_t(\underline{x}))^{2^{NR}-j}\\ &\qquad \cdot (q_t(\underline{x}) - p_t(\underline{y}))^{j-k}(p_t(\underline{y}))^k \frac{k}{j}\\ &= p_t(\underline{y})\sum_{j=1}^{2^{NR}}\binom{2^{NR}}{j}\frac{1}{j}(1 - q_t(\underline{x}))^{2^{NR}-j}\\ &\qquad \cdot \sum_{k=1}^{j}\binom{j}{k}[a^{j-k}kb^{k-1}].\end{aligned}$$

Here, j is the number of codewords that are jointly typical with $\underline{x}$, k is the number of those codewords that equal $\underline{y}$, and term k/j follows from the uniform distribution over jointly typical codewords in the encoder design. In the second equality, $a = q_t(\underline{x}) - p_t(\underline{y})$ and $b = p_t(\underline{y})$. Thus

$$\begin{aligned}E[\widehat{p}_t(\underline{y}|\underline{x})] &= p_t(\underline{y}) \sum_{j=1}^{2^{NR}} \binom{2^{NR}}{j} \frac{1}{j} (1-q_t(\underline{x}))^{2^{NR}-j} \\ &\quad \cdot \frac{\partial}{\partial b}[(a+b)^j - a^j] \\ &= p_t(\underline{y}) \sum_{j=1}^{2^{NR}} \binom{2^{NR}}{j} \frac{1}{j} (1-q_t(\underline{x}))^{2^{NR}-j} \\ &\quad \cdot j(q_t(\underline{x}))^{j-1} \\ &= \frac{p_t(\underline{y})}{q_t(\underline{x})} \left(1-(1-q_t(\underline{x}))^{2^{NR}}\right).\end{aligned}$$

■

Lemma 10: Given $\underline{x} \in \underline{\mathcal{X}}^{(1,1)}$, if $f_t(\underline{x}) \le \epsilon(t)$ and $p((A_{\epsilon,t}^{(N)})^c|\underline{x}) < 2^{-3N\epsilon(t)}$, then

$$q_t(\underline{x}) \ge 2^{-N(I(X_t^{(1,1)};Y_t^{(2,1)})+\epsilon(t)+2a(\epsilon,t)+\frac{1}{N})}$$

for all N sufficiently large.

Proof: For any $\underline{x}$ satisfying the given constraints, we first derive a bound on the number of $\underline{y}$ values for which $(\underline{x},\underline{y}) \in \widehat{A}_{\epsilon,t}^{(N)}$. This is obtained by drawing a random variable $\underline{Y}$ according to conditional distribution $\prod_{\ell=1}^{N} p_t(\underline{y}(\ell)|\underline{x}(\ell))$ and showing that $(\underline{x},\underline{Y}) \in \widehat{A}_{\epsilon,t}^{(N)}$ with probability approaching 1. Since all $\underline{y}$ that are jointly typical with $\underline{x}$ are approximately equally probable, this probability bound leads to a bound on the number of $\underline{y}$ vectors that are jointly typical with $\underline{x}$ and then to a bound on the desired probability.

By the lemma assumptions,

$$\Pr((\underline{X},\underline{Y}) \notin A_{\epsilon,t}^{(N)}|\underline{X}=\underline{x}) < 2^{-3N\epsilon(t)}$$

which approaches 0 as N grows without bound. Thus for N sufficiently large, $p(A_{\epsilon,t}^{(N)}|\underline{x}) \ge 1/2$. Let $F_t(\underline{x}) \stackrel{\text{def}}{=} \{\underline{y} \in \underline{\mathcal{Y}}^{(2,1)} : (\underline{x},\underline{y}) \in A_{\epsilon,t}^{(N)}\}$. Then

$$\begin{aligned}\frac{1}{2} &\le p(A_{\epsilon,t}^{(N)}|\underline{x}) \\ &\le |F_t(\underline{x})| 2^{-N(H(Y_t^{(2,1)}|X_t^{(1,1)})-a(\epsilon,t)-\epsilon(t))}\end{aligned}$$

since $(\underline{x},\underline{y}) \in A_{\epsilon,t}^{(N)}$ implies

$$p(\underline{y}|\underline{x}) \le \frac{2^{-N(H(X_t^{(1,1)},Y_t^{(2,1)})-a(\epsilon,t))}}{2^{-N(H(X_t^{(1,1)})+\epsilon(t))}}.$$

