Network Routing Capacity

Jillian Cannons, Randall Dougherty, Chris Freiling, and Kenneth Zeger

***Abstract*— We define the routing capacity of a network to be the supremum of all possible fractional message throughputs achievable by routing. We prove that the routing capacity of every network is achievable and rational, we present an algorithm for its computation, and we prove that every rational number in $(0, 1]$ is the routing capacity of some solvable network. We also determine the routing capacity for various example networks. Finally, we discuss the extension of routing capacity to fractional coding solutions and show that the coding capacity of a network is independent of the alphabet used.**

***Index Terms*— network coding, capacity, switching, flow**

I. Introduction

A communications network is a finite, directed, acyclic multigraph over which messages can be transmitted from source nodes to sink nodes. The messages are drawn from a specified alphabet, and the edges over which they are transmitted are taken to be error-free, cost-free, and of zero-delay. Traditionally, network messages are treated as physical commodities, which are routed throughout the network without replication or alteration. However, the emerging field of network coding views the messages as information, which can be copied and transformed by any node within the network. Network coding permits each outgoing edge from a node to carry some function of the data received on the incoming edges of the node. A goal in using network coding is to determine a set of edge functions that allow all of the sink node demands to be satisfied. If such a set of functions exists, then the network is said to be *solvable*, and the functions are called a *solution*. Otherwise the network is said to be *unsolvable*.

A solution to a network is said to be a *routing solution* if the output of every edge function equals a particular one of its inputs. A solution to a network is said to be a *linear solution* if the output of every edge function is a linear combination of its inputs, where linearity is defined with respect to some underlying algebraic structure on the alphabet, usually a finite field or ring. Clearly, a routing solution is also a linear solution.

Network messages are fundamentally scalar quantities, but it is also useful to consider blocks of multiple scalar messages from a common alphabet as message vectors. Such vectors may correspond to multiple time units in a network. Likewise, the data transmitted on each network edge can also be considered as vectors. *Fractional coding* refers to the general case where message vectors differ in dimension from edge data vectors (e.g., see [2]). The coding functions performed at nodes take vectors as input on each in-edge and produce vectors as output on each out-edge. A *vector linear solution* has edge functions which are linear combinations of vectors carried on in-edges to a node, where the linear combination coefficients are matrices over the same alphabet as the input vector components. In a *vector routing solution* each edge function copies a collection of components from input edges into a single output edge vector.

For any set of vector functions which satisfies the demands of the sinks, there is a corresponding scalar solution (by using a Cartesian product alphabet). However, it is known that if a network has a vector routing solution, then it does not necessarily have a scalar routing solution. Similarly, if a network has a vector linear solution, then it does not necessarily have a scalar linear solution [16].

Ahlswede, Cai, Li, and Yeung [1] demonstrated that there exist networks with (linear) coding solutions but with no routing solutions, and they gave necessary conditions for solvability of multicast networks (networks with one source and all messages demanded by all sink nodes). Li, Yeung, and Cai [15] proved that any solvable multicast network has a scalar linear solution over some sufficiently large finite field alphabet. For multicast networks, it is known that solvability over a particular alphabet does not necessarily imply scalar linear solvability over the same alphabet (see examples in [4], [18], [16], [20]). For non-multicast networks, it has recently been shown that solvability does not necessarily imply vector linear solvability [5].

Rasala Lehman and Lehman [19] have noted that for some networks, the size of the alphabet needed for a solution can be significantly reduced if the solution does not operate at the full capacity of the network. In particular, they demonstrated that, for certain networks, fractional coding can achieve a solution where the ratio of edge capacity n to message vector dimension k is an arbitrarily small amount above one. The observations in [19] suggest many important questions regarding network solvability using fractional coding.

In the present paper, we focus on such fractional coding for networks in the special case of routing[1]. We refer to

This work was supported by the Institute for Defense Analyses, the National Science Foundation, and Ericsson.

J. Cannons and **K. Zeger** are with the Department of Electrical and Computer Engineering, University of California, San Diego, La Jolla, CA 92093-0407 (jcannons@code.ucsd.edu , zeger@ucsd.edu).

R. Dougherty is with the Center for Communications Research, 4320 Westerra Court, San Diego, CA 92121-1969 (rdough@ccrwest.org).

C. Freiling is with the Department of Mathematics, California State University, San Bernardino, 5500 University Parkway, San Bernardino, CA 92407-2397 (cfreilin@csusb.edu).

[1] Whereas the present paper studies networks with directed edges, some results on fractional coding were obtained by Li et al. [13], [14] for networks with undirected (i.e., bidirectional) edges.

such coding as *fractional routing*. Specifically, we consider message vectors whose dimension may differ from the dimension of the vectors carried on edges. Only routing is considered, so that at any node, any set of components of the node's input vectors may be sent on the out-edges, provided the edges' capacities are not exceeded.

We define a quantity called the *routing capacity* of a network, which characterizes the highest possible capacity obtainable from a fractional routing solution to a network[2]. The routing capacity is the the largest ratio of message dimension to edge capacity for which a routing solution exists. Analogous definitions can be made of the (general) coding capacity over all (linear and non-linear) network codes and the linear coding capacity over all linear network codes. These definitions are with respect to the specified alphabet and are for general networks (e.g., they are not restricted to multicast networks).

It is known that the linear coding capacity (with respect to a finite field alphabet) can depend on the alphabet size [5] whereas the routing capacity is trivially independent of the alphabet. We prove here, however, that the general coding capacity is independent of the alphabet used.

It is not presently known whether the coding capacity or the linear coding capacity of a network must be rational numbers. Also, it is not presently known if the linear coding capacity of a network is always achievable. It has recently been shown, however, that the (general) coding capacity of a network need not be achievable [6]. We prove here that the routing capacity of every network is achievable (and therefore is also rational). We also show that every rational number in $(0, 1]$ is the routing capacity of some solvable network.

The computability of coding capacities is in general an unsolved problem. For example, it is presently not known whether there exists an algorithm for determining the coding capacity or the linear coding capacity (with respect to a given alphabet size) of a network. We prove here that the routing capacity is indeed computable, by explicitly demonstrating a linear program solution. We do not attempt to give a low complexity or efficient algorithm, as our intent is only to establish the computability of routing capacity.

Section II gives formal definitions of the routing capacity and related network concepts. Section III determines the routing capacity of a variety of sample networks in a semi-tutorial fashion. Section IV proves various properties of the routing capacity, including the result that the routing capacity is achievable and rational. Section V gives the construction of a network with a specified routing capacity. Finally, Section VI defines the coding capacity of a network and shows that it is independent of the alphabet used.

[2]Determining the routing capacity of a (directed) network relates to the maximum throughput problem in an undirected network in which multiple multicast sessions exist (see Li et al. [13], [14]), with each demanded message being represented by a multicast group. In the case where only a single multicast session is present in the network, determining the routing capacity corresponds to fractional directed Steiner tree packing, as considered by Wu, Chou, and Jain [23] and, in the undirected case, by Li et al. [13], [14]. In the case where the (directed) network has disjoint demands (i.e., when each message is only demanded by a single sink), determining the routing capacity resembles the maximum concurrent multicommodity flow problem [22].

