A Random Linear Network Coding Approach to Multicast

Tracey Ho, *Member, IEEE*, Muriel Médard, *Senior Member, IEEE*, Ralf Koetter, *Senior Member, IEEE*, David R. Karger, *Associate Member, IEEE*, Michelle Effros, *Senior Member, IEEE*, Jun Shi, and Ben Leong

***Abstract*—We present a distributed random linear network coding approach for transmission and compression of information in general multisource multicast networks. Network nodes independently and randomly select linear mappings from inputs onto output links over some field. We show that this achieves capacity with probability exponentially approaching 1 with the code length. We also demonstrate that random linear coding performs compression when necessary in a network, generalizing error exponents for linear Slepian–Wolf coding in a natural way. Benefits of this approach are decentralized operation and robustness to network changes or link failures. We show that this approach can take advantage of redundant network capacity for improved success probability and robustness. We illustrate some potential advantages of random linear network coding over routing in two examples of practical scenarios: distributed network operation and networks with dynamically varying connections. Our derivation of these results also yields a new bound on required field size for centralized network coding on general multicast networks.**

***Index Terms*—Distributed compression, distributed networking, multicast, network coding, random linear coding.**

I. Introduction

THE capacity of multicast networks with network coding was given in [1]. We present an efficient distributed randomized approach that asymptotically achieves this capacity. We consider a general multicast framework—multisource multicast, possibly with correlated sources, on general networks.

Manuscript received February 26, 2004; revised June 1, 2006. This work was supported in part by the National Science Foundation under Grants CCF-0325324, CCR-0325673, and CCR-0220039, by Hewlett-Packard under Contract 008542-008, and by Caltech's Lee Center for Advanced Networking.

T. Ho was with the Laboratory for Information and Decision Systems (LIDS), Massachusetts Institute of Technology (MIT), Cambridge, MA 02139 USA. She is now with the California Institute of Technology (Caltech), Pasadena, CA 91125 USA (e-mail: tho@caltech.edu).

M. Médard is with Laboratory for Information and Decision Systems, the Massachusetts Institute of Technology (MIT), Cambridge, MA 02139 USA (e-mail: medard@mit.edu).

R. Koetter is with the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (e-mail: koetter@csl.uiuc.edu).

D. R. Karger is with the Computer Science and Artificial Intelligence Laboratory (CSAIL), the Massachusetts Institute of Technology (MIT), Cambridge, MA 02139 USA (e-mail: karger@csail.mit.edu).

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

J. Shi was with the University of California, Los Angeles, CA, USA. He is now with Intel Corporation, Santa Clara, CA 95054 USA (e-mail: junshi@ee.ucla.edu).

B. Leong was with the Computer Science and Artificial Intelligence Laboratory (CSAIL), the Massachusetts Institute of Technology (MIT), Cambridge, MA 02139 USA. He is now with the National University of Singapore, Singapore 119260, Republic of Singapore (e-mail: benleong@comp.nus.edu.sg).

Communicated by A. Ashikhmin, Associate Editor for Coding Theory.

Digital Object Identifier 10.1109/TIT.2006.881746

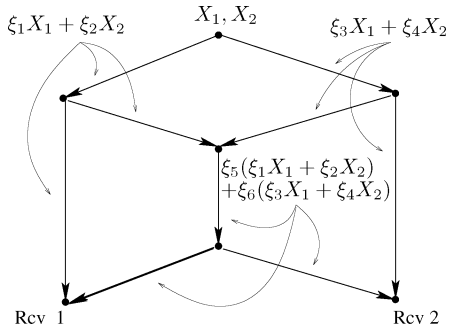

Fig. 1. An example of distributed random linear network coding. X_1 and X_2 are the source processes being multicast to the receivers, and the coefficients ξ_i are randomly chosen elements of a finite field. The label on each link represents the process being transmitted on the link.

This family of problems includes traditional single-source multicast for content delivery and the incast or reachback problem for sensor networks, in which several, possibly correlated, sources transmit to a single receiver. We use a randomized strategy: all nodes other than the receiver nodes perform random linear mappings from inputs onto outputs over some field. These mappings are selected independently at each node. An illustration is given in Fig. 1. The receivers need only know the overall linear combination of source processes in each of their incoming transmissions. This information can be sent with each transmission block or packet as a vector of coefficients corresponding to each of the source processes, and updated at each coding node by applying the same linear mappings to the coefficient vectors as to the information signals. The relative overhead of transmitting these coefficients decreases with increasing length of blocks over which the codes and network remain constant. For instance, if the network and network code are fixed, all that is needed is for the sources to send, once, at the start of operation, a canonical basis through the network.

Our primary results show, first, that such random linear coding achieves multicast capacity with probability exponentially approaching 1 with the length of code. Second, in the context of a distributed source coding problem, we demonstrate that random linear coding also performs compression when necessary in a network, generalizing known error exponents for linear Slepian–Wolf coding [4] in a natural way.

This approach not only recovers the capacity and achievable rates, but also offers a number of advantages. While capacity can be achieved by other deterministic or random approaches, they require, in general, network codes that are planned by or known to a central authority. Random design of network codes was first considered in [1]; our contribution is in showing how random linear network codes can be constructed and efficiently

communicated to receivers in a distributed manner. For the case of distributed operation of a network whose conditions may be varying over time, our work hints at a beguiling possibility: that a network may be operated in a decentralized manner and still achieve the information rates of the optimized solution. Our distributed network coding approach has led to and enabled subsequent developments in distributed network optimization, e.g., [20], [13]. The distributed nature of our approach also ties in well with considerations of robustness to changing network conditions. We show that our approach can take advantage of redundant network capacity for improved success probability and robustness. Moreover, issues of stability, such as those arising from propagation of routing information, are obviated by the fact that each node selects its code independently from the others.

Our results, more specifically, give a lower bound on the probability of error-free transmission for independent or linearly correlated sources, which, owing to the particular form of transfer matrix determinant polynomials, is tighter than the Schwartz–Zippel bound (e.g., [23]) for general polynomials of the same total degree. This bound, which is exponentially dependent on the code length, holds for any feasible set of multicast connections over any network topology (including networks with cycles and link delays). The result is derived using a formulation based on the Edmonds matrix of bipartite matching, which leads also to an upper bound on field size required for deterministic centralized network coding over general networks. We further give, for acyclic networks, tighter bounds based on more specific network structure, and show the effects of redundancy and link reliability on success probability. For arbitrarily correlated sources, we give error bounds for minimum entropy and maximum *a posteriori* probability decoding. In the special case of a Slepian–Wolf source network consisting of a link from each source to the receiver, our error exponents reduce to the corresponding results in [4] for linear Slepian–Wolf coding. The latter scenario may thus be considered a degenerate case of network coding.

We illustrate some possible applications with two examples of practical scenarios—distributed settings and networks with dynamically varying connections—in which random linear network coding shows particular promise of advantages over routing.

This paper is an initial exploration of random linear network coding, posing more questions that it answers. We do not cover aspects such as resource and energy allocation, but focus on optimally exploiting a given set of resources. Resource consumption can naturally be traded off against capacity and robustness, and across multiple communicating sessions; subsequent work on distributed resource optimization, e.g., [13], [21], has used random linear network coding as a component of the solution. There are also many issues surrounding the adaptation of protocols, which generally assume routing, to random coding approaches. We do not address these here, but rather seek to establish that the potential benefits of random linear network coding justify future consideration of protocol compatibility with or adaptation to network codes.

The basic random linear network coding approach involves no coordination among nodes. Implementations for various applications may not be completely protocol-free, but the roles and requirements for protocols may be substantially redefined in this new environment. For instance, if we allow for retrials to find successful codes, we in effect trade code length for some rudimentary coordination.

Portions of this work have appeared in [9], which introduced distributed random linear network coding; [8], which presented the Edmonds matrix formulation and a new bound on required field size for centralized network coding; [12], which generalized previous results to arbitrary networks and gave tighter bounds for acyclic networks; [11], on network coding for arbitrarily correlated sources; and [10], which considered random linear network coding for online network operation in dynamically varying environments.

A. Overview

A brief overview of related work is given in Section I-B. In Section II, we describe the network model and algebraic coding approach we use in our analyses, and introduce some notation and existing results. Section III gives some insights arising from consideration of bipartite matching and network flows. Success/error probability bounds for random linear network coding are given for independent and linearly correlated sources in Section IV and for arbitrarily correlated sources in Section V. We also give examples of practical scenarios in which randomized network coding can be advantageous compared to routing, in Section VI. We present our conclusions and some directions for further work in Section VII. Proofs and ancillary results are given in the Appendix .

B. Related Work

Ahlswede *et al.* [1] showed that with network coding, as symbol size approaches infinity, a source can multicast information at a rate approaching the smallest minimum cut between the source and any receiver. Li *et al.* [19] showed that linear coding with finite symbol size is sufficient for multicast. Koetter and Médard [17] presented an algebraic framework for network coding that extended previous results to arbitrary networks and robust networking, and proved the achievability with time-invariant solutions of the min-cut max-flow bound for networks with delay and cycles. Reference [17] also gave an algebraic characterization of the feasibility of a multicast problem and the validity of a network coding solution in terms of transfer matrices, for which we gave in [8] equivalent formulations obtained by considering bipartite matching and network flows. We used these formulations in obtaining a tighter upper bound on the required field size than the previous bound of [17], and in our analysis of distributed randomized network coding, introduced in [9]. Concurrent independent work by Sanders *et al.* [26] and Jaggi *et al.* [14] considered single-source multicast on acyclic delay-free graphs, showing a similar bound on field size by different means, and giving centralized deterministic and randomized polynomial-time algorithms for finding network coding solutions over a subgraph consisting of flow solutions to each receiver. Subsequent work by Fragouli and Soljanin [7] gave a tighter bound for the case of two sources and for some configurations with more than two sources. Lower bounds on coding field size were presented by Rasala Lehman and Lehman [18] and Feder *et al.* [6]. [6] also gave graph-specific upper bounds based on the number of "clashes" between flows from source to terminals.

Dougherty *et al.* [5] presented results on linear solutions for binary solvable multicast networks, and on nonfinite field alphabets. The need for vector coding solutions in some nonmulticast problems was considered by Rasala Lehman and Lehman [18], Médard *et al.* [22], and Riis [25]. Various practical protocols for and experimental demonstrations of random linear network coding [3] and nonrandomized network coding [29], [24] have also been presented.

II. Model and Preliminaries

A. Basic Model

Our basic network coding model is based on [1], [17]. A network is represented as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of network nodes and $\mathcal{E}$ is the set of links, such that information can be sent noiselessly from node i to j for all $(i, j) \in \mathcal{E}$. Each link $l \in \mathcal{E}$ is associated with a nonnegative real number c_l representing its transmission capacity in bits per unit time.

Nodes i and j are called the *origin* and *destination*, respectively, of link (i, j). The origin and destination of a link $l \in \mathcal{E}$ are denoted $o(l)$ and $d(l)$, respectively. We assume $o(l) \neq d(l)\ \forall\ l \in \mathcal{E}$. The information transmitted on a link $l \in \mathcal{E}$ is obtained as a coding function of information previously received at $o(l)$.

There are r discrete memoryless information source processes $X_1, X_2, \ldots, X_r$ which are random binary sequences. We denote the Slepian–Wolf region of the sources

$$\mathcal{R}_{SW} = \left\{ (R_1, R_2, \ldots, R_r) : \sum_{i \in \mathcal{S}} R_i > H(X_{\mathcal{S}} | X_{\mathcal{S}^c}) \quad \forall\ \mathcal{S} \subseteq \{1, 2, \ldots, r\} \right\}$$

where $X_{\mathcal{S}} = (X_{i_1}, X_{i_2}, \ldots, X_{i_{|\mathcal{S}|}}), i_k \in \mathcal{S}, k = 1, \ldots, |\mathcal{S}|$. Source process X_i is generated at node $a(i)$, and multicast to all nodes $j \in b(i)$, where $a : \{1, \ldots, r\} \rightarrow V$ and $b : \{1, \ldots, r\} \rightarrow 2^V$ are arbitrary mappings. In this paper, we consider the (multisource) multicast case where $b(i) = \{\beta_1, \ldots, \beta_d\}$ for all $i \in [1, r]$. The nodes $a(1), \ldots, a(r)$ are called *source nodes* and the nodes $\beta_1, \ldots, \beta_d$ are called *receiver nodes*, or receivers. For simplicity, we assume subsequently that $a(i) \neq \beta_j\ \forall\ i \in [1, r], j \in [1, d]$. The mapping a, the set $\{\beta_1, \ldots, \beta_d\}$ and the Slepian–Wolf region $\mathcal{R}_{SW}$ specify a set of multicast *connection requirements*. The connection requirements are satisfied if each receiver is able to reproduce, from its received information, the complete source information. A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a set of link capacities $\{c_l | l \in \mathcal{E}\}$, and a set of multicast connection requirements $\mathcal{C}$ specify a multicast connection problem.

We make a number of simplifying assumptions. Our analysis for the case of independent source processes assumes that each source process X_i has an entropy rate of one bit per unit time; sources of larger rate are modeled as multiple sources at the same node. For the case of linearly correlated sources, we assume that the sources can be modeled as given linear combinations of underlying independent source processes, each with an entropy rate of one bit per unit time, as described further in Section II-B. For the case of arbitrarily correlated sources, we consider sources with integer bit rates and arbitrary joint probability distributions.

For the case of independent or linearly correlated sources, each link $l \in \mathcal{E}$ is assumed to have a capacity c_l of one bit per unit time; links with larger capacities are modeled as parallel links. For the case of arbitrarily correlated sources, the link rates c_l are assumed to be integers.

Reference [1] shows that coding enables the multicast information rate from a single source to attain the minimum of the individual receivers' max-flow bounds,[1] and shows how to convert multicast problems with multiple independent sources to single-source problems. Reference [19] shows that linear coding is sufficient to achieve the same individual max-flow rates; in fact, it suffices to do network coding using only scalar algebraic operations in a finite field $\mathbb{F}_{2^u}$, for some sufficiently large u, on length-u vectors of bits that are viewed as elements of $\mathbb{F}_{2^u}$ [17]. The case of linearly correlated sources is similar.

For arbitrarily correlated sources, we consider operations in $\mathbb{F}_2$ on vectors of bits. This vector coding model can, for given vector lengths, be brought into the scalar algebraic framework of [17] by conceptually expanding each source into multiple sources and each link into multiple links, such that each new source and link corresponds to one bit of the corresponding information vectors. We describe this scalar framework in Section II-B, and use it in our analysis of arbitrarily correlated sources in Section V. Note, however, that the linear decoding strategies of [17] do not apply for the case of arbitrarily correlated sources.