Thus

$$|F_t(\underline{x})| \ge 2^{N(H(Y_t^{(2,1)}|X_t^{(1,1)})-a(\epsilon,t)-\epsilon(t)-1/N)}$$

which we use to bound $q_t(\underline{x})$ as

$$\begin{aligned}q_t(\underline{x}) &= \sum_{\underline{y} \in F_t(\underline{x})} p_t(\underline{y}) \\ &\ge |F_t(\underline{x})| 2^{-N(H(Y_t^{(2,1)})+a(\epsilon,t))} \\ &\ge 2^{-N(I(X_t^{(1,1)};Y_t^{(2,1)})+2a(\epsilon,t)+\epsilon(t)+1/N)}.\end{aligned}$$

■

Lemma 11: For all $(\underline{x},\underline{y}) \in \widehat{A}_{\epsilon,t}^{(N)}$,

$$\widehat{p}_t(\underline{y}|\underline{x}) \le p(\underline{y}|\underline{x}) 2^{N(4a(\epsilon,t)+2\epsilon(t)+1/N)}.$$

Proof: By Lemmas 9 and 10 and the usual bounds on the probabilities of typical elements

$$\begin{aligned}\widehat{p}_t(\underline{y}|\underline{x}) &= \frac{p_t(\underline{y})}{q_t(\underline{x})} \left(1-(1-q_t(\underline{x}))^{2^{NR}}\right) \le \frac{p_t(\underline{y})}{q_t(\underline{x})} \\ &\le \frac{p_t(\underline{y}) p(\underline{y}|\underline{x}) p_t(\underline{x}) / p_t(\underline{x},\underline{y})}{2^{-N(I(X_t^{(1,1)};Y_t^{(2,1)})+2a(\epsilon,t)+\epsilon(t)+1/N)}} \\ &\le \frac{p(\underline{y}|\underline{x}) 2^{-N(I(X_t^{(1,1)};Y_t^{(2,1)})-2a(\epsilon,t)-\epsilon(t))}}{2^{-N(I(X_t^{(1,1)};Y_t^{(2,1)})+2a(\epsilon,t)+\epsilon(t)+1/N)}} \\ &= p(\underline{y}|\underline{x}) 2^{N(4a(\epsilon,t)+2\epsilon(t)+1/N)}.\end{aligned}$$

■

Lemma 12 bounds the conditional probability that $(\underline{X}_t,\underline{Y}_t)$ is not jointly typical when the conditional distribution on $\underline{Y}_t$ given $\underline{X}_t$ is the distribution $\widehat{p}_t(\underline{y}_t|\underline{x}_t)$ resulting from the randomly designed emulator $(\alpha_{N,t},\beta_{N,t})$.

Lemma 12: For all $\underline{x} \in \underline{\mathcal{X}}^{(1,1)}$,

$$\begin{aligned}E\left[\widehat{p}_t((\widehat{A}_{\epsilon,t}^{(N)})^c|\underline{x})\right] &\le p((\widehat{A}_{\epsilon,t}^{(N)})^c|\underline{x}) \\ &\quad + e^{-2^{N(R-I(X_t^{(1,1)};Y_t^{(2,1)})-2a(\epsilon,t)-\epsilon(t))}}.\end{aligned}$$