II. DEFINITIONS

A *network* is a directed, acyclic multigraph, together with non-empty sets of source nodes, sink[3] nodes, source node messages, and sink node demands. Each message is an arbitrary element of a fixed finite alphabet and is associated with exactly one source node, and each demand at a sink node is a specification of a specific source message that needs to be obtainable at the sink. A network is *degenerate* if there exists a source message demanded at a particular sink, but with no directed path through the graph from the source to the sink.

Each edge in a network carries a vector of symbols from some alphabet. The maximum allowable dimension of these vectors is called the *edge capacity*. (If an edge carries no alphabet symbols, it is viewed as carrying a vector of dimension zero.) Note that a network with nonuniform, rational-valued edge capacities can always be equivalently modeled as a network with uniform edge capacities by introducing parallel edges. For a given finite alphabet, an *edge function* is a mapping, associated with a particular edge (u, v), which takes as inputs the edge vector carried on each in-edge to the node u and the source messages generated at node u, and produces an output vector to be carried on the edge (u, v). A *decoding function* is a mapping, associated with a message demanded at a sink, which takes as inputs the edge vector carried on each in-edge to the sink and the source messages generated at the sink, and produces an output vector hopefully equal to the demanded message.

A *solution* to a network for a given alphabet is an assignment of edge functions to a subset of edges and an assignment of decoding functions to all sinks in the network, such that each sink node obtains all of its demands. A network is *solvable* if it has a solution for some alphabet. A network solution is a *vector routing solution* if every edge function is defined so that each component of its output is copied from a (fixed) component of one of its inputs. (So, in particular, no "source coding" can occur when generating the outputs of source nodes.) It is clear that vector routing solutions do not depend on the chosen alphabet. A solution is *reducible* if it has at least one edge function which, when removed, still yields a solution. A vector solution is *reducible* if it has at least one component of at least one edge function which, when removed, still yields a vector solution.

A (k, n) *fractional routing solution* of a network is a vector routing solution that uses messages with k components and edges with capacity n, with $k, n \geq 1$. Note that if a network is solvable then it must have a (coding) solution

[3]Although the terminology "sink" in graph theory indicated a node with no out-edges, we do not make that restriction here. We merely refer to a node which demands at least one message as a sink.

with $k = n = 1$. A (k, n) fractional routing solution is *minimal* if it is not reducible and if no (k, n') fractional routing solution exists for any $n' < n$. Solvable networks may or may not have routing solutions. However, every non-degenerate network has a (k, n) fractional routing solution for some k and n. In fact, it is easy to construct such a solution by choosing $k = 1$ and n equal to the total number of messages in the network, since then every edge has enough capacity to carry every message that can reach it from the sources.

The ratio k/n in a (k, n) fractional routing solution quantifies the capacity of the solution and the rational number k/n is said to be an *achievable routing rate* of the network. Define the set

$$U = \{r \in \mathbb{Q} : r \text{ is an achievable routing rate}\}.$$

The *routing capacity* of a network is the quantity

$$\epsilon = \sup U.$$

If a network has no achievable routing rate then we make the convention that $\epsilon = 0$. It is clear that $\epsilon = 0$ if and only if the network is degenerate. Also, $\epsilon < \infty$ (e.g., since k/n is trivially upper bounded by the number of edges in the network). Note that the supremum in the definition of ϵ can be restricted to achievable routing rates associated with minimal routing solutions. The routing capacity is said to be *achievable* if it is an achievable routing rate. Note that an achievable routing capacity must be rational. A fractional routing solution is said to *achieve* the routing capacity if the routing rate of the solution is equal to the routing capacity.

Intuitively, for a given network edge capacity, the routing capacity bounds the largest message dimension for which a routing solution exists. If $\epsilon = 0$, then at least one sink has an unsatisfied demand, which implies that no path between the sink and the source emitting the desired message exists. If $\epsilon \in (0, 1)$, then the edge capacities need be inflated with respect to the message dimension to satisfy the demands of the sinks. If $\epsilon = 1$, then it will follow from results in this paper that a fractional routing solution exists where the message dimensions and edge capacities are identical. If $\epsilon > 1$, then the edge capacities need not even be as large as the message dimension to satisfy the demands of the sinks. Finally, if a network has a routing solution, then the routing capacity of the network satisfies $\epsilon \geq 1$.

III. Routing Capacity of Example Networks

To illustrate the concept of the routing capacity, a number of examples are now considered. For each example in this section, let k be the dimension of the messages and let n be the capacity of the edges. All figures in this section have graph nodes labeled by positive integers. Any node labeled by integer i is referred to as n_i. Also, any edge connecting nodes i and j is referred to as $e_{i,j}$ (instead of the usual notation (i, j)), as is the message vector carried by the edge. The distinction between the two meanings of $e_{i,j}$ is made clear in each such instance.

Example III.1. (See Figure 1.)

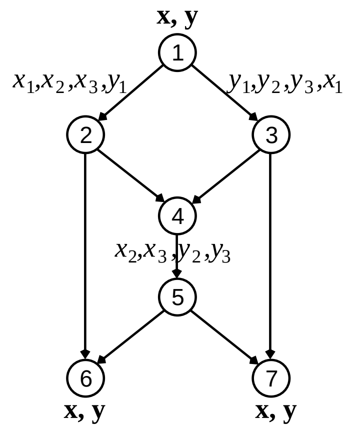

Fig. 1. The multicast network $\mathcal{N}_1$ whose routing capacity is $3/4$.

The single source produces two messages which are both demanded by the two sinks. The network has no routing solution but does have a linear coding solution [1]. The routing capacity of this multicast network is $\epsilon = 3/4$.

Proof. In order to meet the sink node demands, each of the $2k$ message components must be carried on at least two of the three edges $e_{1,2}, e_{1,3}$, and $e_{4,5}$ (because deleting any two of these three edges would make at least one of the sinks unreachable from the source). Hence, we have the requirement $2(2k) \leq 3n$, for arbitrary k and n. Hence $\epsilon \leq 3/4$.

Now, let $k = 3$ and $n = 4$, and route the messages as follows:

$$\begin{aligned} e_{1,2} &= e_{2,6} = (x_1, x_2, x_3, y_1) \\ e_{1,3} &= e_{3,7} = (y_1, y_2, y_3, x_1) \\ e_{2,4} &= (x_2, x_3) \\ e_{3,4} &= (y_2, y_3) \\ e_{4,5} &= (x_2, x_3, y_2, y_3) \\ e_{5,6} &= (y_2, y_3) \\ e_{5,7} &= (x_2, x_3). \end{aligned}$$

This is a fractional routing solution to $\mathcal{N}_1$. Thus, $3/4$ is an achievable routing rate of $\mathcal{N}_1$, so $\epsilon \geq 3/4$. ∎

Example III.2. (See Figure 2.)