We consider both the case of acyclic networks where link delays are not considered, as well as the case of general networks with cycles and link delays. The former case, which we call *delay-free*, includes networks whose links are assumed to have zero delay, as well as networks with link delays that are operated in a burst [19], pipelined [26], or batched [3] fashion, where information is buffered or delayed at intermediate nodes so as to be combined with other incoming information from the same batch. A cyclic graph with v nodes and rate r may also be converted to an expanded acyclic graph with κv nodes and rate at least $(\kappa - v)r$, communication on which can be emulated over κ time steps on the original cyclic graph [1]. For the latter case, we consider general networks without buffering, and make the simplifying assumption that each link has the same delay.

We use some additional definitions in this paper. Link l is an incident outgoing link of node v if $v = o(l)$, and an incident incoming link of v if $v = d(l)$. We call an incident incoming link of a receiver node a *terminal link*, and denote by $\mathcal{T}_\beta$ the set of terminal links of a receiver β. A *path* is a subgraph of the network consisting of a sequence of links $e_1, \ldots, e_k$ such that $d(e_i) = o(e_{i+1})$, $o(e_1) \neq d(e_k)$, and $d(e_i) \neq d(e_j)\ \forall\ i \neq j$, and is denoted $(e_1, \ldots, e_k)$. A *flow solution* for a receiver β is a set of links forming r link-disjoint paths each connecting a different source to β.

[1] That is, the maximum commodity flow from the source to individual receivers.

B. Algebraic Network Coding

In the scalar algebraic coding framework of [17], the source information processes, the receiver output processes, and the information processes transmitted on each link, are sequences of length-u blocks or vectors of bits, which are treated as elements of a finite field $\mathbb{F}_q$, $q = 2^u$. The information process Y_j transmitted on a link j is formed as a linear combination, in $\mathbb{F}_q$, of link j's *inputs*, i.e., source processes X_i for which $a(i) = o(j)$ and random processes Y_l for which $d(l) = o(j)$, if any. For the delay-free case, this is represented by the equation

$$Y_j = \sum_{\{i:a(i)=o(j)\}} a_{i,j}X_i + \sum_{\{l:d(l)=o(j)\}} f_{l,j}Y_l.$$

The ith output process $Z_{\beta,i}$ at receiver node β is a linear combination of the information processes on its terminal links, represented as

$$Z_{\beta,i} = \sum_{\{l:d(l)=\beta\}} b_{\beta,i,l}Y_l.$$

For multicast on a network with link delays, memory is needed at the receiver (or source) nodes, but memoryless operation suffices at all other nodes [17]. We consider unit delay links, modeling links with longer delay as links in series. The corresponding linear coding equations are

$$\begin{aligned} Y_j(t+1) &= \sum_{\{i:a(i)=o(j)\}} a_{i,j}X_i(t) \\ &\quad + \sum_{\{l:d(l)=o(j)\}} f_{l,j}Y_l(t) \\ Z_{\beta,i}(t+1) &= \sum_{u=0}^{\mu} b'_{\beta,i}(u)Z_{\beta,i}(t-u) \\ &\quad + \sum_{\{l:d(l)=\beta\}} \sum_{u=0}^{\mu} b''_{\beta,i,l}(u)Y_l(t-u) \end{aligned}$$

where $X_i(t)$, $Y_j(t)$, $Z_{\beta,i}(t)$, $b'_{\beta,i}(t)$, and $b''_{\beta,i,l}(t)$ are the values of the corresponding variables at time t, respectively, and μ represents the memory required. These equations, as with the random processes in the network, can be represented algebraically in terms of a delay variable D

$$\begin{aligned} Y_j(D) &= \sum_{\{i:a(i)=o(j)\}} Da_{i,j}X_i(D) \\ &\quad + \sum_{\{l:d(l)=o(j)\}} Df_{l,j}Y_l(D) \\ Z_{\beta,i}(D) &= \sum_{\{l:d(l)=\beta\}} b_{\beta,i,l}(D)Y_l(D) \end{aligned}$$

where

$$b_{\beta,i,l}(D) = \frac{\sum_{u=0}^{\mu} D^{u+1}b''_{\beta,i,l}(u)}{1 - \sum_{u=0}^{\mu} D^{u+1}b'_{\beta,i}(u)} \tag{1}$$

and

$$\begin{aligned} X_i(D) &= \sum_{t=0}^{\infty} X_i(t)D^t \\ Y_j(D) &= \sum_{t=0}^{\infty} Y_t(j)D^t, \qquad Y_j(0) = 0 \\ Z_{\beta,i}(D) &= \sum_{t=0}^{\infty} Z_{\beta,i}(t)D^t, \qquad Z_{\beta,i}(0) = 0. \end{aligned}$$

The coefficients $\{a_{i,j}, f_{l,j}, b_{\beta,i,l}\}$ can be collected into $r \times |\mathcal{E}|$ matrices

$$\boldsymbol{A} = \begin{cases} (a_{i,j}) & \text{in the acyclic delay-free case} \\ (Da_{i,j}) & \text{in the general case with delays} \end{cases}$$

and $\boldsymbol{B}_\beta = (b_{\beta,i,l})$, and the $|\mathcal{E}| \times |\mathcal{E}|$ matrix

$$\boldsymbol{F} = \begin{cases} (f_{l,j}) & \text{in the acyclic delay-free case} \\ (Df_{l,j}) & \text{in the general case with delays} \end{cases}$$

whose structure is constrained by the network. A pair $(\boldsymbol{A}, \boldsymbol{F})$ or tuple $(\boldsymbol{A}, \boldsymbol{F}, \boldsymbol{B}_{\beta_1}, \ldots, \boldsymbol{B}_{\beta_d})$ can be called a linear network code.

We also consider a class of linearly correlated sources modeled as given linear combinations of underlying independent processes, each with an entropy and bit rate of one bit per unit time. To simplify the notation in our subsequent development, we work with these underlying independent processes in a similar manner as for the case of independent sources: the jth column of the $\boldsymbol{A}$ matrix is a linear function $\sum_k \alpha_{k,j}\boldsymbol{x}_j^k$ of given column vectors $\boldsymbol{x}_j^k \in \mathbb{F}_2^r$, where $\boldsymbol{x}_j^k$ specifies the mapping from r underlying independent processes to the kth source process at $o(j)$.[2] A receiver that decodes these underlying independent processes is able to reconstruct the linearly correlated source processes.

For acyclic graphs, we assume an ancestral indexing of links in $\mathcal{E}$, i.e., if $d(l_1) = o(l_2)$ for any links l_1, l_2, then l_1 has a lower index than l_2. Such indexing always exists for acyclic networks. It then follows that matrix $\boldsymbol{F}$ is upper triangular with zeros on the diagonal.

Let $\boldsymbol{G} = (\boldsymbol{I} - \boldsymbol{F})^{-1}$.[3] The mapping from source processes $[X_1, \ldots, X_r]$ to output processes $[Z_{\beta,1}, \ldots, Z_{\beta,r}]$ at a receiver β is given by the transfer matrix $\boldsymbol{A}\boldsymbol{G}\boldsymbol{B}_\beta^T$ [17]. For a given multicast connection problem, if some network code $(\boldsymbol{A}, \boldsymbol{G}, \boldsymbol{B}_{\beta_1}, \ldots, \boldsymbol{B}_{\beta_d})$ in a field $\mathbb{F}_q$ (or $\mathbb{F}_q(D)$) satisfies the condition that $\boldsymbol{A}\boldsymbol{G}\boldsymbol{B}_{\beta_k}^T$ has full rank r for each receiver $\beta_k, k = 1, \ldots, d$, then $\tilde{\boldsymbol{B}}_{\beta_k} = (\boldsymbol{B}_{\beta_k}\boldsymbol{G}^T\boldsymbol{A}^T)^{-1}\boldsymbol{B}_{\beta_k}$ satisfies $\boldsymbol{A}\boldsymbol{G}\tilde{\boldsymbol{B}}_{\beta_k}^T = I$, and $(\boldsymbol{A}, \boldsymbol{G}, \tilde{\boldsymbol{B}}_{\beta_1}, \ldots, \tilde{\boldsymbol{B}}_{\beta_d})$ is a *solution* to the multicast connection problem in the same field. A multicast connection problem for which there exists a solution in some field $\mathbb{F}_q$ or $\mathbb{F}_q(D)$ is called *feasible*, and the corresponding connection requirements are said to be feasible for the network.

[2]We can also consider the case where $\boldsymbol{x}_j^k \in \mathbb{F}_{2^m}^r$ by restricting network coding to occur in $\mathbb{F}_q$, $q = 2^{mn}$.

[3]For the acyclic delay-free case, the sequence $(\boldsymbol{I}-\boldsymbol{F})^{-1} = \boldsymbol{I}+\boldsymbol{F}+\boldsymbol{F}^2+\cdots$ converges since $\boldsymbol{F}$ is nilpotent for an acyclic network. For the case with delays, $(\boldsymbol{I} - \boldsymbol{F})^{-1}$ exists since the determinant of $\boldsymbol{I} - \boldsymbol{F}$ is nonzero in its field of definition $\mathbb{F}_2(D, \ldots, f_{l,j}, \ldots)$, as seen by letting $D = 0$. [17]

In subsequent sections, where we consider choosing the value of $(\boldsymbol{A}, \boldsymbol{G})$ by distributed random coding, the following definitions are useful: if for a receiver β_k there exists some value of $\boldsymbol{B}_{\beta_k}$ such that $\boldsymbol{A}\boldsymbol{G}\boldsymbol{B}_{\beta_k}^T$ has full rank r, then $(\boldsymbol{A}, \boldsymbol{G})$ is a *valid* network code for β_k; a network code $(\boldsymbol{A}, \boldsymbol{G})$ is *valid* for a multicast connection problem if it is valid for all receivers.

The lth column of matrix $\boldsymbol{A}\boldsymbol{G}$ specifies the mapping from source processes to the random process on link l. We denote by $\boldsymbol{G}_{\mathcal{H}}$ the submatrix consisting of columns of $\boldsymbol{G}$ corresponding to a set of links $\mathcal{H}$.

For a receiver β to decode, it needs to know the mapping $\boldsymbol{A}\boldsymbol{G}_{\mathcal{T}_\beta}$ from the source processes to the random processes on its terminal links. The entries of $\boldsymbol{A}\boldsymbol{G}_{\mathcal{T}_\beta}$ are scalar elements of $\mathbb{F}_q$ in the acyclic delay-free case, and polynomials in delay variable D in the case with link delays. In the latter case, the number of terms of these polynomials and the memory required at the receivers depend on the number of links involved in cycles, which act like memory registers, in the network.

We use the notational convention that matrices are named with bold upper case letters and vectors are named with bold lower case letters.

III. Insights From Bipartite Matching and Network Flows

As described in the previous section, for a multicast connection problem with independent or linearly correlated sources, the transfer matrix condition of [17] for the problem to be feasible (or for a particular linear network code defined by matrices $(\boldsymbol{A}, \boldsymbol{G})$ to be valid for the connection problem) is that for each receiver β, the transfer matrix $\boldsymbol{A}\boldsymbol{G}\boldsymbol{B}_\beta^T$ has nonzero determinant. The following result shows the equivalence of this transfer matrix condition and the Edmonds matrix formulation for checking if a bipartite graph has a perfect matching (e.g., [23]). The problem of determining whether a bipartite graph has a perfect matching is a classical reduction of the problem of checking the feasibility of an $s-t$ flow [15].[4] This latter problem can be viewed as a degenerate case of network coding, restricted to the binary field and without any coding; it is interesting to find that the two formulations are equivalent for the more general case of linear network coding in higher order fields.

Lemma 1:

(a) For an acyclic delay-free network, the determinant of the transfer matrix $\boldsymbol{M}_1 = \boldsymbol{A}(\boldsymbol{I} - \boldsymbol{F})^{-1}\boldsymbol{B}_\beta^T$ for receiver β is equal to

$$|\boldsymbol{M}_1| = (-1)^{r(|\mathcal{E}|+1)}\,|\boldsymbol{M}_2|$$

where

$$\boldsymbol{M}_2 = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{I} - \boldsymbol{F} & \boldsymbol{B}_\beta^T \end{bmatrix}$$

is the corresponding Edmonds matrix.

(b) For an arbitrary network with unit delay links, the transfer matrix $\boldsymbol{A}(D)(\boldsymbol{I} - \boldsymbol{F}(D))^{-1}\boldsymbol{B}_\beta^T(D)$ for receiver β is nonsingular if and only if the corresponding Edmonds matrix

$$\begin{bmatrix} \boldsymbol{A}(D) & \boldsymbol{0} \\ \boldsymbol{I} - \boldsymbol{F}(D) & \boldsymbol{B}_\beta^T(D) \end{bmatrix}$$

is nonsingular.

Proof: See Appendix A. □

The usefulness of this result is in making apparent various characteristics of the transfer matrix determinant polynomial that are obscured in the original transfer matrix by the matrix products and inverse. For instance, the maximum exponent of a variable, the total degree of the polynomial, and its form for linearly correlated sources are easily deduced, leading to Theorems 1 and 2.

For the acyclic delay-free case, Lemma 2 below is another alternative formulation of the same transfer matrix condition which illuminates similar properties of the transfer matrix determinant as Lemma 1. Furthermore, by considering network coding as a superposition of flow solutions, Lemma 2 allows us to tighten, in Theorem 3, the bound of Theorem 2 for random network coding on given acyclic networks in terms of the number of links in a flow solution for an individual receiver.

Lemma 2: A multicast connection problem with r sources is feasible (or a particular network code $(\boldsymbol{A}, \boldsymbol{F})$ is valid for the problem) if and only if each receiver β has a set $\mathcal{H}_\beta$ of r terminal links for which

$$\sum_{\substack{\{\text{link-disjoint paths } \mathcal{E}_1=(l_{11},\ldots,l_{1n_1}),\ldots,\\ \mathcal{E}_r=(l_{r1},\ldots,l_{rn_r}):o(l_{i1})=a(i),l_{in_i}\in\mathcal{H}_\beta\}}} \left|\boldsymbol{A}_{\{l_{11},\ldots,l_{r1}\}}\right| \prod_{j=1}^{r} g(\mathcal{E}_j) \neq 0$$

where $\boldsymbol{A}_{\{l_{11},\ldots,l_{r1}\}}$ is the submatrix of $\boldsymbol{A}$ consisting of links $\{l_{11}, \ldots, l_{r1}\}$, and

$$g(\mathcal{E}) = \begin{cases} f_{e_1,e_2} f_{e_2,e_3}, \ldots, f_{e_{k-1},e_k}, & \text{if } k > 1 \\ 1, & \text{if } k = 1 \end{cases}$$

is the product of gains on the path $\mathcal{E} = (e_1, \ldots, e_k)$. The sum is over all flow solutions from the sources to links in $\mathcal{H}_\beta$, each such solution being a set of r link-disjoint paths each connecting a different source to a different link in $\mathcal{H}_\beta$.