Proof: If $f_t(\underline{x}) > \epsilon(t)$ or $p((A_{\epsilon,t}^{(N)})^c|\underline{x}) > 2^{-3N\epsilon(t)}$, then

$$E\left[\widehat{p}_t((\widehat{A}_{\epsilon,t}^{(N)})^c|\underline{x})\right] = p((\widehat{A}_{\epsilon,t}^{(N)})^c|\underline{x}) = 1$$

by definition of $\widehat{A}_{\epsilon,t}^{(N)}$, and the result is immediate. Otherwise, $(\underline{x},\underline{y}) \notin \widehat{A}_{\epsilon,t}^{(N)}$ implies that none of the 2^{NR} codewords of $\beta_{N,t}$ is jointly typical with $\underline{x}$. In this case, using definition (20) and following the proof of the rate-distortion theorem (cf. [18, Steps 10.93–10.102]),

$$\begin{aligned}&E\left[\widehat{p}_t((\widehat{A}_{\epsilon,t}^{(N)})^c|\underline{x})\right] \\ &= \left(1-\sum_{\underline{y}} p_t(\underline{y}) K_t(\underline{x},\underline{y})\right)^{2^{NR}} \\ &\stackrel{(a)}{\le} 1-\sum_{\underline{y}} p(\underline{y}|\underline{x}) K_t(\underline{x},\underline{y}) \\ &\quad + e^{-2^{N(R-I(X_t^{(1,1)};Y_t^{(2,1)})-2a(\epsilon,t)-\epsilon(t))}} \\ &= p_t((\widehat{A}_{\epsilon,t}^{(N)})^c|\underline{X}^{(1,1)}=\underline{x}) \\ &\quad + e^{-2^{N(R-I(X_t^{(1,1)};Y_t^{(2,1)})-2a(\epsilon,t)-\epsilon(t))}}.\end{aligned}$$

Here, (a) follows from $(1-ab)^k \leq 1-a+e^{-bk}$[18, Lemma 10.5.3] and the bound

$$\begin{aligned} p_t(\underline{y}) &= p(\underline{y}|\underline{x})\frac{p_t(\underline{y})p_t(\underline{x})}{p_t(\underline{x},\underline{y})} \\ &\geq p(\underline{y}|\underline{x})2^{-N(I(X_t^{(1,1)};Y_t^{(2,1)})+2a(\epsilon,t)+\epsilon(t))} \end{aligned}$$

for all $(\underline{x},\underline{y}) \in \widehat{A}_{\epsilon,t}^{(N)}$, which follows from the definition of the typical set. ■

ACKNOWLEDGMENT

R. Koetter was with the Technical University of Munich when the early drafts of this work were written. He spent many of his last working hours pursuing its completion with the loving support of his wife Nuala, to whom this paper is dedicated. The final manuscript was completed after his passing. His co-authors take full responsibility for any errors or omissions.

REFERENCES

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. IT-46, pp. 1204–1216, Jul. 2000.

[2] A. R. Lehman and E. Lehman, "Complexity classifications of network information flow problems," in *Allerton Conf. on Communications, Control, and Computing*, Monticello, IL, Sep. 2003.

[3] T. Chan and A. Grant, "Dualities between entropy functions and network codes," *IEEE Trans. Inf. Theory*, vol. 54, pp. 4470–4487, Oct. 2008.

[4] L. Song, R. W. Yeung, and N. Cai, "Zero-error network coding for acyclic networks," *IEEE Trans. Inf. Theory*, vol. 49, pp. 3129–3139, Jul. 2003.

[5] N. Harvey, R. Kleinberg, and A. R. Lehman, "On the capacity of information networks," *IEEE Trans. Inf. Theory*, vol. 52, pp. 2345–2364, Jun. 2006.

[6] A. Subramanian and A. Thangaraj, "A simple algebraic formulation for the scalar linear network coding problem," *ArXiv e-Prints*, Jul. 2008.

[7] S. Borade, "Network information flow: Limits and achievability," in *Proc. 2002 IEEE Int. Symp. Information Theory*, Jul. 2002, p. 139.

[8] R. Koetter and M. Médard, "An algebraic approach to network coding," *IEEE/ACM Trans. Netw.*, vol. 11, pp. 782–795, Oct. 2003.

[9] N. Harvey and R. Kleinberg, "Tighter cut-set bounds for k-pairs communication problems," in *Proc. Allerton Conf. Communication, Control, and Computing*, Monticello, IL, Sep. 2005.

[10] G. Kramer and S. Savari, "Capacity bounds for relay networks," in *Information Theory and Applications Workshop*, San Diego, CA, Jan. 2006.