Each of the two sources emits a message and both messages are demanded by the two sinks. The network has no routing solution but does have a linear coding solution (similar to Example III.1). The routing capacity of this network is $\epsilon = 1/2$.

Proof. The only path over which message $\mathbf{x}$ can be transmitted from source n_1 to sink n_6 is n_1, n_3, n_4, n_6. Similarly, the only path feasible for the transmission of message $\mathbf{y}$ from source n_2 to sink n_5 is n_2, n_3, n_4, n_5. Thus, there must be sufficient capacity along edge $e_{3,4}$ to accommodate both

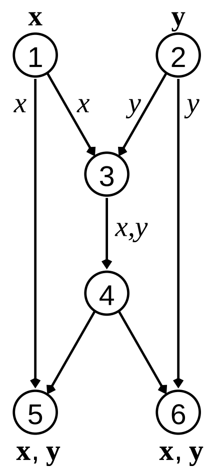

Fig. 2. The network $\mathcal{N}_2$ whose routing capacity is $1/2$.

messages. Hence, we have the requirement $2k \leq n$, yielding $k/n \leq 1/2$ for arbitrary k and n. Thus, $\epsilon \leq 1/2$.

Now, let $k = 1$ and $n = 2$, and route the messages as follows:

$$\begin{aligned} e_{1,5} = e_{1,3} = e_{4,6} &= (\mathbf{x}) \\ e_{2,6} = e_{2,3} = e_{4,5} &= (\mathbf{y}) \\ e_{3,4} &= (\mathbf{x}, \mathbf{y}). \end{aligned}$$

This is a fractional routing solution to $\mathcal{N}_2$. Thus, $1/2$ is an achievable routing rate of $\mathcal{N}_2$, so $\epsilon \geq 1/2$. ■

Example III.3. (See Figure 3.)

The network $\mathcal{N}_3$ contains a single source n_1 with two messages, $\mathbf{x}$ and $\mathbf{y}$. The second layer consists of two nodes, n_2 and n_3. The third and fourth layers each contain $2N$ nodes. The bottom layer contains $\binom{2N}{N}$ sink nodes, where each such node is connected to a distinct set of N nodes from the fourth layer. Each of these sink nodes demands both source messages. The network has no routing solution but does have a linear coding solution for $N \geq 2$ (since the network is multicast and the minimum cut size is 2 for each sink node [15]). The routing capacity of this network is $\epsilon = N/(N+1)$.

Proof. Let $\mathbf{D}$ be a $2k \times 2N$ binary matrix satisfying $\mathbf{D}_{i,j} = 1$ if and only if the i^{th} symbol in the concatenation of messages $\mathbf{x}$ and $\mathbf{y}$ is present on the j^{th} vertical edge between the third and fourth layers. Since the dimension of these vertical edges is at most n, each column of $\mathbf{D}$ has weight at most n. Thus, there are at least $2k-n$ zeros in each column of $\mathbf{D}$ and, therefore, at least $2N(2k-n)$ zeros in the entire matrix.

Since each sink receives input from only N fourth layer nodes and must be able to reconstruct all $2k$ components of the messages, every possible choice of N columns must have at least one 1 in each row. Thus, each row in $\mathbf{D}$ must have weight at least $N+1$, implying that each row in $\mathbf{D}$ has at most $2N - (N+1) = N-1$ zeros. Thus, counting along the rows, $\mathbf{D}$ has at most $2k(N-1)$ zeros. Relating this upper bound and the previously calculated lower bound on the number of zeros yields $2N(2k-n) \leq 2k(N-1)$ or equivalently $k/n \leq N/(N+1)$, for arbitrary k and n. Thus, $\epsilon \leq N/(N+1)$.

Now, let $k = N$ and $n = N+1$, and route the messages as follows:

$$\begin{aligned} e_{1,2} &= (x_1, \ldots, x_k) \\ e_{1,3} &= (y_1, \ldots, y_k) \\ e_{2,i} &= (x_1, \ldots, x_k) && (4 \leq i \leq 2N+3) \\ e_{3,i} &= (y_1, \ldots, y_k) && (4 \leq i \leq 2N+3) \\ e_{i,2N+i} &= (x_1, \ldots, x_k, y_{i-3}) && (4 \leq i \leq N+3) \\ e_{i,2N+i} &= (y_1, \ldots, y_k, x_{i-(N+3)}) && (N+4 \leq i \leq 2N+3). \end{aligned}$$

Each node in the fourth layer simply passes to its out-edges exactly what it receives on its in-edge. If a sink node in the bottom layer is connected to nodes n_i and n_j where $2N+4 \leq i \leq 3N+3$ and $3N+4 \leq j \leq 4N+3$ (i.e., a node in the left half of the fourth layer and a node in the right half of the fourth layer) then the sink receives all of message $\mathbf{x}$ from n_i and all of message $\mathbf{y}$ from n_j. On the other hand, if a sink is connected only to nodes in the left half of the fourth layer, then it receives all of message $\mathbf{x}$ from each such node, and receives a distinct component of message $\mathbf{y}$ from each of the fourth layer nodes, thus giving all of $\mathbf{y}$. A similar situation occurs if a sink node is only connected to fourth layer nodes on the right half.

Thus, this assignment is a fractional routing solution to $\mathcal{N}_3$. Therefore, $N/(N+1)$ is an achievable routing rate of $\mathcal{N}_3$, so $\epsilon \geq N/(N+1)$. ■

Example III.4. (See Figure 4.)

The network $\mathcal{N}_4$ contains a single source n_1 with m messages. The second layer of the network consists of N nodes, each connected to the source via a single edge. The third layer consists of $\binom{N}{I}$ nodes, each receiving a distinct set of I in-edges from the second layer. Each third layer node demands all messages. The network is linearly solvable if and only if $m \leq I$ (since the network is multicast and the minimum cut size is I for each sink node [15]). The routing capacity of this network is $\epsilon = N/(m(N-I+1))$.

Proof. In order to meet the demands of each node in the bottom layer, every subset of I nodes in layer two must receive all mk message components from the source. Thus, each of the mk message components must appear at least $N-(I-1)$ times on the N out-edges of the source (otherwise there would be some set of I of the N layer two nodes not containing some message component). Since the total number of symbols on the N source out-edges is Nn, we must have $mk(N-(I-1)) \leq Nn$ or equivalently $k/n \leq N/(m(N-I+1))$, for arbitrary k and n. Hence, $\epsilon \leq N/(m(N-I+1))$.