Proof: See Appendix A. □

Lemma 1 leads to the following upper bound on required field size for a feasible multicast problem, which tightens the upper bound of $q > rd$ given in [17], where r is the number of processes being transmitted in the network.

Theorem 1: For a feasible multicast connection problem with independent or linearly correlated sources and d receivers, in both the acyclic delay-free case and the general case with delays,

[4]The problem of checking the feasibility of an $s - t$ flow of size r on graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ can be reduced to a bipartite matching problem by constructing the following bipartite graph: one set of the bipartite graph has r nodes $u_1, \ldots, u_r$, and a node $v_{l,1}$ corresponding to each link $l \in \mathcal{E}$; the other set of the bipartite graph has r nodes $w_1, \ldots, w_r$, and a node $v_{l,2}$ corresponding to each link $l \in \mathcal{E}$. The bipartite graph has links joining each node u_i to each node $v_{l,1}$ such that $o(l) = s$, a link joining node $v_{l,1}$ to the corresponding node $v_{l,2}$ for all $l \in \mathcal{E}$, links joining node $v_{l,2}$ to $v_{j,1}$ for each pair $(l, j) \in \mathcal{E} \times \mathcal{E}$ such that $d(l) = o(j)$, and links joining each node w_i to each node $v_{l,2}$ such that $d(l) = t$. The $s - t$ flow is feasible if and only if the bipartite graph has a perfect matching.

there exists a solution $(\boldsymbol{A}, \boldsymbol{F}, \boldsymbol{B}_{\beta_1}, \ldots, \boldsymbol{B}_{\beta_d})$ in finite field $\mathbb{F}_q$ if $q > d$.

Proof: See Appendix A. □

Work done in [14], [26] independently of and concurrently with the initial conference publication of this result showed, by different means, the sufficiency of $q \geq d$ for the acyclic delay-free case. Subsequent work in [7] gave a tighter bound of $q \geq \sqrt{2d - 7/4} + 1/2$ for the case of two sources and for some configurations with more than two sources that satisfy some regularity conditions.

IV. Random Linear Network Coding for Independent or Linearly Correlated Sources

In this section, we consider random linear network codes in which some or all of the network code coefficients $\{a_{i,j}\ (\alpha_{k,j}$ for linearly correlated sources), $f_{l,j}\}$ are chosen independently and uniformly over $\mathbb{F}_q$, where q is greater than the number of receivers d.

The next two results cover the case where some coefficients are fixed instead of being randomly chosen, as long as there exists a solution to the network connection problem with the same values for these fixed coefficients. For instance, if a node receives linearly dependent processes on two links l_1, l_2, it can fix $f_{l_1,j} = 0$ for all outgoing links j. Nodes that cannot determine the appropriate code coefficients from local information choose the coefficients independently and uniformly from $\mathbb{F}_q$.

Theorem 2: Consider a multicast connection problem on an arbitrary network with independent or linearly correlated sources, and a network code in which some or all network code coefficients $\{a_{i,j}, \alpha_{k,j}, f_{l,j}\}$ are chosen uniformly at random from a finite field $\mathbb{F}_q$ where $q > d$, and the remaining code coefficients, if any, are fixed. If there exists a solution to the network connection problem with the same values for the fixed code coefficients, then the probability that the random network code is valid for the problem is at least $(1 - d/q)^{\eta}$, where η is the number of links j with associated random coefficients $\{a_{i,j}, \alpha_{k,j}, f_{l,j}\}$.

Proof: See Appendix B. □

The code length u is the logarithm of the field size $q = 2^u$. It affects computational complexity and delay, since algebraic operations are performed on codewords of length u. Note that the bound, derived using Lemma 1, is tighter than the bound of $1 - d\eta/q$ obtained by direct application of the Schwartz–Zippel theorem (e.g., [23]) which only considers the total degree of the polynomial. The corresponding upper bound on the error probability is on the order of the inverse of the field size, so the error probability decreases exponentially with the number of codeword bits u.

The bound of Theorem 2 is very general, applying across all networks with the same number of receivers and the same number of links with associated random code coefficients, without considering specific network structure. However, it is intuitive that having more redundant capacity in the network, for instance, should increase the probability that a random linear code will be valid. Tighter bounds can be obtained by taking into account a more specific network structure. Three such bounds, for the acyclic delay-free case, are given below. We have not proven or disproven whether they extend to networks with cycles.

The first tightens the bound of Theorem 2 for the acyclic delay-free case, by using in its derivation Lemma 2 in place of Lemma 1. It is used in Section VI to derive a bound on the probability of obtaining a valid random network code on a grid network.

Theorem 3: Consider a multicast connection problem on an acyclic network with independent or linearly correlated sources, and a network code in which some or all network code coefficients $\{a_{i,j}, \alpha_{k,j}, f_{l,j}\}$ are chosen uniformly at random from a finite field $\mathbb{F}_q$ where $q > d$, and the remaining code coefficients, if any, are fixed. If there exists a solution to the network connection problem with the same values for the fixed code coefficients, then the probability that the random network code is valid for the problem is at least $(1 - d/q)^{\eta'}$, where η' is the maximum number of links with associated random coefficients in any set of links constituting a flow solution for any receiver.

Proof: See Appendix B. □

The next bound is useful in cases where analysis of connection feasibility is easier than direct analysis of random linear coding.

Theorem 4: Consider a multicast connection problem with independent or linearly correlated sources on an acyclic graph $\mathcal{G}$. The probability that a random linear network code in $\mathbb{F}_q$ is valid for the problem on $\mathcal{G}$ is greater than or equal to the probability that the same connection requirements are feasible on a modified graph formed by deleting each link of $\mathcal{G}$ with probability d/q.

Proof: See Appendix B. □

The above theorem is used in obtaining the following result showing how spare network capacity and/or more reliable links allow us to use a smaller field size to surpass a particular success probability.

Theorem 5: Consider a multicast connection problem on an acyclic network $\mathcal{G}$ with independent or linearly correlated sources of joint entropy rate r, and links which fail (are deleted from the network) with probability p. Let y be the minimum redundancy, i.e., the original connection requirements are feasible on a network obtained by deleting any y links in $\mathcal{G}$. The probability that a random linear network code in $\mathbb{F}_q$ is valid for a particular receiver is at least

$$\sum_{x=r}^{r+y} \binom{r+y}{x} \left(1 - p - \frac{1-p}{q}\right)^{Lx} \left(1 - \left(1 - p - \frac{1-p}{q}\right)^{L}\right)^{r+y-x}$$

where L is the longest source–receiver path in the network.

Proof: See Appendix B. □

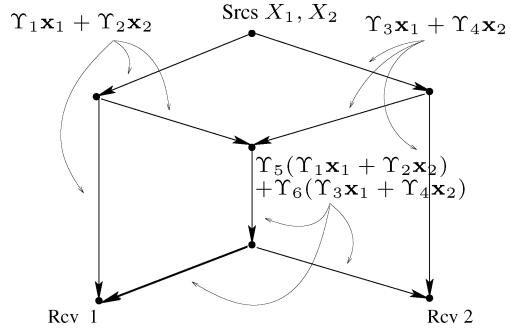

Fig. 2. An example illustrating vector linear coding. $\boldsymbol{x}_1 \in \mathbb{F}_2^{nr_1}$ and $\boldsymbol{x}_2 \in \mathbb{F}_2^{nr_2}$ are vectors of source bits being multicast to the receivers, and the matrices Υ_i are matrices of random bits. Suppose the capacity of each link is c. Matrices Υ_1 and Υ_3 are $nr_1 \times nc$, Υ_2 and Υ_4 are $nr_2 \times nc$, and Υ_5 and Υ_6 are $nc \times nc$. The label on each link represents the process being transmitted on the link.

V. Random Linear Network Coding for Arbitrarily Correlated Sources

So far we have been considering independent or linearly correlated sources. We next consider transmission of arbitrarily correlated sources, using random linear network coding, over networks where compression may be required.

Analogously to Slepian and Wolf [28], we consider the problem of distributed encoding and joint decoding of two sources whose output values in each unit time period are drawn independent and identically distributed (i.i.d.) from the same joint distribution Q. The difference is that in our problem, transmission occurs across an arbitrary network of intermediate nodes that can perform network coding. In the special case of a network consisting of one direct link from each source to a common receiver, this reduces to the original Slepian–Wolf problem.

We consider a vector linear network code that operates on blocks of bits. Linear coding is done in $\mathbb{F}_2$ over blocks consisting of nr_i bits from each source X_i, where r_i is the bit rate of source X_i. Each node transmits, on each of its incident outgoing links l, nc_l bits for each block, formed as random linear combinations of corresponding source bits originating at that node and bits transmitted on incident incoming links, if any, as illustrated in Fig. 2. An α-decoder (which may be a minimum entropy or maximum Q-probability decoder) [4] at a receiver maps a block of received bits to a block of decoded values that has minimum entropy or maximum Q-probability among all possible source values consistent with the received block.

We bound the probability of decoding error at a receiver, i.e., the probability that a block of source values differs from the decoded values. Specifically, we consider the case of two sources whose output values in each unit time period are drawn i.i.d. from the same joint distribution Q. Let m_1 and m_2 be the minimum cut capacities between the receiver and each of the sources, respectively, and let m_3 be the minimum cut capacity between the receiver and both sources. We denote by L the maximum source–receiver path length. Our approach follows that in [4], whose results we extend. As there, the type $P_{\boldsymbol{x}}$ of a vector $\boldsymbol{x} \in \mathbb{F}_2^{\tilde{n}}$ is the distribution on $\mathbb{F}_2$ defined by the relative frequencies of the elements of $\mathbb{F}_2$ in $\boldsymbol{x}$, and joint types $P_{\boldsymbol{xy}}$ are analogously defined.

Theorem 6: The error probability of the random linear network code is at most $\sum_{i=1}^{3} p_e^i$, where

$$p_e^1 \le \exp\left\{ -n \min_{X,Y} \left(D(P_{XY}\|Q) + \left| m_1(1 - \frac{1}{n}\log L) - H(X|Y) \right|^+ \right) + 2^{2r_1+r_2}\log(n+1) \right\}$$

$$p_e^2 \le \exp\left\{ -n \min_{X,Y} \left(D(P_{XY}\|Q) + \left| m_2(1 - \frac{1}{n}\log L) - H(Y|X) \right|^+ \right) + 2^{r_1+2r_2}\log(n+1) \right\}$$

$$p_e^3 \le \exp\left\{ -n \min_{X,Y} \left(D(P_{XY}\|Q) + \left| m_3(1 - \frac{1}{n}\log L) - H(XY) \right|^+ \right) + 2^{2r_1+2r_2}\log(n+1) \right\}$$

and X, Y are dummy random variables with joint distribution P_{XY}.

Proof: See Appendix B. □

The error exponents

$$e^1 = \min_{X,Y} \left(D(P_{XY}\|Q) + \left| m_1(1 - \frac{1}{n}\log L) - H(X|Y) \right|^+ \right)$$

$$e^2 = \min_{X,Y} \left(D(P_{XY}\|Q) + \left| m_2(1 - \frac{1}{n}\log L) - H(Y|X) \right|^+ \right)$$

$$e^3 = \min_{X,Y} \left(D(P_{XY}\|Q) + \left| m_3(1 - \frac{1}{n}\log L) - H(XY) \right|^+ \right)$$

for general networks reduce to those obtained in [4] for the Slepian–Wolf network where $L = 1$, $m_1 = R_1$, $m_2 = R_2$, $m_3 = R_1 + R_2$

$$e^1 = \min_{X,Y} \left(D(P_{XY}\|Q) + |R_1 - H(X|Y)|^+ \right)$$

$$e^2 = \min_{X,Y} \left(D(P_{XY}\|Q) + |R_2 - H(Y|X)|^+ \right)$$

$$e^3 = \min_{X,Y} \left(D(P_{XY}\|Q) + |R_1 + R_2 - H(XY)|^+ \right).$$

TABLE I
SUCCESS PROBABILITIES OF RANDOMIZED FLOODING SCHEME RF AND RANDOM LINEAR CODING SCHEME RC. THE TABLE GIVES BOUNDS AS WELL AS SOME ACTUAL PROBABILITY VALUES WHERE EXACT CALCULATIONS ARE TRACTABLE

Receiver position		(2,2)	(3,3)	(4,4)	(10,10)	(2,3)	(9,10)	(2,4)	(8,10)
RF	actual	0.75	0.672	0.637	-	0.562	-	0.359	-
	upper bound	0.75	0.688	0.672	0.667	0.625	0.667	0.563	0.667
RC	$\mathbb{F}_{2^4}$ lower bound	0.772	0.597	0.461	0.098	0.679	0.111	0.597	0.126
	$\mathbb{F}_{2^6}$ lower bound	0.939	0.881	0.827	0.567	0.910	0.585	0.882	0.604
	$\mathbb{F}_{2^8}$ lower bound	0.984	0.969	0.954	0.868	0.977	0.875	0.969	0.882

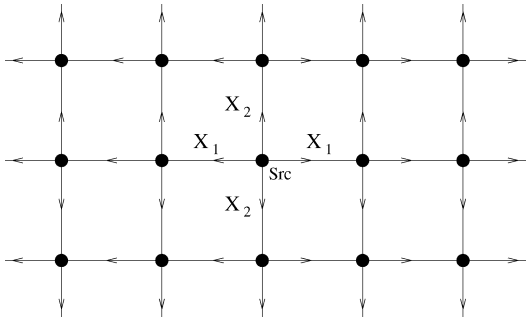

Fig. 3. Rectangular grid network with two processes X_1 and X_2 originating at a source node. The links are all directed outwards from the source node. The labels on the links show the source transmissions in the random flooding scheme RF, where one process is sent in both directions on one axis and the other process in both directions along the other axis.

VI. BENEFITS OF RANDOMIZED CODING OVER ROUTING

Network coding, as a superset of routing, has been shown to offer significant capacity gains for networks with special structure [26]. For many other networks, network coding does not give higher capacity than centralized optimal routing, but can offer other advantages when centralized optimal routing is difficult. In this section, we consider two types of network scenarios in which distributed random linear coding can be particularly useful.

A. Distributed Settings

In networks with large numbers of nodes and/or changing topologies, it may be expensive or infeasible to reliably maintain routing state at network nodes. Distributed randomized routing schemes have been proposed [2], [27] which address this kind of issue. However, not allowing different signals to be combined can impose intrinsic penalties in efficiency compared to using network coding.