[11] G. Kramer and S. Savari, "Edge-cut bounds on network coding rates," *J. Netw. Syst. Manag.*, vol. 14, pp. 49–67, Mar. 2006.

[12] A. Avestimehr, S. Diggavi, and D. Tse, "Approximate capacity of Gaussian relay networks," in *Proc. 2008 IEEE Int. Symp. Information Theory*, Jul. 2008, pp. 474–478.

[13] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, 1948.

[14] R. Ahlswede, "Multi-way communication channels," in *Proc. 2nd. Int. Symp. Information Theory*, Tsahkadsor, Armenia, 1971, pp. 23–52.

[15] H. Liao, "Multiple Access Channels," Ph. D. dissertation, Dept. Elect. Eng., Univ. Hawaii, Honolulu, 1972.

[16] T. M. Cover, "Broadcast channels," *IEEE Trans. Inf. Theory*, vol. IT-18, pp. 2–14, Jan. 1972.

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

[18] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. Hoboken, NJ: Wiley, 2006.

[19] R. G. Gallager, *Information Theory and Reliable Communication*. New York: Wiley, 1968.

[20] C. Bennett, P. Shor, J. Smolin, and A. Thapliyal, "Entanglement-assisted capacity of a quantum channel and the reverse Shannon theorem," *IEEE Trans. Inf. Theory*, vol. 48, pp. 2637–2655, Oct. 2002.

[21] W.-H. Gu and M. Effros, "A strong converse for a collection of network source coding problems," in *Proc. 2009 IEEE Int. Symp. Information Theory*, Seoul, Korea, Jun. 2009, pp. 2316–2320.

[22] W.-H. Gu, "On Achievable Rate Regions for Source Coding Networks," Ph.D. dissertation, Cal. Inst. Technol., Pasadena, CA, 2009.

[23] U. Madhow, *Fundamentals of Digital Communication*. Cambridge, U.K.: Cambridge Univ. Press, 1998.

Ralf Koetter (S'91–M'96–SM'06–F'09) was born in Königstein im Taunus, Germany, on October 10, 1963. He received a Diploma in Electrical Engineering from the Technische Universität Darmstadt, Germany, in 1990 and the Ph.D. degree from the Department of Electrical Engineering, Linköping University, Sweden, in 1996. From 1996 to 1997, he was a Visiting Scientist at the IBM Almaden Research Center, San Jose, CA. He was a Visiting Assistant Professor at the University of Illinois, Urbana-Champaign, and a Visiting Scientist at CNRS in Sophia-Antipolis, France, from 1997 to 1998. During 1999–2006, he was member of the faculty at the University of Illinois, Urbana-Champaign. In 2006, he joined the faculty of the Technische Universität München, Germany, as the Head of the Institute for Communications Engineering (Lehrstuhl für Nachrichtentechnik). He passed away on February 2, 2009.

His research interests were in coding theory and information theory, and in their applications to communication systems.

During 1999–2001, Prof. Koetter was an Associate Editor for Coding Theory and Techniques for the IEEE TRANSACTIONS ON COMMUNICATIONS, and during 2000–2003, he served as an Associate Editor for Coding Theory for the IEEE TRANSACTIONS ON INFORMATION THEORY. He was Technical Program Co-Chair for the 2008 International Symposium on Information Theory, and twice Co-Editor-in-Chief for special issues of the IEEE TRANSACTIONS ON INFORMATION THEORY. During 2003–2008, he was a member of the Board of Governors of the IEEE Information Theory Society. He received an IBM Invention Achievement Award in 1997, an NSF CAREER Award in 2000, an IBM Partnership Award in 2001, and a Xerox Award for faculty research in 2006. He also received the IEEE Information Theory Society Paper Award in 2004, the Vodafone Innovationspreis in 2008, the Best Paper Award from the IEEE Signal Processing Society in 2008, and the IEEE Communications Society & Information Theory Society Joint Paper Award twice, in 2009 and in 2010.