Now, let $k = N$ and $n = m(N-I+1)$ and denote the components of the m messages (in some order) by

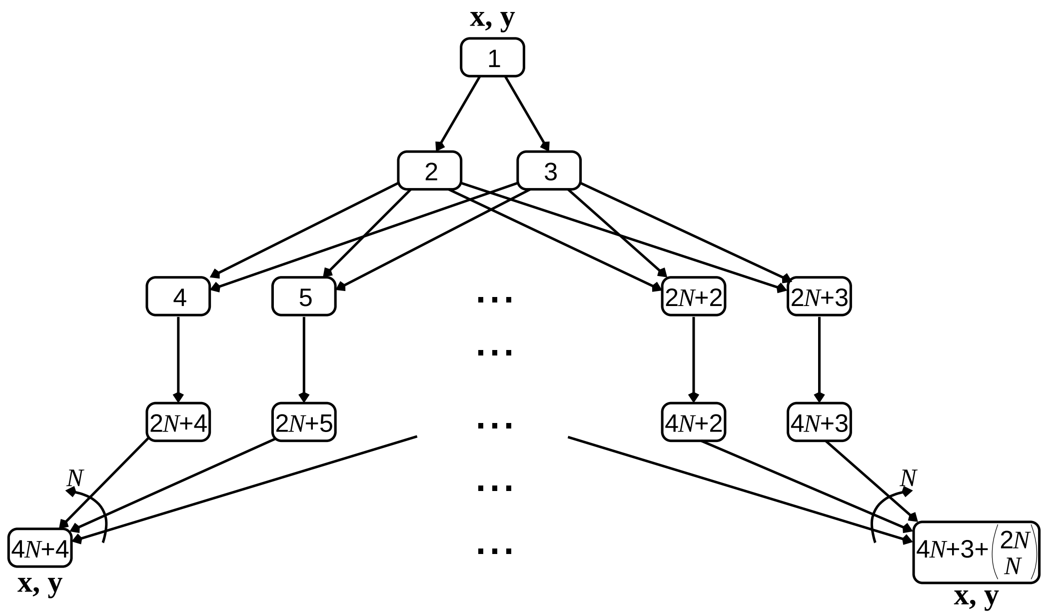

Fig. 3. The multicast network $\mathcal{N}_3$ whose routing capacity is $N/(N+1)$.

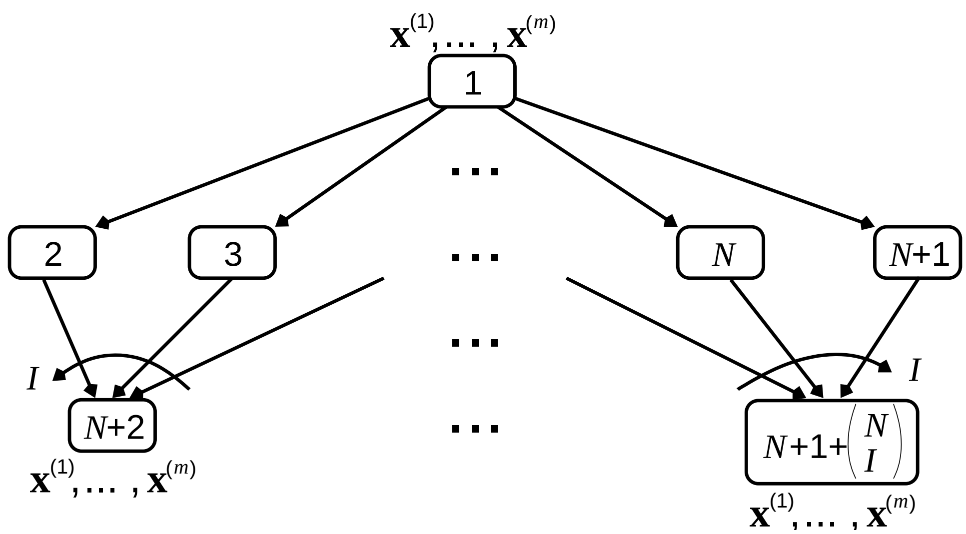

Fig. 4. The multicast network $\mathcal{N}_4$ whose routing capacity is $N/(m(N-I+1))$.

$b_1, \ldots, b_{mk}$. Let $\mathbf{D}$ be an $n \times N$ matrix filled with message components from left to right and from top to bottom, with each message component being repeated $N - I + 1$ times in a row, i.e., $\mathbf{D}_{i,j} = b_{\lfloor (N(i-1)+j-1)/(N-I+1) \rfloor + 1}$ with $1 \leq i \leq m(N-I+1)$ and $1 \leq j \leq N$.

Let the N columns of the matrix determine the vectors carried on the N out-edges of the source. Since each message component is placed in $N-I+1$ different columns of the matrix, every set of I layer two nodes will receive all of the mN message components. The $m(N-I+1) = n$ components at each layer two node are then transmitted directly to all adjacent layer three nodes.

Thus, this assignment is a fractional routing solution to $\mathcal{N}_4$. Therefore, $N/(m(N-I+1))$ is an achievable routing rate of $\mathcal{N}_4$, so $\epsilon \geq N/(m(N-I+1))$. ■

We next note several facts about the network shown in Figure 4.

- The capacity of this network was independently obtained (in a more lengthy argument) by Ngai and Yeung [17]. See also Sanders, Egner, and Tolhuizen [21].
- Ahlswede and Riis [20] studied the case obtained by using the parameters $m = 5, N = 12$, and $I = 8$, which we denote by $\mathcal{N}_5$. They showed that this network has no binary scalar linear solution and yet it has a nonlinear binary scalar solution based upon a $(5, 12, 5)$ Nordstrom-Robinson error correcting code. We note that, by our above calculation, the routing capacity of the Ahlswede-Riis network is $\epsilon = 12/25$.

- Rasala Lehman and Lehman [18] studied the case obtained by using the parameters $m = 2, N = p$, and $I = 2$. They proved that the network is solvable, provided that the alphabet size is at least equal to the square root of the number of sinks. We note that, by our above calculation, the routing capacity of the Rasala Lehman-Lehman network is $\epsilon = p/(2(p-1))$.
- Using the parameters $m = 2$ and $N = I = 3$ illustrates that the network's routing capacity can be greater than 1. In this case, the network consists of a single source, three second layer nodes, and a single third layer node. The routing capacity of this network is $\epsilon = 3/2$.

Example III.5. (See Figure 5.)

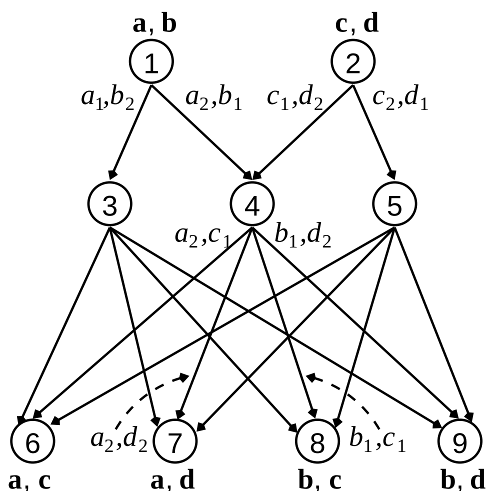

Fig. 5. The network $\mathcal{N}_6$ whose routing capacity is 1.