Consider as a simple example the problem of sending two processes from a source node to receiver nodes in unknown locations on a rectangular grid network, shown in Fig. 3. For simplicity, we analyze the acyclic delay-free case, which may correspond to synchronized, burst, or pipelined operation where each transmission at a node v occurs upon reception of transmissions on all incident incoming links of v.

Suppose we wish to use a distributed transmission scheme that does not involve any coordination among nodes or routing state. The network aims to maximize the probability that any node will receive two distinct processes, by flooding in a way that preserves message diversity, for instance using the following random flooding scheme RF.

- The source node sends one process in both directions on one axis and the other process in both directions along the other axis, as illustrated in Fig. 3.
- A node receiving information on one link sends the same information on its three outgoing links (these are nodes along the grid axes passing through the source node).
- A node receiving information on two links sends one of the incoming processes on one of its two outgoing links with equal probability, and the other process on the remaining link.

For comparison, we consider the same rectangular grid problem with the following simple random coding scheme RC (ref Fig. 3).

- The source node sends one process in both directions on one axis and the other process in both directions along the other axis.
- A node receiving information on one link sends the same information on its three outgoing links.
- A node receiving information on two links sends a random linear combination of the source processes on each of its two outgoing links.[5]

Proposition 1: For the randomized flooding scheme RF, the probability that a receiver located at grid position (x, y) relative to the source receives both source processes is at most

$$\frac{1 + 2^{||x|-|y||+1}(4^{\min(|x|,|y|)-1} - 1)/3}{2^{|x|+|y|-2}}.$$

Proof: See Appendix C. □

Proposition 2: For the random coding scheme RC, the probability that a receiver located at grid position (x, y) relative to the source can decode both source processes is at least $(1 - 1/q)^{2(x+y-2)}$.

Proof: See Appendix C. □

Table I gives, for various values of x and y, the values of the success probability bounds as well as some actual probabilities for the random flooding scheme RF when x and y are small. Note that an increase in grid size from 3×3 to 10×10 requires only an increase of two in codeword length for the random coding scheme RC to obtain success probability lower bounds close to 0.9, which are substantially better than the upper bounds for RF.

[5]This simple scheme, unlike the randomized flooding scheme RF, leaves out the optimization that each node receiving two linearly independent processes should always send out two linearly independent processes.

TABLE II
A SAMPLE OF RESULTS ON GRAPHS GENERATED WITH THE FOLLOWING PARAMETERS: NUMBER OF NODES n, NUMBER OF SOURCES r, NUMBER OF RECEIVERS d, TRANSMISSION RANGE ρ, MAXIMUM IN-DEGREE AND OUT-DEGREE i. b_r AND b_c ARE THE RATE OF BLOCKED CONNECTIONS FOR ROUTING AND CODING, RESPECTIVELY, AND t_r AND t_c ARE THE CORRESPONDING THROUGHPUTS

Parameters						Results				
nodes n	srcs s	rcvrs d	deg i	range ρ	prob p_o	Network	b_r	t_r	b_c	t_c
8	6	1	4	0.5	0.6	1	1.54	1.46	1.55	1.46
						2	0.72	2.27	0.74	2.31
						3	0.26	2.78	0.23	2.74
9	6	2	3	0.5	0.7	1	2.14	0.84	2.17	0.83
						2	0.70	2.31	0.68	2.28
						3	0.90	2.05	0.71	2.26
10	4	2	4	0.5	0.6	1	0.61	1.43	0.50	1.45
						2	1.62	0.53	1.52	0.54
						3	0.14	1.96	0.00	2.05
10	6	2	4	0.5	0.5	1	1.31	1.63	0.71	2.28
						2	0.74	2.17	0.64	2.42
						3	1.51	1.54	1.49	1.61
10	9	3	3	0.5	0.7	1	1.05	2.37	1.14	2.42
						2	1.36	2.22	1.06	2.39
						3	2.67	0.87	2.56	0.89
12	6	2	4	0.5	0.6	1	1.44	1.67	0.71	2.31
						2	0.28	2.72	0.29	2.75
						3	0.75	2.28	0.73	2.31
12	8	2	3	0.5	0.7	1	2.39	1.73	2.34	1.74
						2	2.29	1.73	2.23	1.74
						3	1.57	2.48	1.52	2.51

B. Dynamically Varying Connections

Another scenario in which random linear network coding can be advantageous is for multisource multicast with dynamically varying connections. We compare distributed randomized coding to an approximate online Steiner tree routing approach from [16] in which, for each transmitter, a tree is selected in a centralized fashion. Since the complexity of setting up each connection is a significant consideration in the dynamic scenario we consider, we use one tree per connection; more complicated online routing approaches using multiple Steiner trees may be able to achieve a smaller performance gap compared to network coding, but this is not within the scope of our paper.

Since sophisticated routing algorithms are difficult to analyze, we used a simulation-based approach. We ran trials on randomly generated graphs with the following parameters: number of nodes n, number of sources r, number of receivers d, transmission range ρ, and maximum in-degree and out-degree i. For each trial, n nodes were scattered uniformly over a unit square. To create an acyclic graph we ordered the nodes by their x-coordinate and chose the direction of each link to be from the lower numbered to the higher numbered node. Any pair of nodes within Euclidian distance ρ of each other was connected by a link, up to the maximum in-degree and out-degree of the nodes involved. The receiver nodes were chosen as the d highest numbered nodes, and r source nodes were chosen randomly (with replacement) from among the lower numbered half of the nodes. The parameter values for the tests were chosen such that the resulting random graphs would in general be connected and able to support some of the desired connections, while being small enough for the simulations to run efficiently.

Each trial consisted of a number of time slots. In each time slot, a source was either on, i.e., transmitting source information, or off, i.e., not transmitting source information. For the approximate Steiner tree routing algorithm, each source that was on was associated with a Steiner tree, link-disjoint from the others, connecting it to all the receivers.

At the beginning of each time slot, any source that was on stayed on with probability $1 - p_o$ or else turned off, and any source that was off stayed off with probability $1 - p_o$ or else underwent, in turn, the following procedure.

- For the approximate Steiner tree routing algorithm, the algorithm was applied to search for a Steiner tree, link-disjoint with the Steiner trees of other sources that were currently on, connecting that source to all the receivers. If such a Steiner tree was found, the source turned on, using that Steiner tree to transmit its information to all receivers; if not, the source was blocked, i.e., stayed off.
- For network coding, up to three random linear network codes were chosen. If one of them was valid for transmitting information to all receivers from that source as well as other sources that were currently on, the source turned on; otherwise, the source was blocked.

We used as performance metrics the frequency of blocked requests and the average throughput, which were calculated for windows of 250 time slots until these measurements reached steady state, i.e., measurements in three consecutive windows being within a factor of 0.1 from each other, so as to avoid transient initial startup behavior. Some results for various randomly generated networks are given in Table II.

These simulations do not attempt to quantify precisely the differences in performance and overhead of random linear coding and online routing, but are useful as a preliminary indication. With regard to throughput and blocking probability, the simulations show that random linear network coding outperforms the Steiner tree heuristic on a non-negligible set of randomly constructed graphs, indicating that when connections vary dynami-

cally, coding can offer advantages that are not circumscribed to a few carefully chosen examples. With regard to overhead, the additional overhead of network coding comes from the linear coding operations at each node, the decoding operations at the receivers, and the coefficient vectors sent with each block or packet. Each of these types of overhead depends on the coding field size. Our theoretical bounds of previous sections guarantee the optimality of random linear coding for large enough field sizes, but they are tight only for worst case network connection problems. The simulations illustrate the kinds of field sizes needed in practice for networks with a moderate number of nodes. To this end, we use a small field size that allows random linear coding to generally match the performance of the Steiner heuristic, and to surpass it in networks whose topology makes Steiner tree routing difficult. The simulations show the applicability of short network code lengths of 4–5 bits for networks of 8–12 nodes.

VII. Conclusion

We have presented a distributed random linear network coding approach which asymptotically achieves capacity, as given by the max-flow min-cut bound of [1], in multisource multicast networks. We have given a general bound on the success probability of such codes for arbitrary networks, showing that error probability decreases exponentially with code length. Our analysis uses insights from network flows and bipartite matching, which also lead to a new bound on required field size for centralized network coding. We have also given tighter bounds for acyclic networks which take into account more specific network structure, and show how redundant network capacity and link reliability affect the probability of obtaining a valid random linear code.

Taking a source coding perspective, we have shown that distributed random linear network coding effectively compresses correlated sources within a network, providing error exponents that generalize corresponding results for linear Slepian–Wolf coding.

Finally, two examples of scenarios in which randomized network coding shows benefits over routing approaches have been presented. These examples suggest that the decentralized nature and robustness of random linear network coding can offer significant advantages in settings that hinder optimal centralized network control.

Further work includes extensions to nonuniform code distributions, possibly chosen adaptively or with some rudimentary coordination, to optimize different performance goals. Another question concerns selective placement of random linear coding nodes. The randomized and distributed nature of the approach also leads us naturally to consider applications in network security. It would also be interesting to consider protocol issues for different communication scenarios, and to compare specific coding and routing protocols over a range of performance metrics.

Appendix
Proofs and Ancillary Results

A. Edmonds Matrix and Flow Formulations

Proof of Lemma 1:

(a) Note that

$$\begin{bmatrix} \boldsymbol{I} & -\boldsymbol{A}(\boldsymbol{I}-\boldsymbol{F})^{-1} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{I}-\boldsymbol{F} & \boldsymbol{B}_\beta^T \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} & -\boldsymbol{A}(\boldsymbol{I}-\boldsymbol{F})^{-1}\boldsymbol{B}_\beta^T \\ \boldsymbol{I}-\boldsymbol{F} & \boldsymbol{B}_\beta^T \end{bmatrix}.$$

The first matrix

$$\begin{bmatrix} \boldsymbol{I} & -\boldsymbol{A}(\boldsymbol{I}-\boldsymbol{F})^{-1} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix}$$

has determinant 1. So

$$\det\left(\begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{I}-\boldsymbol{F} & \boldsymbol{B}_\beta^T \end{bmatrix}\right)$$

equals

$$\det\left(\begin{bmatrix} \boldsymbol{0} & -\boldsymbol{A}(\boldsymbol{I}-\boldsymbol{F})^{-1}\boldsymbol{B}_\beta^T \\ \boldsymbol{I}-\boldsymbol{F} & \boldsymbol{B}_\beta^T \end{bmatrix}\right)$$

which can be expanded as follows:

$$\begin{aligned} &\det\left(\begin{bmatrix} \boldsymbol{0} & -\boldsymbol{A}(\boldsymbol{I}-\boldsymbol{F})^{-1}\boldsymbol{B}_\beta^T \\ \boldsymbol{I}-\boldsymbol{F} & \boldsymbol{B}_\beta^T \end{bmatrix}\right) \\ &= (-1)^{r|\mathcal{E}|}\det\left(\begin{bmatrix} -\boldsymbol{A}(\boldsymbol{I}-\boldsymbol{F})^{-1}\boldsymbol{B}_\beta^T & \boldsymbol{0} \\ \boldsymbol{B}_\beta^T & \boldsymbol{I}-\boldsymbol{F} \end{bmatrix}\right) \\ &= (-1)^{r|\mathcal{E}|}\det(-\boldsymbol{A}(\boldsymbol{I}-\boldsymbol{F})^{-1}\boldsymbol{B}_\beta^T)\det(\boldsymbol{I}-\boldsymbol{F}) \\ &= (-1)^{r(|\mathcal{E}|+1)}\det(\boldsymbol{A}(\boldsymbol{I}-\boldsymbol{F})^{-1}\boldsymbol{B}_\beta^T)\det(\boldsymbol{I}-\boldsymbol{F}). \end{aligned}$$

The result follows from observing that $\det(\boldsymbol{I}-\boldsymbol{F})=1$ since $\boldsymbol{F}$ is upper-triangular with zeros along the main diagonal.

(b) As in part (a)

$$\det\left(\begin{bmatrix} \boldsymbol{A}(D) & \boldsymbol{0} \\ \boldsymbol{I}-\boldsymbol{F}(D) & \boldsymbol{B}_\beta^T(D) \end{bmatrix}\right) = (-1)^{r(|\mathcal{E}|+1)} \det(\boldsymbol{A}(D)(\boldsymbol{I}-\boldsymbol{F}(D))^{-1}\boldsymbol{B}_\beta^T(D))\det(\boldsymbol{I}-\boldsymbol{F}(D)).$$

Since $\det(\boldsymbol{I}-\boldsymbol{F}(D))$ is nonzero, the result follows. □

Proof of Lemma 2: Recall that we assume an ancestral numbering for the links of an acyclic graph. For $1 \le h' \le h \le |\mathcal{E}|$, let $S_{h',h}$ be the set of all sets of integers $\{e_1, e_2, \ldots, e_k\}$ such that $h' = e_1 < e_2 < \cdots < e_k = h$. Let $\mathcal{H} = \{h_1, \ldots, h_r\}$, where $1 \le h_1 < \cdots < h_r \le |\mathcal{E}|$.