Michelle Effros (S'93–M'95–SM'03–F'09) received the B.S. degree (with distinction) in 1989, the M.S. degree in 1990, and the Ph.D. degree in 1994, all in electrical engineering, from Stanford University, Stanford, CA. She joined the faculty at the California Institute of Technology, Pasadena, in 1994 and is currently a Professor of Electrical Engineering. Her research interests include information theory, network coding, data compression, and communications.

Prof. Effros received Stanford's Frederick Emmons Terman Engineering Scholastic Award (for excellence in engineering) in 1989, the Hughes Masters Full-Study Fellowship in 1989, the National Science Foundation Graduate Fellowship in 1990, the AT&T Ph.D. Scholarship in 1993, the NSF CAREER Award in 1995, the Charles Lee Powell Foundation Award in 1997, the Richard Feynman-Hughes Fellowship in 1997, an Okawa Research Grant in 2000, and was cited by *Technology Review* as one of the world's top 100 young innovators in 2002. She and her co-authors received the Communications Society and Information Theory Society Joint Paper Award in 2009. She is a member of Tau Beta Pi, Phi Beta Kappa, and Sigma Xi. She served as the Editor of the IEEE INFORMATION THEORY SOCIETY NEWSLETTER from 1995 to 1998 and as a Member of the Board of Governors of the IEEE Information Theory Society from 1998 to 2003 and 2008 to the present. She has been a member of the Advisory Committee for the Computer and Information Science and Engineering (CISE) Directorate at the National Science Foundation from 2009 to the present. She served on the IEEE Signal Processing Society Image and Multi-Dimensional Signal Processing (IMDSP) Technical Committee from 2001 to 2007 and on ISAT from 2006 to 2009. She served as Associate Editor for the joint special issue on Networking and Information Theory in the IEEE TRANSACTIONS ON INFORMATION THEORY and the IEEE/ACM TRANSACTIONS ON NETWORKING and as Associate Editor for Source Coding for the IEEE TRANSACTIONS ON INFORMATION THEORY from 2004 to 2007. She has served on numerous technical program committees and review boards, including serving as general co-chair for the 2009 Network Coding Workshop.

Muriel Médard (S'90–M'95–SM'02–F'08) received B.S. degrees in electrical engineering and computer science as well as mathematics in 1989, the B.S. degree in humanities in 1990, the M.S. degree in electrical engineering in 1991, and the Sc.D. degree in electrical engineering in 1995, all from the Massachusetts Institute of Technology (MIT), Cambridge.

She is a Professor of electrical engineering and computer science at MIT. She was previously an Assistant Professor in the Electrical and Computer Engineering Department and a member of the Coordinated Science Laboratory at the University of Illinois at Urbana-Champaign. From 1995 to 1998, she was a Staff Member in the Optical Communications and the Advanced Networking Groups, MIT Lincoln Laboratory. Her research interests are in the areas of network coding and reliable communications, particularly for optical and wireless networks.

Prof. Médard has served as an Associate Editor for the Optical Communications and Networking Series of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, as an Associate Editor in Communications for the IEEE TRANSACTIONS ON INFORMATION THEORY, and as an Associate Editor for the OSA JOURNAL OF OPTICAL NETWORKING. She has also served as Guest Editor for the IEEE/OSA JOURNAL OF LIGHTWAVE TECHNOLOGY, for the 2006 Joint Special Issue of the IEEE TRANSACTION ON INFORMATION THEORY and the IEEE/ACM TRANSACTIONS ON NETWORKING on "Networking and Information Theory" and for the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY Special Issue on "Statistical Methods for Network Security and Forensics." She serves as an associate editor for the IEEE/OSA JOURNAL OF LIGHTWAVE SCIENCE TECHNOLOGY. She is a recipient of the William R. Bennett Prize in the Field of Communications Networking, the 2002 IEEE Leon K. Kirchmayer Prize Paper Award, and the Best Paper Award at the Fourth International Workshop on the Design of Reliable Communication Networks (DRCN 2003). She received the NSF CAREER Award in 2001 and was corecipient of the 2004 Harold E. Edgerton Faculty Achievement Award at MIT. She was named a 2007 Gilbreth Lecturer by the National Academy of Engineering. She serves as a member of the Board of Governors of the IEEE Information Theory Society.