This network, due to R. Koetter, was used by Médard et al. [16] to demonstrate that there exists a network with no scalar linear solution but with a vector linear solution. The network consists of two sources, each emitting two messages, and four sinks, each demanding two messages. The network has a vector routing solution of dimension two. The routing capacity of this network is $\epsilon = 1$.

Proof. Each source must emit at least $2k$ components and the total capacity of each source's two out-edges is $2n$. Thus, the relation $2k \leq 2n$ must hold, for arbitrary k and n, yielding $\epsilon \leq 1$.

Now let $k = 2$ and $n = 2$, and route the messages as follows (as given in [16]):

$$\begin{array}{lll} e_{1,3} = (a_1, b_2) & e_{1,4} = (a_2, b_1) & \\ e_{2,4} = (c_1, d_2) & e_{2,5} = (c_2, d_1) & \\ e_{3,6} = (a_1) & e_{4,6} = (a_2, c_1) & e_{5,6} = (c_2) \\ e_{3,7} = (a_1) & e_{4,7} = (a_2, d_2) & e_{5,7} = (d_1) \\ e_{3,8} = (b_2) & e_{4,8} = (b_1, c_1) & e_{5,8} = (c_2) \\ e_{3,9} = (b_2) & e_{4,9} = (b_1, d_2) & e_{5,9} = (d_1) \end{array}$$

This is a fractional routing solution to $\mathcal{N}_6$. Thus, 1 is an achievable routing rate of $\mathcal{N}_6$, so $\epsilon \geq 1$. ■

Example III.6. (See Figure 6.)

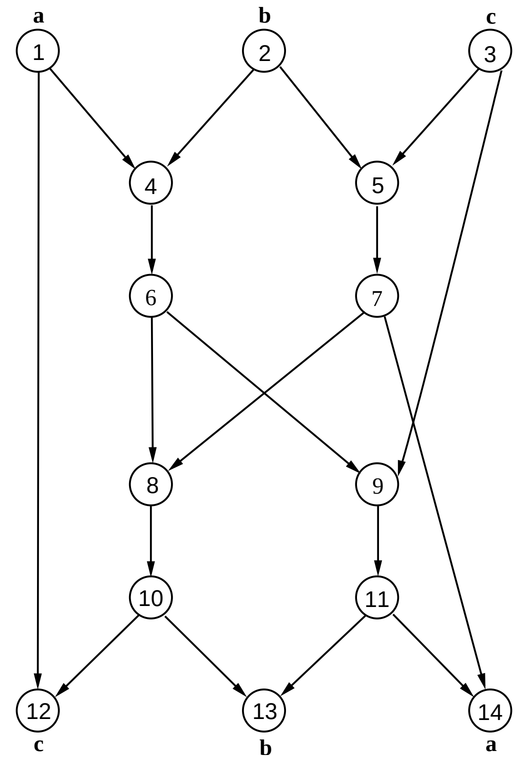

Fig. 6. The network $\mathcal{N}_7$ whose routing capacity is $2/3$.

The network $\mathcal{N}_7$ was demonstrated in [5] to have no linear solution for any vector dimension over a finite field of odd cardinality. The network has three sources n_1, n_2, and n_3 emitting messages **a**, **b**, and **c**, respectively. The messages **c**, **b**, and **a** are demanded by sinks n_{12}, n_{13}, and n_{14}, respectively. The network has no routing solution but does have a coding solution. The routing capacity of this network is $\epsilon = 2/3$.

Proof. First, note that the edges $e_{1,12}, e_{3,9}$, and $e_{7,14}$ cannot have any affect on a fractional routing solution, so they can be removed. Thus, edges $e_{4,6}$ and $e_{5,7}$ must carry all of the information from the sources to the sinks. Therefore, $3k \leq 2n$, for arbitrary k and n, yielding an upper bound on the routing capacity of $\epsilon \leq 2/3$.

Now, let $k = 2$ and $n = 3$ and route the messages as follows:

$$\begin{array}{ll} e_{1,4} = (a_1, a_2) & e_{2,4} = (b_1) \\ e_{2,5} = (b_2) & e_{3,5} = (c_1, c_2) \\ e_{4,6} = (a_1, a_2, b_1) & e_{5,7} = (c_1, c_2, b_2) \\ e_{6,9} = (a_1, a_2, b_1) & e_{7,8} = (b_2, c_1, c_2) \\ e_{8,10} = (b_2, c_1, c_2) & e_{9,11} = (a_1, a_2, b_1) \\ e_{10,12} = (c_1, c_2) & e_{10,13} = (b_2) \\ e_{11,13} = (b_1) & e_{11,14} = (a_1, a_2). \end{array}$$

This is a fractional routing solution to $\mathcal{N}_7$. Thus, $2/3$ is an achievable routing rate of $\mathcal{N}_7$, so $\epsilon \geq 2/3$. ■

Example III.7. (See Figure 7.)

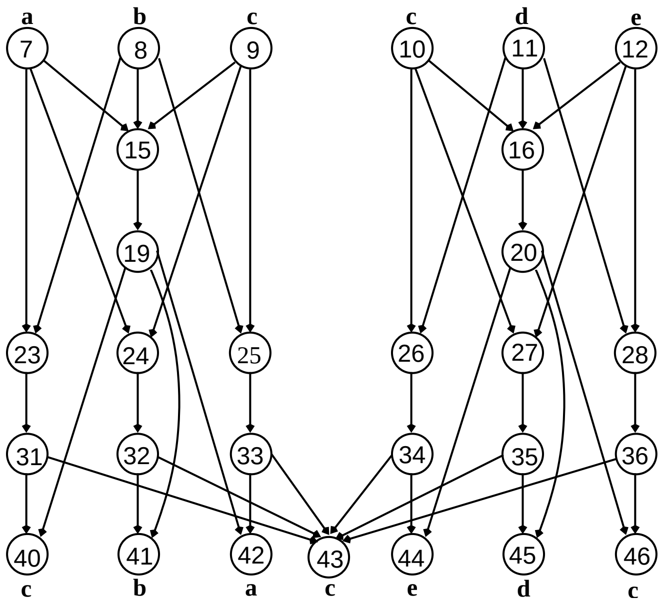

Fig. 7. The network $\mathcal{N}_8$ whose routing capacity is $1/3$.