Let $\boldsymbol{a}_h$ and $\boldsymbol{c}_h$ denote column h of $\boldsymbol{A}$ and $\boldsymbol{AG}$, respectively. It follows from the definitions of transfer matrices $\boldsymbol{A}$ and $\boldsymbol{G} = \boldsymbol{I}+\boldsymbol{F}+\boldsymbol{F}^2+\cdots$ that $\boldsymbol{c}_h$ can be computed recursively as follows:

$$\boldsymbol{c}_1 = \boldsymbol{a}_1 \tag{2}$$

$$\boldsymbol{c}_h = \sum_{i=1}^{h-1} \boldsymbol{c}_i f_{i,h} + \boldsymbol{a}_h, \qquad h = 2,3,\ldots,|\mathcal{E}|. \tag{3}$$

Expanding the determinant of $\boldsymbol{AG}_{\mathcal{H}}$ linearly in the $h_r{}^{th}$ column using (3), we obtain

$$|\boldsymbol{AG}_{\mathcal{H}}| = \left|\begin{array}{ccc} | & & | \\ \boldsymbol{c}_{h_1} & \ldots & \boldsymbol{c}_{h_r} \\ | & & | \end{array}\right| = \sum_{\substack{\{i:1\le i<h_r,\\ i\neq h_1,\ldots,h_{r-1}\}}} \left|\begin{array}{cccc} | & | & & | \\ \boldsymbol{c}_{h_1} & \ldots & \boldsymbol{c}_{h_{r-1}} & \boldsymbol{c}_i \\ | & | & & | \end{array}\right| f_{i,h_r} + \left|\begin{array}{cccc} | & | & & | \\ \boldsymbol{c}_{h_1} & \ldots & \boldsymbol{c}_{h_{r-1}} & \boldsymbol{a}_{h_r} \\ | & | & & | \end{array}\right|.$$

We proceed recursively, expanding each determinant linearly in its column $\boldsymbol{c}_h$ whose index h is highest, using (3) for $h > 1$ and (2) for $h = 1$. At each expansion stage, the expression for $\boldsymbol{AG}_{\mathcal{H}}$ is a linear combination of matrix determinants. Each nonzero determinant corresponds to a matrix composed of columns $\{\boldsymbol{a}_{k_1},\ldots,\boldsymbol{a}_{k_s},\boldsymbol{c}_{k_{s+1}},\ldots,\boldsymbol{c}_{k_r}\}$ such that $k_i \neq k_j \forall i \neq j$, and $\min(k_1,\ldots,k_s) > \max(k_{s+1},\ldots,k_r)$. Its coefficient in the linear combination is a product of terms $f_{i,h}$ such that $h > k_{s+1},\ldots,k_r$, and is of the form $\prod_{j=1}^{r} g(\mathcal{E}_j)$ where $\mathcal{E}_j \in S_{k_{j'},h_j}$ for some $j' \in [1,r]$ and $\mathcal{E}_i \cap \mathcal{E}_j = \emptyset \ \forall\ i \neq j$. By induction we have that these properties hold for all nonzero determinant terms in the course of the expansion. The expansion terminates when the expression is a linear combination of determinants of the form $|\boldsymbol{a}_{l_1}\ldots\boldsymbol{a}_{l_r}|$, at which point we have

$$|\boldsymbol{AG}_{\mathcal{H}}| = \sum_{\substack{\{(h'_1,\ldots,h'_r):\\ 1\le h'_j\le h_j,\\ h'_i\neq h'_j\ \forall\, i\neq j\}}} \left|\begin{array}{ccc} | & & | \\ \boldsymbol{a}_{h'_1} & \ldots & \boldsymbol{a}_{h'_r} \\ | & & | \end{array}\right| \sum_{\substack{\{(\mathcal{E}_1,\ldots,\mathcal{E}_r):\\ \mathcal{E}_j\in S_{h'_j,h_j},\\ \mathcal{E}_i\cap\mathcal{E}_j=\emptyset\\ \forall\, i\neq j\}}} \prod_{j=1}^{r} g(\mathcal{E}_j).$$

The result follows by noting that each set $\mathcal{E} = \{e_1,e_2,\ldots,e_k\}$ such that $g(\mathcal{E}) \neq 0$ corresponds to a network path consisting of links $e_1,\ldots,e_k$; that the condition $\mathcal{E}_j \cap \mathcal{E}_k = \emptyset$ for all $j \neq k,\ 1 \le j,k \le r$ implies that the corresponding paths $\mathcal{E}_1,\ldots,\mathcal{E}_r$ are disjoint; and that $\left|\boldsymbol{a}_{h'_1}\ldots\boldsymbol{a}_{h'_r}\right|$ is nonzero only when links $h_{j'}$ transmit r linearly independent combinations of source processes. □

Proof of Theorem 1: By Lemma 1, the transfer matrix determinant $|AGB_\beta^T|$ for any receiver β is nonzero if and only if the determinant of the corresponding Edmonds matrix is nonzero. Thus, we consider the determinant P_β of the latter matrix. Since each variable $a_{x,j}$ ($\alpha_{k,j}$ in the case of linearly correlated sources), $f_{i,j}$ or $b_{\beta,i,l}$ appears in exactly one column of the Edmonds matrix, the largest exponent of each of these variables in P_β is 1, and the largest exponent of each variable in the product $P = \prod_\beta P_\beta$ of d receivers' determinants is at most d.

For the acyclic delay-free case, we use an induction argument similar to that in [17] to show that there exists a solution in $\mathbb{F}_q, q > d$, such that P is nonzero. Consider one of the variables $a_{x,j}$, $\alpha_{k,j}$, $f_{i,j}$, or $b_{\beta,i,l}$, denoting it by ξ_1, and consider P as a polynomial in the other variables with coefficients that are polynomials in ξ_1. Since these coefficients have maximum degree d, they are not divisible by $\xi_1^q - \xi_1$. Thus, ξ_1 can take some value in $\mathbb{F}_q$ such that at least one of the coefficients is nonzero. Repeating this procedure for each of the other variables gives the desired result.

Going from the acyclic delay-free case to the general case with delays, variables $a_{x,j}$, $\alpha_{k,j}$, $f_{i,j}$ are replaced by $Da_{x,j}$, $D\alpha_{k,j}$, $Df_{i,j}$ in the Edmonds matrix, and variables $b_{\beta,i,l}$ become rational functions in $D, b'_{\beta,i}(u), b''_{\beta,i,l}(u)$ given by (1) in Section II-B. Each variable $b''_{\beta,i,l}(u)$ appears in only one entry of the Edmonds matrix, and each variable $b'_{\beta,i}(u)$ appears in only one column of the Edmonds matrix in a linear expression that forms the denominator of each nonzero entry of the column. Thus, P_β can be expressed as a ratio of polynomials whose numerator is linear in each variable $a_{x,j}$, $\alpha_{k,j}$, $f_{i,j}$, $b'_{\beta,i}(u)$ or $b''_{\beta,i,l}(u)$. Proceeding similarly as for the acyclic delay-free case yields the result. □

B. Analysis of Random Linear Network Coding

Lemma 3: Consider a random network code $(\boldsymbol{A},\boldsymbol{F})$ in which η links j have associated code coefficients $a_{i,j}$, ($\alpha_{k,j}$ for the case of linearly correlated sources) and/or $f_{l,j}$ that are randomly chosen variables. The determinant polynomial of the corresponding Edmonds matrix

$$\begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{I}-\boldsymbol{F} & \boldsymbol{B}_\beta^T \end{bmatrix}$$

has maximum degree η in the random variables $\{a_{i,j},\alpha_{k,j},f_{l,j}\}$, and is linear in each of these variables.

Proof: Each variable $\{a_{i,j},\alpha_{k,j},f_{l,j}\}$ appears in only one column of the Edmonds matrix. Only the η columns corresponding to links transmitting random combinations of input processes contain variable terms $\{a_{i,j},\alpha_{k,j},f_{l,j}\}$.

The determinant can be written as the sum of products of $r+|\mathcal{E}|$ entries, one from each row and column. Each such product is linear in each variable term $\{a_{i,j},\alpha_{k,j},f_{l,j}\}$, and has degree at most η in these variables. □

Lemma 4: Let P be a nonzero polynomial in $\mathbb{F}[\xi_1,\xi_2,\ldots]$ of degree less than or equal to $d\eta$, in which the largest exponent of any variable ξ_i is at most d. Values for $\xi_1,\xi_2,\ldots$ are chosen independently and uniformly at random from $\mathbb{F}_q \subseteq \mathbb{F}$. The probability that P equals zero is at most $1-(1-d/q)^\eta$ for $d < q$.

Proof: For any variable ξ_1 in P, let d_1 be the largest exponent of ξ_1 in P. Express P in the form $P = \xi_1^{d_1}P_1 + R_1$, where P_1 is a polynomial of degree at most $d\eta - d_1$ that does not contain variable ξ_1, and R_1 is a polynomial in which the largest exponent of ξ_1 is less than d_1. By the Principle of Deferred Decisions (e.g., [23]), the probability $\Pr[P=0]$ is unaffected if we set the value of ξ_1 last after all the other coefficients have been set. If, for some choice of the other coefficients, $P_1 \neq 0$, then P becomes a polynomial in $\mathbb{F}[\xi_1]$ of degree d_1.

By the Schwartz–Zippel theorem (e.g., [23]), this probability $\Pr[P=0|P_1 \neq 0]$ is upper-bounded by d_1/q. So

$$\Pr[P=0] \leq \Pr[P_1 \neq 0]\frac{d_1}{q} + \Pr[P_1 = 0] = \Pr[P_1 = 0]\left(1 - \frac{d_1}{q}\right) + \frac{d_1}{q}. \quad (4)$$

Next we consider $\Pr[P_1 = 0]$, choosing any variable ξ_2 in P_1 and letting d_2 be the largest exponent of ξ_2 in P_1. We express P_1 in the form $P_1 = \xi_2^{d_2} P_2 + R_2$, where P_2 is a polynomial of degree at most $d\eta - d_1 - d_2$ that does not contain variables ξ_1 or ξ_2, and R_2 is a polynomial in which the largest exponent of ξ_2 is less than d_2. Proceeding similarly, we assign variables ξ_i and define d_i and P_i for $i = 3, 4, \ldots$ until we reach $i = k$ where P_k is a constant and $\Pr[P_k = 0] = 0$. Note that $1 \leq d_i \leq d < q \forall\, i$ and $\sum_{i=1}^{k} d_i \leq d\eta$, so $k \leq d\eta$. Applying Schwartz–Zippel as before, we have for $k' = 1, 2, \ldots, k$

$$\Pr[P_{k'} = 0] \leq \Pr[P_{k'+1} = 0]\left(1 - \frac{d_{k'+1}}{q}\right) + \frac{d_{k'+1}}{q}. \quad (5)$$

Combining all the inequalities recursively, we can show by induction that

$$\Pr[P=0] \leq \frac{\sum_{i=1}^{k} d_i}{q} - \frac{\sum_{i \neq j} d_i d_j}{q^2} + \cdots + (-1)^{k-1}\frac{\prod_{i=1}^{k} d_i}{q^k}.$$

Now consider the integer optimization problem

$$\begin{aligned} \text{Maximize} \quad & f = \frac{\sum_{i=1}^{d\eta} d_i}{q} - \frac{\sum_{i \neq j} d_i d_j}{q^2} + \cdots + (-1)^{d\eta-1}\frac{\prod_{i=1}^{d\eta} d_i}{q^{d\eta}} \\ \text{subject to} \quad & 0 \leq d_i \leq d < q \,\forall\, i \in [1, d\eta], \\ & \sum_{i=1}^{d\eta} d_i \leq d\eta, \quad \text{and} \quad d_i \ \text{integer} \end{aligned} \quad (6)$$

whose maximum is an upper bound on $\Pr[P = 0]$.

We first consider the problem obtained by relaxing the integer condition on the variables d_i. Let $\boldsymbol{d}^* = \{d_1^*, \ldots, d_{d\eta}^*\}$ be an optimal solution.

For any set S_h of h distinct integers from $[1, d\eta]$, let

$$f_{S_h} = 1 - \frac{\sum_{i \in S_h} d_i}{q} + \frac{\sum_{i,j \in S_h, i \neq j} d_i d_j}{q^2} - \cdots + (-1)^h \frac{\prod_{i \in S_h} d_i}{q^h}.$$

We can show by induction on h that $0 < f_{S_h} < 1$ for any set S_h of h distinct integers in $[1, d\eta]$. If $\sum_{i=1}^{d\eta} d_i^* < d\eta$, then there is some $d_i^* < d$, and there exists a feasible solution $\boldsymbol{d} = \{d_1, \ldots, d_{d\eta}\}$ such that $d_i = d_i^* + \epsilon$, $\epsilon > 0$, and $d_h = d_h^*$ for $h \neq i$, which satisfies

$$f(\boldsymbol{d}) - f(\boldsymbol{d}^*) = \frac{\epsilon}{q}\left(1 - \frac{\sum_{h \neq i} d_h^*}{q} + \cdots + (-1)^{d\eta-1}\frac{\prod_{h \neq i} d_h^*}{q^{d\eta-1}}\right).$$

This is positive, contradicting the optimality of $\boldsymbol{d}^*$, so $\sum_{i=1}^{d\eta} d_i^* = d\eta$.

Next suppose $0 < d_i^* < d$ for some d_i^*. Then there exists some d_j^* such that $0 < d_j^* < d$, since if $d_j^* = 0$ or d for all other j, then $\sum_{i=1}^{d\eta} d_i^* \neq d\eta$. Assume without loss of generality that $0 < d_i^* \leq d_j^* < d$. Then there exists a feasible vector $\boldsymbol{d} = \{d_1, \ldots, d_{d\eta}\}$ such that $d_i = d_i^* - \epsilon$, $d_j = d_j^* + \epsilon$, $\epsilon > 0$, and $d_h = d_h^* \,\forall\, h \neq i, j$, which satisfies

$$f(\boldsymbol{d}) - f(\boldsymbol{d}^*) = -\left(\frac{(d_i^* - d_j^*)\epsilon - \epsilon^2}{q^2}\right)\left(1 - \frac{\sum_{h \neq i,j} d_h^*}{q} - \cdots + (-1)^{d\eta-2}\frac{\prod_{h \neq i,j} d_h^*}{q^{d\eta-2}}\right).$$

This is again positive, contradicting the optimality of $\boldsymbol{d}^*$.

Thus, $\sum_{i=1}^{d\eta} d_i^* = d\eta$, and $d_i^* = 0$ or d. So exactly η of the variables d_i^* are equal to d. Since the optimal solution is an integer solution, it is also optimal for the integer program (6). The corresponding optimal

$$f = \eta\frac{d}{q} - \binom{\eta}{2}\frac{d^2}{q^2} + \cdots + (-1)^{\eta-1}\frac{d^\eta}{q^\eta} = 1 - \left(1 - \frac{d}{q}\right)^\eta. \ \square$$

Proof of Theorem 2: There are η links j with associated code coefficients

$$\{a_{i,j}\ (\alpha_{k,j} \text{ in the case of linearly correlated sources}), f_{l,j}\}$$

that are chosen independently and uniformly at random over $\mathbb{F}_q$. To check if the resulting network code $(\boldsymbol{A}, \boldsymbol{F})$ is valid for a receiver β, it suffices to check that the determinant of the corresponding Edmonds matrix is nonzero (Lemma 1). This determinant, which we denote by P_β, is a polynomial linear in each variable $\{a_{x,j}, \alpha_{k,j}, f_{i,j}\}$, with total degree at most η in these variables (Lemma 3). The product $\prod_\beta P_\beta$ for d receivers is, accordingly, a polynomial in $\{a_{x,j}, \alpha_{k,j}, f_{i,j}\}$ of total degree at most $d\eta$, and in which the largest exponent of each of these variables is at most d. Applying Lemma 4, $\prod_\beta P_\beta$ is nonzero with probability at least $1 - \left(1 - \frac{d}{q}\right)^\eta$.