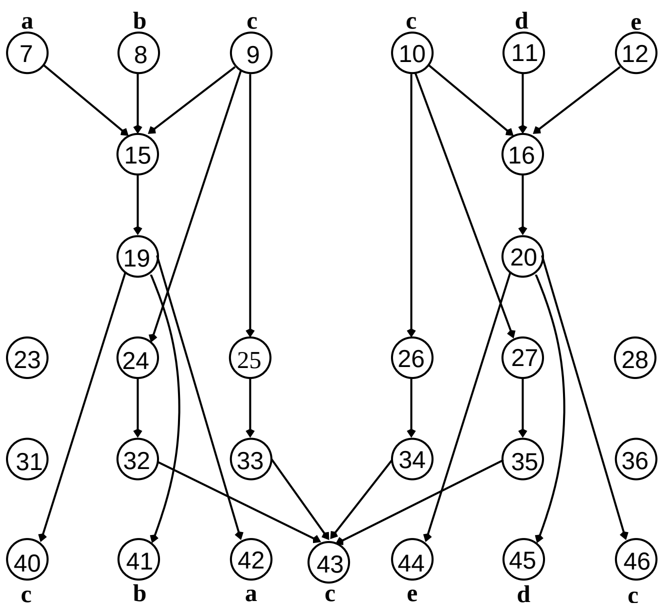

Fig. 8. Reduced form of the network $\mathcal{N}_8$ given in Figure 7.

The network $\mathcal{N}_8$ shown in Figure 7 was given in [5] as a portion of a larger network which was solvable but not vector linearly solvable. This network piece consists of six sources, n_7 through n_{12}, emitting messages $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{c}, \mathbf{d}$, and $\mathbf{e}$, respectively. The network contains seven sinks, n_{40} through n_{46}, demanding messages $\mathbf{c}, \mathbf{b}, \mathbf{a}, \mathbf{c}, \mathbf{e}, \mathbf{d}$, and $\mathbf{c}$, respectively. The network has no routing solution but does have a coding solution. The routing capacity of this network is $\epsilon = 1/3$.

Proof. A number of edges in the network do not affect any fractional routing solution and can be removed, yielding the reduced network shown in Figure 8. Clearly the demands of node n_{43} are easily met. The remaining portion of the network can be divided into two disjoint, symmetric portions. In each case all $3k$ symbols of information must flow across a single edge (either $e_{15,19}$ or $e_{16,20}$), implying that $3k \leq n$. Thus, $\epsilon \leq 1/3$.

Now, let $k = 1$ and $n = 3$ and route the messages as follows:

$$e_{15,19} = (a_1, \ldots, a_k, b_1, \ldots, b_k, c_1, \ldots, c_k)$$
$$e_{16,20} = (c_1, \ldots, c_k, d_1, \ldots, d_k, e_1, \ldots, e_k).$$

This is a fractional routing solution to $\mathcal{N}_8$. Thus, $1/3$ is an achievable routing rate of $\mathcal{N}_8$, so $\epsilon \geq 1/3$. ■

By combining networks $\mathcal{N}_7$ and $\mathcal{N}_8$ (i.e., by adding shared sources a, b, and c) a network was created which established that linear vector codes are not sufficient for all solvable networks [5]. In the combined network, the two pieces effectively operate independently, and thus the routing capacity of the entire network is limited by the second portion, namely $\epsilon = 1/3$.

IV. Routing Capacity Achievability

The examples of the previous section have illustrated various techniques to determine the routing capacity of a network. In this section, some properties of the routing capacity are developed and a concrete method is given, by which the routing capacity of a network can be found.

To begin, a set of inequalities which are satisfied by any minimal fractional routing solution is formulated. These inequalities are then used to prove that the routing capacity of any network is achievable. To facilitate the construction of these inequalities, a variety of subgraphs for a given network are first defined.

Consider a network and its associated graph, $G = (V, E)$, sources S, messages M, and sinks K. For each message $\mathbf{x}$, we say that a directed subgraph of G is an $\mathbf{x}$*-tree* if the subgraph has exactly one directed path from the source emitting $\mathbf{x}$ to each destination node which demands $\mathbf{x}$, and the subgraph is minimal with respect to this property[4]. (Note that such a subgraph can be both an $\mathbf{x}$-tree and a $\mathbf{y}$-tree for distinct messages $\mathbf{x}$ and $\mathbf{y}$.) For each message $\mathbf{x}$, let $s(\mathbf{x})$ denote the number of $\mathbf{x}$-trees. For a given network and for each message $\mathbf{x}$, let $T_1^{\mathbf{x}}, T_2^{\mathbf{x}}, \ldots, T_{s(\mathbf{x})}^{\mathbf{x}}$ be an enumeration

[4]The definition of an $\mathbf{x}$-tree is similar to that of a directed Steiner tree (also known as a Steiner arborescence). Given a directed, edge-weighted graph, a subset of the nodes in the graph, and a root node, a directed Steiner tree is a minimum-weight subgraph which includes a directed path from the root to every other node in the subset [9]. Thus, an $\mathbf{x}$-tree is a directed Steiner tree where the source node is the root node, the subset contains the source and all sinks demanding $\mathbf{x}$, the edge weights are taken to be 0, and with the additional restrictions that only one directed path from the root to each sink is present, and edges not along these directed paths are not included in the subgraph. In the undirected case, the first additional restriction coupled with the 0 edge-weight case corresponds to the requirement that the subgraph be a tree, which is occasionally incorporated in the definition of a Steiner tree [11].

of all the $\mathbf{x}$-trees in the network. Figure 9 depicts all of the $\mathbf{x}$-trees and $\mathbf{y}$-trees for the network $\mathcal{N}_2$ shown in Figure 2.

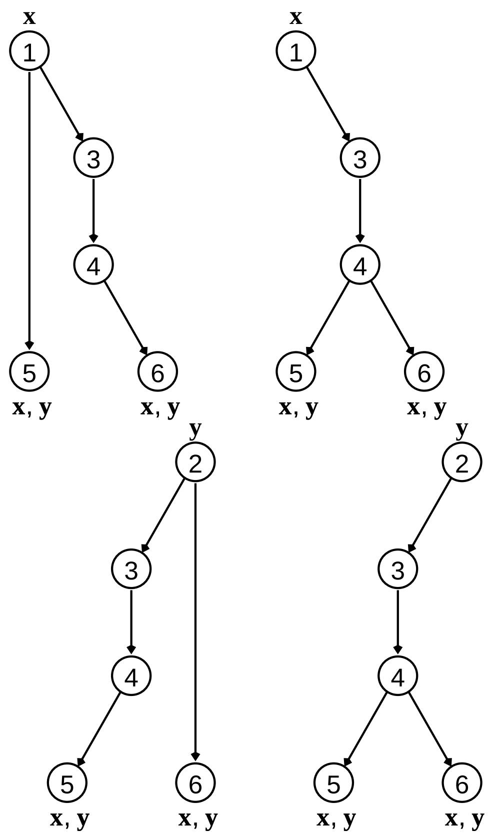

Fig. 9. All of the $\mathbf{x}$-trees and $\mathbf{y}$-trees of the network $\mathcal{N}_2$.

If $\mathbf{x}$ is a message and j is the unique index in a minimal (k,n) fractional routing solution such that every edge carrying a component x_i appears in $T_j^{\mathbf{x}}$, then we say the $\mathbf{x}$-tree $T_j^{\mathbf{x}}$ *carries* the message component x_i. Such a tree is guaranteed to exist since in the supposed solution each message component must be routed from its source to every destination node demanding the message, and the minimality of the solution ensures that the edges carrying the message form an $\mathbf{x}$-tree.