The bound is attained with equality for a network with independent sources that consists only of link-disjoint paths, one for each source–receiver pair. In this case, there is a one-to-one correspondence between links and variables $\{a_{i,j}, f_{l,j}\}$. Each of these variables must be nonzero in order for the code to be valid for all receivers. $\square$

Proof of Theorem 3: By Lemma 2, a given network code is valid if, for each receiver β, a linear combination of product terms of the form $a_{x_1,l_1} \ldots a_{x_r,l_r} f_{i_1,l_{r+1}} \ldots f_{i_{\eta_\beta},l_{r+\eta_\beta}}$, where $\{l_1, \ldots, l_{r+\eta_\beta}\}$ form a flow solution to β, is nonzero. The product of the corresponding expressions for d receivers has degree less than or equal to $d\eta'$, where $\eta' = \max_\beta \eta_\beta$, and the largest exponent of any variable is at most d. Applying Lemma 4 yields the result. $\square$

The same proof holds for linearly correlated sources, by considering variables $\alpha_{k,j}$ in place of variables $a_{i,j}$.

Proof of Theorem 4: Recall that links in an acyclic graph are numbered ancestrally. For $0 \leq j \leq |\mathcal{E}|$, suppose random network coding is done over links $i \leq j$, if any, and any link $i \geq j+1$ is deleted with probability d/q. Let $\mathcal{G}_j$ be the graph formed by removing deleted links from $\mathcal{G}$. Let $E_{\beta,j}$ be the event that there exist code coefficients for undeleted links $i \geq j+1$

such that the resulting network code is valid for receiver β over $\mathcal{G}_j$. Denote by $p_{\beta,j}$ be the probability of $E_{\beta,j}$.

To prove the theorem, we need to show that

$$\Pr\left(\bigcup_{\beta} E_{\beta,0}\right) \leq \Pr\left(\bigcup_{\beta} E_{\beta,|\mathcal{E}|}\right)$$

which follows from showing that

$$\Pr\left(\bigcup_{\beta} E_{\beta,j}\right) \leq \Pr\left(\bigcup_{\beta} E_{\beta,j+1}\right)$$

for $j = 0, \ldots, |\mathcal{E}|$.

For any $0 \leq j \leq |\mathcal{E}| - 1$, consider any given subset of links $j+2, \ldots, |\mathcal{E}|$ that are deleted, forming graph $\mathcal{G}_{j+1}$, and any given code coefficients for links $1, \ldots, j$. We compare the conditional probability of event $\bigcup_\beta E_{\beta,j}$ (when link $j+1$ is deleted with probability d/q) and event $\bigcup_\beta E_{\beta,j+1}$ (when random code coefficients are chosen for $j+1$). There are three cases.

Case 1: event $\bigcup_\beta E_{\beta,j}$ occurs if link $j+1$ is deleted. Then event $\bigcup_\beta E_{\beta,j}$ occurs regardless of whether link $j+1$ is deleted, and event $\bigcup_\beta E_{\beta,j+1}$ occurs regardless of the values of the random code coefficients for link $j+1$, since a valid code exists over $\mathcal{G}_{j+1}$ with the given code coefficients for links $1, \ldots, j$ and zero coefficients $f_{j+1,k}$ for any link $k > j+1$ such that $o(k) = d(j+1)$.

Case 2: event $\bigcup_\beta E_{\beta,j}$ does not occur if link $j+1$ is deleted, but occurs if link $j+1$ is not deleted. Then there exists at least one choice of code coefficients for link $j+1$ and any undeleted links $i \geq j+1$ such that the resulting network code is valid for all receivers over $\mathcal{G}_{j+1}$. Each receiver β has a set of r terminal links whose coefficient vectors form a full rank set. Consider the determinant of the associated matrix, expressed as a polynomial $P_{\beta,j+1}$ with the code coefficients $\{a_{x,j+1}, f_{i,j+1}\}$ for link $j+1$ as random variables. From Lemma 1, $P_{\beta,j+1}$ is linear in the variables $\{a_{x,j+1}, f_{i,j+1}\}$. The product $\prod_\beta P_{\beta,j+1}$ for d receivers is a polynomial of degree at most d in the variables $\{a_{x,j+1}, f_{i,j+1}\}$. If this product is nonzero, the corresponding code is valid. By the Schwartz–Zippel theorem, this product takes a nonzero value with probability $1 - d/q$ when the variables $\{a_{x,j+1}, f_{i,j+1}\}$ are chosen uniformly at random from a finite field of size q. Thus, the conditional probability of event $\bigcup_\beta E_{\beta,j+1}$ is at least $1 - d/q$.

Case 3: event $\bigcup_\beta E_{\beta,j}$ does not occur regardless of whether link $j+1$ is deleted. Then event $\bigcup_\beta E_{\beta,j+1}$ does not occur regardless of the values of the random code coefficients for link $j+1$, since no valid code exists over $\mathcal{G}_{j+1}$ with the given code coefficients for links $1, \ldots, j$.

In all three cases, the conditional probability of event $\bigcup_\beta E_{\beta,j}$ is less than or equal to the conditional probability of event $\bigcup_\beta E_{\beta,j+1}$. □

The same proof holds for linearly correlated sources, by considering variables $\alpha_{k,j}$ in place of variables $a_{i,j}$.

Proof of Theorem 5 : Note that for any multicast connection problem, the probability that a random network code is valid for a particular receiver β is equal to the probability that a random network code is valid for a modified connection problem in which β is the only receiver. Consider any single-receiver connection problem C on a graph $\mathcal{G}_C$. Let p_C be the probability that the connection requirements of C are feasible on the graph obtained by deleting links of $\mathcal{G}_C$ with probability $1-(1-p)(1-1/q)$. Let $\mathcal{G}'_C$ be the graph obtained by deleting links of $\mathcal{G}_C$ with probability p, and $\mathcal{G}''_C$ the graph obtained by deleting links of $\mathcal{G}'_C$ with probability $1/q$. By Theorem 4, the probability that random network coding gives a code valid for the connection requirements of C on $\mathcal{G}'_C$ can be lower-bounded by the probability that the connection requirements are feasible on $\mathcal{G}''_C$, which is equal to p_C.

Consider a single-receiver connection problem C_1 with r source processes originating at a common source node, on a graph $\mathcal{G}_{C_1}$ consisting of $r+y$ link-disjoint source–receiver paths of length L. Let C_2 be any other single-receiver connection problem with r source processes on a y-redundant graph $\mathcal{G}_{C_2}$ with source–receiver paths of length at most L. Suppose links of each graph are deleted with probability $1-(1-p)(1-1/q)$. We show by induction on y that $p_{C_2} \geq p_{C_1}\ \forall\, y, r, L$.

For $i = 1, 2$, we consider a set $\mathcal{P}_i$ of links in graph $\mathcal{G}_{C_i}$ forming r link-disjoint source–receiver paths sufficient to transmit all processes to the receiver. We distinguish two cases.

Case 1: None of the links in $\mathcal{P}_i$ are deleted. In this case, the connections are feasible.

Case 2: There exists some link $j_i \in \mathcal{P}_i$ that is deleted.

Then we have

$$\begin{aligned}\Pr(\text{success}) &= \Pr(\text{case 1}) + \Pr(\text{case 2})\Pr(\text{success}|\text{case 2}) \\ &= 1 - \Pr(\text{case 2})\left(1 - \Pr(\text{success}|\text{case 2})\right).\end{aligned}$$

Since $\mathcal{P}_1$ has at least as many links as $\mathcal{P}_2$

$$\Pr(\text{case 2}, i=1) \geq \Pr(\text{case 2}, i=2).$$

Thus, if we can show that

$$\Pr(\text{success}|\text{case 2}, i=1) \leq \Pr(\text{success}|\text{case 2}, i=2)$$

the induction hypothesis

$$\Pr(\text{success}|i=1) \leq \Pr(\text{success}|i=2)$$

follows.

For $y = 0$, the hypothesis is true since $\Pr(\text{success}|\text{case 2}, i=1) = 0$. For $y > 0$, in case 2 we can remove link j_i leaving a $(y-1)$-redundant graph $\tilde{\mathcal{G}}_{C_i}$. By the induction hypothesis, the probability of success for $\tilde{\mathcal{G}}_{C_1}$ is less than or equal to that for $\tilde{\mathcal{G}}_{C_2}$.

Thus, $p_{C_2} \geq p_{C_1}$, which is the probability that all links on at least r of $r+y$ length-L paths are not deleted. The result follows from observing that the probability that the links on a path are not deleted is $\left((1-p)(1-\frac{1}{q})\right)^L$. □

Proof of Theorem 6: We consider transmission, by random linear network coding, of one block of source bits, represented by vector $[\boldsymbol{x}_1 \quad \boldsymbol{x}_2] \in \mathbb{F}_2^{n(r_1+r_2)}$. The transfer matrix $\boldsymbol{AG}_{\mathcal{T}}$ specifies the mapping from the vector of source bits $[\boldsymbol{x}_1 \quad \boldsymbol{x}_2]$ to the vector $\boldsymbol{z}$ of processes on the set $\mathcal{T}$ of terminal links incident to the receiver.

The first part of the proof parallels the analysis in [4]. The α-decoder maps a vector $\boldsymbol{z}$ of received processes onto a vector $[\tilde{\boldsymbol{x}_1} \quad \tilde{\boldsymbol{x}_2}] \in \mathbb{F}_2^{n(r_1+r_2)}$ minimizing $\alpha(P_{\boldsymbol{x}_1\boldsymbol{x}_2})$ subject

to $[\boldsymbol{x}_1 \quad \boldsymbol{x}_2]\boldsymbol{AG}_{\mathcal{T}} = \boldsymbol{z}$. For a minimum entropy decoder, $\alpha(P_{\boldsymbol{x}_1\boldsymbol{x}_2}) \equiv H(P_{\boldsymbol{x}_1\boldsymbol{x}_2})$, while for a maximum Q-probability decoder, $\alpha(P_{\boldsymbol{x}_1\boldsymbol{x}_2}) \equiv -\log Q^n(\boldsymbol{x}_1\boldsymbol{x}_2)$. We consider three types of errors: in the first type, the decoder has the correct value for $\boldsymbol{x}_2$ but outputs the wrong value for $\boldsymbol{x}_1$; in the second, the decoder has the correct value for $\boldsymbol{x}_1$ but outputs the wrong value for $\boldsymbol{x}_2$; in the third, the decoder outputs wrong values for both $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. The error probability is upper-bounded by the sum of the probabilities of the three types of errors, $\sum_{i=1}^{3} p_e^i$.

As in [4], (joint) types of sequences are considered as (joint) distributions P_X ($P_{X,Y}$, etc.) of dummy variables X, Y, etc. The set of different types of sequences in $\mathbb{F}_2^k$ is denoted by $\mathcal{P}(\mathbb{F}_2^k)$. Defining the sets of types

$$\mathcal{P}_n^i = \begin{cases} \{P_{X\tilde{X}Y\tilde{Y}} \in \mathcal{P}(\mathbb{F}_2^{nr_1} \times \mathbb{F}_2^{nr_1} \times \mathbb{F}_2^{nr_2} \times \mathbb{F}_2^{nr_2}) \mid \\ \quad \tilde{X} \neq X, \tilde{Y} = Y\}, & i = 1 \\ \{P_{X\tilde{X}Y\tilde{Y}} \in \mathcal{P}(\mathbb{F}_2^{nr_1} \times \mathbb{F}_2^{nr_1} \times \mathbb{F}_2^{nr_2} \times \mathbb{F}_2^{nr_2}) \mid \\ \quad \tilde{X} = X, \tilde{Y} \neq Y\}, & i = 2 \\ \{P_{X\tilde{X}Y\tilde{Y}} \in \mathcal{P}(\mathbb{F}_2^{nr_1} \times \mathbb{F}_2^{nr_1} \times \mathbb{F}_2^{nr_2} \times \mathbb{F}_2^{nr_2}) \mid \\ \quad \tilde{X} \neq X, \tilde{Y} \neq Y\}, & i = 3 \end{cases}$$

and the sets of sequences

$$\begin{aligned} \mathcal{T}_{XY} &= \{[\boldsymbol{x}_1 \quad \boldsymbol{x}_2] \in \mathbb{F}_2^{n(r_1+r_2)} \mid \\ &\qquad P_{\boldsymbol{x}_1\boldsymbol{x}_2} = P_{XY}\} \\ \mathcal{T}_{\tilde{X}\tilde{Y}|XY}(\boldsymbol{x}_1\boldsymbol{x}_2) &= \{[\tilde{\boldsymbol{x}_1} \quad \tilde{\boldsymbol{x}_2}] \in \mathbb{F}_2^{n(r_1+r_2)} \mid \\ &\qquad P_{\tilde{\boldsymbol{x}_1}\tilde{\boldsymbol{x}_2}\boldsymbol{x}_1\boldsymbol{x}_2} = P_{\tilde{X}\tilde{Y}XY}\} \end{aligned}$$

we have

$$\begin{aligned} p_e^1 &\leq \sum_{\substack{P_{X\tilde{X}Y\tilde{Y}} \in \mathcal{P}_n^1: \\ \alpha(P_{\tilde{X}Y}) \leq \alpha(P_{XY})}} \sum_{\substack{(\boldsymbol{x}_1,\boldsymbol{x}_2) \in \\ \mathcal{T}_{XY}}} Q^n(\boldsymbol{x}_1\boldsymbol{x}_2) \Pr\Big(\exists(\tilde{\boldsymbol{x}_1}, \tilde{\boldsymbol{x}_2}) \in \\ &\qquad \mathcal{T}_{\tilde{X}\tilde{Y}|XY}(\boldsymbol{x}_1\boldsymbol{x}_2) \text{ s.t. } [\boldsymbol{x}_1 - \tilde{\boldsymbol{x}_1} \quad \boldsymbol{0}]\boldsymbol{AG}_{\mathcal{T}} = \boldsymbol{0}\Big) \\ &\leq \sum_{\substack{P_{X\tilde{X}Y\tilde{Y}} \in \mathcal{P}_n^1: \\ \alpha(P_{\tilde{X}Y}) \leq \alpha(P_{XY})}} \sum_{(\boldsymbol{x}_1,\boldsymbol{x}_2) \in \mathcal{T}_{XY}} Q^n(\boldsymbol{x}_1\boldsymbol{x}_2) \\ &\qquad \min\left\{ \sum_{\substack{(\tilde{\boldsymbol{x}_1},\tilde{\boldsymbol{x}_2}) \in \\ \mathcal{T}_{\tilde{X}\tilde{Y}|XY}(\boldsymbol{x}_1\boldsymbol{x}_2)}} \Pr([\boldsymbol{x}_1 - \tilde{\boldsymbol{x}_1} \; \boldsymbol{0}]\boldsymbol{AG}_{\mathcal{T}} = \boldsymbol{0}), 1 \right\}. \end{aligned}$$