Note that we consider $T_i^{\mathbf{x}}$ and $T_j^{\mathbf{y}}$ to be distinct when $\mathbf{x} \neq \mathbf{y}$, even if they are topologically the same directed subgraph of the network. That is, such trees are determined by their topology together with their associated message.

Denote by T_i the i^{th} tree in some fixed ordering of the set

$$\bigcup_{\mathbf{x}\in M} \{T_1^{\mathbf{x}}, \ldots, T_{s(\mathbf{x})}^{\mathbf{x}}\}$$

and define the following index sets:

$$\begin{aligned} A(\mathbf{x}) &= \{i : T_i \text{ is an } \mathbf{x}\text{-tree}\} \\ B(e) &= \{i : T_i \text{ contains edge } e\}. \end{aligned}$$

Note that the sets $A(\mathbf{x})$ and $B(e)$ are determined by the network, rather than by any particular solution to the network. Denote the total number of trees T_i by

$$t = \sum_{\mathbf{x}\in M} s(\mathbf{x}).$$

For any given minimal (k,n) fractional routing solution, and for each $i = 1, \ldots, t$, let c_i denote the number of message components carried by tree T_i in the given solution.

Lemma IV.1. *For any given minimal (k,n) fractional routing solution to a non-degenerate network, the following inequalities hold:*

(a) $\sum_{i\in A(\mathbf{x})} c_i \geq k \qquad (\forall \mathbf{x} \in M)$

(b) $\sum_{i\in B(e)} c_i \leq n \qquad (\forall e \in E)$

(c) $0 \leq c_i \leq k \qquad (\forall i \in \{1,\ldots,t\})$

(d) $0 \leq n \leq k|M| \leq kt.$

Proof.

(a) Follows from the fact that all k components of every message must be sent to every destination node demanding them.

(b) Follows from the fact that every edge can carry at most n message components.

(c) Follows from that fact that each message has k components.

(d) Since the routing solution is minimal, it must be the case that $n \leq k|M|$, since edge capacities of size $k|M|$ suffice to carry every component of every message. Also, clearly $|M| \leq t$, since the network is non-degenerate.

■

Lemma IV.2. *For any given minimal (k,n) fractional routing solution to a non-degenerate network, the following inequalities, over the real variables $d_1, \ldots, d_t, \rho$, have a rational solution*[5]*:*

$$\sum_{i\in A(\mathbf{x})} d_i \geq 1 \qquad (\forall \mathbf{x} \in M) \qquad (1)$$

$$\sum_{i\in B(e)} d_i \leq \rho \qquad (\forall e \in E) \qquad (2)$$

$$0 \leq d_i \leq 1 \qquad (\forall i \in \{1,\ldots,t\}) \qquad (3)$$

$$0 \leq \rho \leq t \qquad (4)$$

by choosing $d_i = c_i/k$ and $\rho = n/k$.

Proof. (1)–(4) follow immediately from Lemma IV.1(a)–(d), respectively, by division by k.

■

We refer to (1)–(4) as the *network inequalities* associated with a given network.[6] Note that the routing rate in the given (k,n) fractional routing solution in Lemma IV.2 is $1/\rho$.

[5]If a solution $(d_1, \ldots, d_t, \rho)$ to these inequalities has all rational components, then it is said to be a *rational solution*.

[6]Similar inequalities are well-known for undirected network flow problems (e.g., see [11] for the case of single-source networks).

For convenience, define the sets

$$\begin{aligned} V &= \{\rho \in \mathbb{R} : (d_1, \ldots, d_t, \rho) \text{ is a solution to the network inequalities for some } (d_1, \ldots, d_t)\} \\ \hat{V} &= \{r : 1/r \in V\}. \end{aligned}$$

Lemma IV.3. *If the network inequalities corresponding to a non-degenerate network have a rational solution with $\rho > 0$, then there exists a fractional routing solution to the network with achievable routing rate $1/\rho$.*

Proof. Let $(d_1, \ldots, d_t, \rho)$ be a rational solution to the network inequalities with $\rho > 0$. To construct a fractional routing solution, let the dimension k of the messages be equal to the least common multiple of the denominators of the non-zero components of $(d_1, \ldots, d_t, \rho)$. Also, let the capacity of the edges be $n = k\rho$, which is an integer. Now, for each $i = 1, \ldots, t$, let $c_i = d_i k$, each of which is an integer. A (k, n) fractional routing solution can be constructed by, for each message $\mathbf{x}$, arbitrarily partitioning the k components of the message over all $\mathbf{x}$-trees such that exactly c_i components are sent along each associated tree T_i. ■

The following corollary shows that the set U (defined in Section II) of achievable routing rates of any network is the same as the set of reciprocals of rational ρ that satisfy the corresponding network inequalities.

Corollary IV.4. *For any non-degenerate network, $\hat{V} \cap \mathbb{Q} = U$.*

Proof. Lemma IV.2 implies that $U \subseteq \hat{V} \cap \mathbb{Q}$ and Lemma IV.3 implies that $\hat{V} \cap \mathbb{Q} \subseteq U$. ■

We next use the network inequalities to prove that the routing capacity of a network is achievable. To prove this property, the network inequalities are viewed as a set of equations in $t+1$ variables, $d_1, \ldots, d_t, \rho$, which one can attempt to solve. By formulating a linear programming problem, it is possible to determine a fractional routing solution to the network which achieves the routing capacity. As a consequence, the routing capacity of every network is rational and the routing capacity of every non-degenerate network is achievable. The following theorem gives the latter result in more detail.

Theorem IV.5. *The routing capacity of every non-degenerate network is achievable.*

Proof. We first demonstrate that the network inequalities can be used to determine the routing capacity of a network. Let

$$H = \{(d_1, \ldots d_t, \rho) \in \mathbb{R}^{t+1} : \text{ the network inequalities are satisfied}\}$$

$$\rho_0 = \inf V$$

and define the linear function

$$f(d_1, \ldots, d_t, \rho) = \rho.$$

Note that H is non-empty since a rational solution to the network inequalities can be found for any network by setting $d_i = 1$, $\forall i$ and $\rho = t$. Also, since H is compact (i.e., a closed and bounded polytope), the restriction of f to H achieves its infimum ρ_0 on H. Thus, there exist $\hat{d}_1, \ldots, \hat{d}_t \in \mathbb{R}$ such that $(\hat{d}_1, \ldots, \hat{d}_t, \rho_0) \in H$. In fact, a linear program can be used to minimize f on H, yielding ρ_0. Furthermore, since the variables $d_1, \ldots, d_t, \rho$ in the network inequalities have rational coefficients, we can assume without loss of generality that $\hat{d}_1, \ldots, \hat{d}_t, \rho_0 \in \mathbb{Q}$. Now, by Corollary IV.4, we have

$$\begin{aligned} \epsilon &= \sup U \\ &= \sup \left(\hat{V} \cap \mathbb{Q}\right) \\ &= \sup\{r \in \mathbb{Q} : (d_1, \ldots, d_t, 1/r) \in H\} \\ &= \sup\{1/\rho \in \mathbb{Q} : (d_1, \ldots, d_t, \rho) \in H\} \\ &= \max\{1/\rho \in \mathbb{Q} : (d_1, \ldots, d_t, \rho) \in H\} \\ &= 1/\rho_0. \end{aligned}$$

Thus, the network inequalities can be used to determine the routing capacity of a network.