Similarly

$$\begin{aligned} p_e^2 &\leq \sum_{\substack{P_{X\tilde{X}Y\tilde{Y}} \in \mathcal{P}_n^2: \\ \alpha(P_{X\tilde{Y}}) \leq \alpha(P_{XY})}} \sum_{(\boldsymbol{x}_1,\boldsymbol{x}_2) \in \mathcal{T}_{XY}} Q^n(\boldsymbol{x}_1\boldsymbol{x}_2) \\ &\qquad \min\left\{ \sum_{\substack{(\tilde{\boldsymbol{x}_1},\tilde{\boldsymbol{x}_2}) \in \\ \mathcal{T}_{\tilde{X}\tilde{Y}|XY}(\boldsymbol{x}_1\boldsymbol{x}_2)}} \Pr([\boldsymbol{0} \quad \boldsymbol{x}_2 - \tilde{\boldsymbol{x}_2}]\boldsymbol{AG}_{\mathcal{T}} = \boldsymbol{0}), 1 \right\} \\ p_e^3 &\leq \sum_{\substack{P_{X\tilde{X}Y\tilde{Y}} \in \mathcal{P}_n^3: \\ \alpha(P_{\tilde{X}\tilde{Y}}) \leq \alpha(P_{XY})}} \sum_{(\boldsymbol{x}_1,\boldsymbol{x}_2) \in \mathcal{T}_{XY}} Q^n(\boldsymbol{x}_1\boldsymbol{x}_2) \\ &\qquad \min\left\{ \sum_{\substack{(\tilde{\boldsymbol{x}_1},\tilde{\boldsymbol{x}_2}) \in \\ \mathcal{T}_{\tilde{X}\tilde{Y}|XY}(\boldsymbol{x}_1\boldsymbol{x}_2)}} \Pr([\boldsymbol{x}_1 - \tilde{\boldsymbol{x}_1} \quad \boldsymbol{x}_2 - \tilde{\boldsymbol{x}_2}]\boldsymbol{AG}_{\mathcal{T}} = \boldsymbol{0}), 1 \right\} \end{aligned}$$

where the probabilities are taken over realizations of the network transfer matrix $\boldsymbol{AG}_{\mathcal{T}}$ corresponding to the random network code. The probabilities

$$\begin{aligned} P_1 &= \Pr([\boldsymbol{x}_1 - \tilde{\boldsymbol{x}_1} \quad \boldsymbol{0}]\boldsymbol{AG}_{\mathcal{T}} = \boldsymbol{0}) \\ P_2 &= \Pr([\boldsymbol{0} \quad \boldsymbol{x}_2 - \tilde{\boldsymbol{x}_2}]\boldsymbol{AG}_{\mathcal{T}} = \boldsymbol{0}) \\ P_3 &= \Pr([\boldsymbol{x}_1 - \tilde{\boldsymbol{x}_1} \quad \boldsymbol{x}_2 - \tilde{\boldsymbol{x}_2}]\boldsymbol{AG}_{\mathcal{T}} = \boldsymbol{0}) \end{aligned}$$

for nonzero $\boldsymbol{x}_1 - \tilde{\boldsymbol{x}_1}, \boldsymbol{x}_2 - \tilde{\boldsymbol{x}_2}$ can be calculated for a given network, or bounded in terms of n and parameters of the network as we will show later.

As in [4], we can apply some simple cardinality bounds

$$\begin{aligned} |\mathcal{P}_n^1| &< (n+1)^{2^{2r_1+r_2}} \\ |\mathcal{P}_n^2| &< (n+1)^{2^{r_1+2r_2}} \\ |\mathcal{P}_n^3| &< (n+1)^{2^{2r_1+2r_2}} \\ |\mathcal{T}_{XY}| &\leq \exp\{nH(XY)\} \\ |\mathcal{T}_{\tilde{X}\tilde{Y}|XY}(\boldsymbol{x}_1\boldsymbol{x}_2)| &\leq \exp\{nH(\tilde{X}\tilde{Y}|XY)\} \end{aligned}$$

and the identity

$$Q^n(\boldsymbol{x}_1\boldsymbol{x}_2) = \exp\{-n(D(P_{XY}\|Q) + H(XY))\}, \quad (\boldsymbol{x}_1, \boldsymbol{x}_2) \in \mathcal{T}_{XY} \tag{7}$$

to obtain

$$\begin{aligned} p_e^1 &\leq \exp\left\{ -n \min_{\substack{P_{X\tilde{X}Y\tilde{Y}} \in \mathcal{P}_n^1: \\ \alpha(P_{\tilde{X}Y}) \leq \alpha(P_{XY})}} \Big(D(P_{XY}\|Q) + \right. \\ &\qquad \left. \left| -\frac{1}{n}\log P_1 - H(\tilde{X}|XY) \right|^+ \Big) + 2^{2r_1+r_2}\log(n+1) \right\} \\ p_e^2 &\leq \exp\left\{ -n \min_{\substack{P_{X\tilde{X}Y\tilde{Y}} \in \mathcal{P}_n^2: \\ \alpha(P_{X\tilde{Y}}) \leq \alpha(P_{XY})}} \Big(D(P_{XY}\|Q) + \right. \\ &\qquad \left. \left| -\frac{1}{n}\log P_2 - H(\tilde{Y}|XY) \right|^+ \Big) + 2^{r_1+2r_2}\log(n+1) \right\} \\ p_e^3 &\leq \exp\left\{ -n \min_{\substack{P_{X\tilde{X}Y\tilde{Y}} \in \mathcal{P}_n^3: \\ \alpha(P_{\tilde{X}\tilde{Y}}) \leq \alpha(P_{XY})}} \Big(D(P_{XY}\|Q) \right. \\ &\qquad + \left| -\frac{1}{n}\log P_3 - H(\tilde{X}\tilde{Y}|XY) \right|^+ \Big) \\ &\qquad \left. + 2^{2r_1+2r_2}\log(n+1) \right\} \end{aligned}$$

where the exponents and logs are taken with respect to base 2.

For the minimum entropy decoder, we have

$$\alpha(P_{\tilde{X}\tilde{Y}}) \leq \alpha(P_{XY}) \Rightarrow \begin{cases} H(\tilde{X}|XY) \leq H(\tilde{X}|Y) \leq H(X|Y), & \text{for } Y = \tilde{Y} \\ H(\tilde{Y}|XY) \leq H(\tilde{Y}|X) \leq H(Y|X), & \text{for } X = \tilde{X} \\ H(\tilde{X}\tilde{Y}|XY) \leq H(\tilde{X}\tilde{Y}) \leq H(XY) \end{cases}$$

which gives

$$p_e^1 \le \exp\left\{ -n \min_{XY} \left(D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_1 - H(X|Y) \right|^+ \right) + 2^{2r_1+r_2}\log(n+1) \right\} \quad (8)$$

$$p_e^2 \le \exp\left\{ -n \min_{XY} \left(D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_2 - H(Y|X) \right|^+ \right) + 2^{r_1+2r_2}\log(n+1) \right\} \quad (9)$$

$$p_e^3 \le \exp\left\{ -n \min_{XY} \left(D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_3 - H(XY) \right|^+ \right) + 2^{2r_1+2r_2}\log(n+1) \right\}. \quad (10)$$

We next show that these bounds also hold for the maximum Q-probability decoder, for which, from (7),

$$\begin{aligned} \alpha(P_{\tilde{X}\tilde{Y}}) \le \alpha(P_{XY}) &\Rightarrow D(P_{\tilde{X}\tilde{Y}}\|Q) + H(\tilde{X}\tilde{Y}) \\ &\le D(P_{XY}\|Q) + H(XY). \end{aligned} \quad (11)$$

For $i = 1$, $\tilde{Y} = Y$, and (11) gives

$$D(P_{\tilde{X}Y}\|Q) + H(\tilde{X}|Y) \le D(P_{XY}\|Q) + H(X|Y). \quad (12)$$

We show the inequality at the bottom of the page by considering two possible cases for any $X, \tilde{X}, Y$ satisfying (12).

Case 1: $\frac{1}{n}\log P_1 - H(X|Y) < 0$. Then

$$\begin{aligned} D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_1 - H(\tilde{X}|Y) \right|^+ &\ge D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_1 - H(X|Y) \right|^+ \\ &\ge \min_{XY} \left(D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_1 - H(X|Y) \right|^+ \right) \end{aligned}$$

Case 2: $-\frac{1}{n}\log P_1 - H(X|Y) \ge 0$. Then

$$\begin{aligned} D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_1 - H(\tilde{X}|Y) \right|^+ &\ge D(P_{XY}\|Q) + \left(-\frac{1}{n}\log P_1 - H(\tilde{X}|Y) \right) \\ &\ge D(P_{\tilde{X}Y}\|Q) + \left(-\frac{1}{n}\log P_1 - H(X|Y) \right) \text{ by (12)} \\ &= D(P_{\tilde{X}Y}\|Q) + \left| -\frac{1}{n}\log P_1 - H(X|Y) \right|^+ \end{aligned}$$

which gives

$$\begin{aligned} D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_1 - H(\tilde{X}|Y) \right|^+ &\ge \frac{1}{2}\left[D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_1 - H(\tilde{X}|Y) \right|^+ \right. \\ &\quad \left. + D(P_{\tilde{X}Y}\|Q) + \left| -\frac{1}{n}\log P_1 - H(X|Y) \right|^+ \right] \\ &\ge \frac{1}{2}\left[D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_1 - H(X|Y) \right|^+ \right. \\ &\quad \left. + D(P_{\tilde{X}Y}\|Q) + \left| -\frac{1}{n}\log P_1 - H(\tilde{X}|Y) \right|^+ \right] \\ &\ge \min_{XY} \left(D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_1 - H(X|Y) \right|^+ \right). \end{aligned}$$

A similar proof holds for $i = 2$.

For $i = 3$, we show the inequality at the top of the following page by considering two possible cases for any $X, \tilde{X}, Y, \tilde{Y}$ satisfying (11).

Case 1: $-\frac{1}{n}\log P_3 - H(XY) < 0$. Then

$$\begin{aligned} D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_3 - H(\tilde{X}\tilde{Y}) \right|^+ &\ge D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_3 - H(XY) \right|^+ \\ &\ge \min_{XY} \left(D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_3 - H(XY) \right|^+ \right) \end{aligned}$$

$$\begin{aligned} \min_{\substack{P_{X\tilde{X}Y\tilde{Y}} \in \mathcal{P}_n^1: \\ \alpha(P_{\tilde{X}\tilde{Y}}) \le \alpha(P_{XY})}} & \left(D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_1 - H(\tilde{X}|XY) \right|^+ \right) \\ &\ge \min_{\substack{P_{X\tilde{X}Y\tilde{Y}} \in \mathcal{P}_n^1: \\ \alpha(P_{\tilde{X}\tilde{Y}}) \le \alpha(P_{XY})}} \left(D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_1 - H(\tilde{X}|Y) \right|^+ \right) \\ &\ge \min_{XY} \left(D(P_{XY}\|Q) + \left| -\frac{1}{n}\log P_1 - H(X|Y) \right|^+ \right) \end{aligned}$$

$$\min_{\substack{P_{X\tilde{X}Y\tilde{Y}}\in\mathcal{P}_n^3:\\ \alpha(P_{\tilde{X}\tilde{Y}})\le\alpha(P_{XY})}} \left(D(P_{XY}\|Q)+\left|-\frac{1}{n}\log P_3 - H(\tilde{X}\tilde{Y}|XY)\right|^+\right)$$
$$\ge \min_{\substack{P_{X\tilde{X}Y\tilde{Y}}\in\mathcal{P}_n^3:\\ \alpha(P_{\tilde{X}\tilde{Y}})\le\alpha(P_{XY})}} \left(D(P_{XY}\|Q)+\left|-\frac{1}{n}\log P_3 - H(\tilde{X}\tilde{Y})\right|^+\right)$$
$$\ge \min_{XY}\left(D(P_{XY}\|Q)+\left|-\frac{1}{n}\log P_3 - H(XY)\right|^+\right)$$

Case 2: $-\frac{1}{n}\log P_3 - H(XY)\ge 0$. Then

$$\begin{aligned} D(P_{XY}\|Q)&+\left|-\frac{1}{n}\log P_3 - H(\tilde{X}\tilde{Y})\right|^+ \\ &\ge D(P_{XY}\|Q)+\left(-\frac{1}{n}\log P_3 - H(\tilde{X}\tilde{Y})\right) \\ &\ge D(P_{\tilde{X}\tilde{Y}}\|Q)+\left(-\frac{1}{n}\log P_3 - H(XY)\right) \quad \text{by (11)} \\ &= D(P_{\tilde{X}\tilde{Y}}\|Q)+\left|-\frac{1}{n}\log P_3 - H(XY)\right|^+ \end{aligned}$$

which gives

$$\begin{aligned} D(P_{XY}\|Q)&+\left|-\frac{1}{n}\log P_3 - H(\tilde{X}\tilde{Y})\right|^+ \\ &\ge \frac{1}{2}\left[D(P_{XY}\|Q)+\left|-\frac{1}{n}\log P_3 - H(\tilde{X}\tilde{Y})\right|^+\right. \\ &\quad \left.+D(P_{\tilde{X}\tilde{Y}}\|Q)+\left|-\frac{1}{n}\log P_3 - H(XY)\right|^+\right] \\ &\ge \min_{XY}\left(D(P_{XY}\|Q)+\left|-\frac{1}{n}\log P_3 - H(XY)\right|^+\right). \end{aligned}$$

Here the analysis diverges from that of [4], as we consider general networks instead of the simple Slepian–Wolf network. We bound the probabilities P_i in terms of n and the network parameters $m_i, i=1,2$, the minimum cut capacity between the receiver and source X_i, m_3, the minimum cut capacity between the receiver and both sources, and L, the maximum source–receiver path length.

Let $\mathcal{G}_1$ and $\mathcal{G}_2$ be subgraphs of graph $\mathcal{G}$ consisting of all links downstream of sources 1 and 2, respectively, where a link l is considered downstream of a source X_i if $a(i)=o(l)$ or if there is a directed path from the source to $o(l)$. Let $\mathcal{G}_3$ be equal to $\mathcal{G}$.

Note that in a random linear network code, any link l which has at least one nonzero input transmits the zero process with probability $\frac{1}{2^{nc_l}}$, where c_l is the capacity of l. This is the same as the probability that a pair of distinct values for the inputs of l are mapped to the same output value on l.

For a given pair of distinct source values, let E_l be the event that the corresponding inputs to link l are distinct, but the corresponding values on l are the same. Let $E(\tilde{\mathcal{G}})$ be the event that E_l occurs for some link l on every source–receiver path in graph $\tilde{\mathcal{G}}$. P_i is then equal to the probability of event $E(\mathcal{G}_i)$.

Let $\mathcal{G}_i', i=1,2,3$ be the graph consisting of m_i node-disjoint paths, each consisting of L links each of unit capacity. We show by induction on m_i that P_i is upper-bounded by the probability of event $E(\mathcal{G}_i')$.

We let $\tilde{\mathcal{G}}$ be the graphs $\mathcal{G}_i, \mathcal{G}_i', i=1,2,3$ in turn, and consider any particular source–receiver path $\mathcal{P}_{\tilde{\mathcal{G}}}$ in $\tilde{\mathcal{G}}$. We distinguish two cases.

Case 1: E_l does not occur for any of the links l on the path $\mathcal{P}_{\tilde{\mathcal{G}}}$. In this case, the event $E(\tilde{\mathcal{G}})$ occurs with probability 0.

Case 2: There exists some link $\hat{l}$ on the path $\mathcal{P}_{\tilde{\mathcal{G}}}$ for which E_l occurs.

Thus, we have

$$\Pr(E(\tilde{\mathcal{G}})) = \Pr(\text{case 2})\Pr(E(\tilde{\mathcal{G}})|\text{case 2}).$$

Since $\mathcal{P}_{\mathcal{G}_i'}$ has at least as many links as $\mathcal{P}_{\mathcal{G}_i}$

$$\Pr(\text{case 2 for } \mathcal{G}_i') \ge \Pr(\text{case 2 for } \mathcal{G}_i).$$

Therefore, if we can show that

$$\Pr(E(\mathcal{G}_i')|\text{case 2}) \ge \Pr(E(\mathcal{G}_i)|\text{case 2})$$

the induction hypothesis $\Pr(E(\mathcal{G}_i')) \ge \Pr(E(\mathcal{G}_i))$ follows.

For $m_i=1$, the hypothesis is true since $\Pr(E(\mathcal{G}_i')|\text{case 2})=1$. For $m_i>1$, in case 2, removing the link $\hat{l}$ leaves, for $\mathcal{G}_i'$, the effective equivalent of a graph consisting of m_i-1 node-disjoint length-L paths, and, for $\mathcal{G}_i$, a graph of minimum cut at least m_i-1. The result follows from applying the induction hypothesis to the resulting graphs.

Thus, $\Pr(E(\mathcal{G}_i'))$ gives an upper bound on probability P_i

$$\begin{aligned} P_i &\le \left(1-(1-\frac{1}{2^n})^L\right)^{m_i} \\ &\le \left(\frac{L}{2^n}\right)^{m_i}. \end{aligned}$$

Substituting this into the error bounds (8) – (10) gives the desired result. □

C. Random Flooding Versus . Random Coding on a Grid

Proof of Proposition 1: To simplify notation, we assume without loss of generality that the axes are chosen such that the source is at $(0,0)$, and $0<x\le y$. Let $E_{x,y}$ be the event that two different processes are received by a node at grid position (x,y) relative to the source. The statement of the proposition is then

$$\Pr[E_{x,y}] \le \left(1+2^{y-x+1}(4^{x-1}-1)/3\right)/2^{y+x-2} \qquad (13)$$

which we prove by induction.

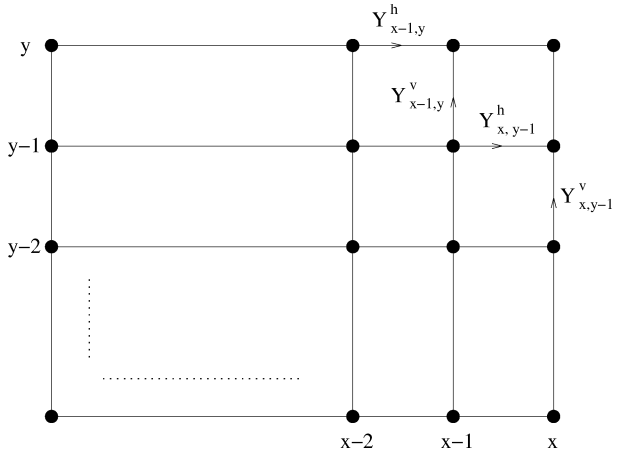

Fig. 4. Rectangular grid network. $Y^h_{x,y}$ denotes the process transmitted on the link between $(x-1,y)$ and (x,y), and $Y^v_{x,y}$ denotes the process transmitted on the link between $(x,y-1)$ and (x,y).

Let $Y^h_{x,y}$ denote the process transmitted on the link between $(x-1,y)$ and (x,y) and let $Y^v_{x,y}$ denote the process transmitted on the link between $(x,y-1)$ and (x,y) (refer to Fig. 4).

Observe that $\Pr[E_{x,y}|E_{x-1,y}] = 1/2$, since with probability $1/2$ node $(x-1,y)$ transmits to node (x,y) the process complementary to whatever process is being transmitted from node $(x,y-1)$. Similarly, $\Pr[E_{x,y}|E_{x,y-1}] = 1/2$, so $\Pr[E_{x,y}|E_{x-1,y} \text{ or } E_{x,y-1}] = 1/2$.

Case 1: $E_{x-1,y-1}$

Case 1a: $Y^h_{x-1,y} \neq Y^v_{x,y-1}$. With probability $\frac{1}{2}$, $Y^v_{x-1,y} \neq Y^h_{x-1,y}$, resulting in $E_{x,y-1} \cup E_{x-1,y}$. With probability $\frac{1}{2}$, $Y^v_{x,y-1} = Y^h_{x,y-1}$, resulting in $E_{x,y}$. So

$$\Pr[E_{x,y}| \text{ Case 1a}] = \frac{1}{2} \times \frac{1}{2} + \frac{1}{2} = \frac{3}{4}.$$

Case 1b: $Y^h_{x-1,y} = Y^v_{x,y-1}$. Either $E_{x,y-1} \cup \overline{E}_{x-1,y}$ or $\overline{E}_{x,y-1} \cup E_{x-1,y}$, so

$$\Pr[E_{x,y}| \text{ Case 1b}] = 1/2.$$

Case 2: $\overline{E}_{x-1,y-1}$

Case 2a: $Y^h_{x-1,y} \neq Y^v_{x,y-1}$. Either $E_{x,y-1} \cup \overline{E}_{x-1,y}$ or $\overline{E}_{x,y-1} \cup E_{x-1,y}$, so

$$\Pr[E_{x,y}| \text{ Case 2a}] = 1/2.$$

Case 2b: $Y^h_{x-1,y} = Y^v_{x,y-1} = Y^h_{x-1,y-1}$. By the assumption of Case 2, $Y^v_{x,y-1}$ is also equal to this same process, and $\Pr[E_{x,y}| \text{ Case 2b}] = 0$.

Case 2c: $Y^h_{x-1,y} = Y^v_{x,y-1} \neq Y^h_{x-1,y-1}$. Then $E_{x,y-1}$ and $E_{x-1,y}$, so $\Pr[E_{x,y}| \text{ Case 2c}] = 1/2$.

So

$$\begin{aligned}\Pr[E_{x,y}|E_{x-1,y-1}] &\leq \max(\Pr[E_{x,y}| \text{ Case 1a}] \\ &\qquad \Pr[E_{x,y}| \text{ Case 1b}]) \\ &= 3/4 \\ \Pr[E_{x,y}|\overline{E}_{x-1,y-1}] &\leq \max(\Pr[E_{x,y}| \text{ Case 2a}], \\ &\qquad \Pr[E_{x,y}| \text{ Case 2b}], \\ &\qquad \Pr[E_{x,y}| \text{ Case 2c}]) \\ &= 1/2 \\ \Pr[E_{x,y}] &\leq \frac{3}{4}\Pr[E_{x-1,y-1}] \\ &\qquad + \frac{1}{2}\Pr[\overline{E}_{x-1,y-1}] \\ &= \frac{1}{2} + \frac{1}{4}\Pr[E_{x-1,y-1}].\end{aligned}$$

If (13) holds for some (x,y), then it also holds for $(x+1,y+1)$

$$\begin{aligned}\Pr[E_{x+1,y+1}] &\leq \frac{1}{2} + \frac{1}{4}\Pr[E_{x,y}] \\ &= \frac{1}{2} + \frac{1}{4}\left(\frac{1 + 2^{y-x+1}(1+4+\cdots+4^{x-2})}{2^{y+x-2}}\right) \\ &= \frac{1 + 2^{y-x+1}(4^x-1)/3}{2^{y+1+x+1-2}}.\end{aligned}$$

Now $\Pr[E_{1,y'}] = 1/2^{y'-1}$, since there are $y'-1$ nodes, $(1,1),\ldots,(1,y'-1)$, at which one of the processes being transmitted to $(1,y')$ is eliminated with probability $1/2$. Setting $y' = y - x + 1$ gives the base case which completes the induction. □

Proof of Proposition 2: In the random coding scheme we consider, the only randomized variables are the $f_{i,j}$ variables at nodes receiving information on two links. The number of such nodes on each source–receiver path is $x+y-2$, so the total degree of P_β is $2(x+y-2)$. Applying Theorem 3 yields the result. □

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their detailed comments and suggestions which helped to substantially improve the presentation of this paper.

REFERENCES

[1] R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[2] K. P. Birman, M. Hayden, O. Ozkasap, Z. Xiao, M. Budiu, and Y. Minsky, "Bimodal multicast," *ACM Trans. Comp. Syst.*, vol. 17, no. 2, pp. 41–88, 1999.

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

[4] I. Csiszar, "Linear codes for sources and source networks: Error exponents, universal coding," *IEEE Trans. Inf. Theory*, vol. IT-28, no. 4, pp. 585–592, Jul. 1982.

[5] R. Dougherty, C. Freiling, and K. Zeger, "Linearity and solvability in multicast networks," *IEEE Tran. Inf. Theory*, vol. 50, no. 10, pp. 2243–2256, Oct. 2004.

[6] M. Feder, D. Ron, and A. Tavory, "Bounds on linear codes for network multicast," *Electronic Colloquium on Computational Complexity*, vol. 10, no. 033, 2003.

[7] C. Fragouli and E. Soljanin, "Information flow decomposition for network coding," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 829–848, Mar. 2004.

[8] T. Ho, D. R. Karger, M. Médard, and R. Koetter, "Network coding from a network flow perspective," in *Proc. IEEE Int. Symp. Information Theory*, Yokohama, Japan, Jun./Jul. 2003, p. 441.

[9] T. Ho, R. Koetter, M. Médard, D. R. Karger, and M. Effros, "The benefits of coding over routing in a randomized setting," in *Proc. IEEE Int. Symp. Information Theory*, Yokohama, Japan, Jun./Jul. 2003, p. 442.

[10] T. Ho, B. Leong, M. Medard, R. Koetter, Y. Chang, and M. Effros, "On the utility of network coding in dynamic environments," in *Proc. Int. Workshop on Wireless Ad-Hoc Networks*, Oulu, Finland, May/Jun. 2004, pp. 196–200.

[11] T. Ho, M. Médard, M. Effros, R. Koetter, and D. R. Karger, "Network coding for correlated sources," in *Proc. Conf. Information Sciences and Systems*, Princeton, NJ, 2004.
[12] T. Ho, M. Médard, J. Shi, M. Effros, and D. R. Karger, "On randomized network coding," in *Proc. 41st Annu. Allerton Conf. Communication, Control, and Computing*, Monticello, IL, Oct. 2003.
[13] T. Ho and H. Viswanathan, "Dynamic algorithms for multicast with intra-session network coding," in *Proc. 43rd Annu. Allerton Conf. Communication, Control, and Computing*, Monticello, IL, Sep. 2005.
[14] S. Jaggi, P. Chou, and K. Jain, "Low complexity algebraic network codes," in *Proc. IEEE Int. Symp. Information Theory*, Yokohama, Japan, Jun./Jul. 2003, p. 368.
[15] R. M. Karp, E. Upfal, and A. Wigderson, "Constructing a perfect matching is in random nc," *Combinatorica*, vol. 6, no. 1, pp. 35–48, 1986.
[16] M. S. Kodialam, T. V. Lakshman, and S. Sengupta, "Online multicast routing with bandwidth guarantees: A new approach using multicast network flow," *Measu. Modeling of Comp. Syst.*, pp. 296–306, 2000.
[17] R. Koetter and M. Médard, "An algebraic approach to network coding," *IEEE/ACM Trans. Netw.*, vol. 11, no. 5, pp. 782–795, Oct. 2003.
[18] A. R. Lehman and E. Lehman, "Complexity classification of network information flow problems," in *Proc. Symp. Discrete Algorithms*, New Orleans, LA, 2004, pp. 142–150.
[19] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Trans. Inf. Theory*, vol. 49, no. 2, pp. 371–381, Feb. 2003.
[20] D. S. Lun, N. Ratnakar, R. Koetter, M. Médard, E. Ahmed, and H. Lee, "Achieving minimum cost multicast: A decentralized approach based on network coding," in *Proc. IEEE Infocom*, Miami, FL, Mar. 2005, pp. 1607–1617.
[21] D. S. Lun, N. Ratnakar, M. Médard, R. Koetter, D. R. Karger, T. Ho, E. Ahmed, and F. Zhao, "Minimum-cost multicast over coded packet networks," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2608–2623, Jun. 2006.
[22] M. Médard, M. Effros, T. Ho, and D. R. Karger, "On coding for non-multicast networks," in *Proc. 41st Annu. Allerton Conf. Communication, Control, and Computing*, Monticello, IL, Oct. 2003.
[23] R. Motwani and P. Raghavan, *Randomized Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 1995.
[24] T. Noguchi, T. Matsuda, and M. Yamamoto, "Performance evaluation of new multicast architecture with network coding," *IEICE Trans. Commun.*, vol. E86-B, no. 6, Jun. 2003.
[25] S. Riis, Linear Versus Non-Linear Boolean Functions in Network Flow Nov. 2003, Preprint.
[26] P. Sanders, S. Egner, and L. Tolhuizen, "Polynomial time algorithms for network information flow," in *Proc. 15th ACM Symp. Parallel Algorithms and Architectures*, San Diego, CA, 2003, pp. 286–294.
[27] S. D. Servetto and G. Barrenechea, "Constrained random walks on random graphs: Routing algorithms for large scale wireless sensor networks," in *Proce. 1st ACM Int. Workshop on Wireless Sensor Networks and Applications*, Atlanta, GA, 2002, pp. 12–21.
[28] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. IT-19, no. 4, pp. 471–480, Jul. 1973.
[29] Y. Zhu, B. Li, and J. Guo, "Multicast with network coding in application-layer overlay networks," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 1, pp. 107–120, Jan. 2004.