Furthermore, the fractional routing solution induced by the solution $(\hat{d}_1, \ldots, \hat{d}_t, \rho_0)$ to the network inequalities has achievable routing rate $1/\rho_0 = \epsilon$. Thus, the routing capacity of any network is achievable. ■

Corollary IV.6. *The routing capacity of every network is rational.*

Proof. If a network is degenerate, then its capacity is zero, which is rational. Otherwise, Theorem IV.5 guarantees that there exists a (k, n) fractional routing solution such that the routing capacity equals k/n, which is rational. ■

Since any linear programming algorithm (e.g., the simplex method) will work in the proof of Theorem IV.5, we also obtain the following corollary.

Corollary IV.7. *There exists an algorithm for determining the routing capacity of a network.*

We note that the results in Section IV can be generalized to networks whose edge capacities are arbitrary rational numbers. In such case, the term ρ in (2) of the network inequalities would be multiplied by the capacity of the edge e, and the term t in (4) would be multiplied by the maximum edge capacity.

V. Network Construction for Specified Routing Capacity

Given any rational number $r \geq 0$ it is possible to form a network whose routing capacity is $\epsilon = r$. The following two theorems demonstrate how to construct such networks. The first theorem considers the general case when $r \geq 0$, but the resulting network is unsolvable (i.e., for $k = n$) for $r < 1$. The second theorem considers the case when $0 < r \leq 1$ and yields a solvable network.

Theorem V.1. *For each rational $r \geq 0$ there exists a network whose routing capacity is $\epsilon = r$.*

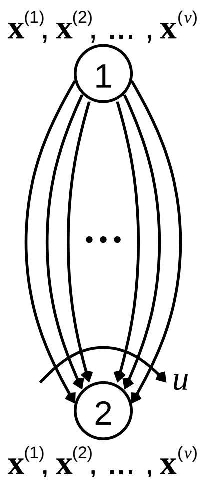

Fig. 10. A network $\mathcal{N}_9$ that has routing capacity $r = u/v \geq 0$.

Proof. If $r = 0$ then any degenerate network suffices. Thus, assume $r > 0$ and let $r = u/v$ where u is a non-negative integer and v is a positive integer. Consider a network with a single source and a single sink connected by u edges, as shown in Figure 10. The source emits messages $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(v)}$ and all messages are demanded by the sink. Let k denote the message dimension and n denote the edge capacity.

In a fractional routing solution, the full vk components must be transferred along the u edges of capacity n. Thus, for a fractional routing solution to exist we require $vk \leq un$ and hence the routing capacity is upper bounded by u/v.

If $k = u$ and $n = v$, then $kv = uv$ message components can be sent arbitrarily along the u edges since the cumulative capacity of all the edges is $nu = vu$. Thus, the routing capacity upper bound is achievable.

Thus, for each rational $r \geq 0$, a single-source, single-sink network can be constructed which has routing capacity $\epsilon = r$.

■

The network $\mathcal{N}_9$ discussed in Theorem V.1 is unsolvable for $0 < r < 1$, since the min cut across the network does not have the required transmission capacity. However, the network is indeed solvable for $r \geq 1$ using a routing solution.

Theorem V.2. *For each rational $r \in (0, 1]$ there exists a solvable network whose routing capacity is $\epsilon = r$.*

Proof. Let $r = p/m$ where $p \leq m$. Consider a network with four layers, as shown in Figure 11 where all edges point downward. The network contains m sources, all in the first layer. Each source emits a unique message, yielding messages $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}$ in the network. The second layer of the network contains p nodes, each of which is connected to all m sources, forming a complete connection between the first and second layers. The third layer also contains p nodes and each is connected in a straight through fashion to a corresponding node in the second layer. The fourth layer consists of m sinks, each demanding all m messages. The third and fourth layers are also completely connected. Finally, each sink is connected to a unique set of $m - 1$ sources, forming a complete connection except the straight through edges between the first and fourth layers. Thus, the network can be thought of as containing both a direct and an indirect route between the sources and sinks.

The routing capacity of this network is now shown to be $\epsilon = r = p/m$. Let k be the dimension of the messages and let n be the capacity of the edges. To begin, the routing capacity is demonstrated to be upper bounded by p/m. First, note that since each sink is directly connected to all but one of the sources and since $r = p/m \leq 1$, each sink can receive all but one of the messages directly. Furthermore, in each case, the missing message must be transmitted to the sink along the indirect route (from the source through the second and third layers to the sink). Since each of the m messages is missing from one of the sinks, a total of mk message components must be transmitted along the indirect paths. The cumulative capacity of the indirect paths is pn, as clearly seen by considering the straight through connections between layers two and three. Thus, the relation $mk \leq pn$ must hold, yielding $k/n \leq p/m$ for arbitrary k and n. Thus $\epsilon \leq p/m$.

To prove that this upper bound on the routing capacity is achievable, consider a solution which sets $k = p$ and $n = m$. As noted previously, direct transmission of $m - 1$ of the messages to each sink is clearly possible. Now, each second layer node receives all k components of all m messages, for a total of $mk = mp$ components. The cumulative capacity of the links from the second to third layers is $pn = pm$. Thus, since the sinks receive all data received by the third layer nodes, the mp message components can be assigned arbitrarily amongst the pm straight through slots, allowing each sink to receive the correct missing message. Hence, this assignment is a fractional routing solution. Therefore, p/m is an achievable routing rate of the network, so $\epsilon \geq p/m$.

Now, the network is shown to be solvable by presenting a solution. Let the alphabet from which the components of the messages are drawn be an Abelian group. As previously, all but one message is received by each source along the direct links from the sources to the sinks. Now, note that node n_{m+1} receives all m messages from the sources. Thus, it is possible to send the combination $\mathbf{x}^{(1)} + \mathbf{x}^{(2)} + \cdots + \mathbf{x}^{(m)}$ along edge $e_{m+1,m+p+1}$. Node n_{m+p+1} then passes this combination along to each of the sinks. Since each sink possesses all but one message, it can extract the missing message from the combination received from node n_{m+p+1}. Thus, the demands of each sink are met.

Hence, the generalized network shown in Figure 11 represents a solvable network whose routing capacity is the rational $r = p/m \in (0, 1]$.

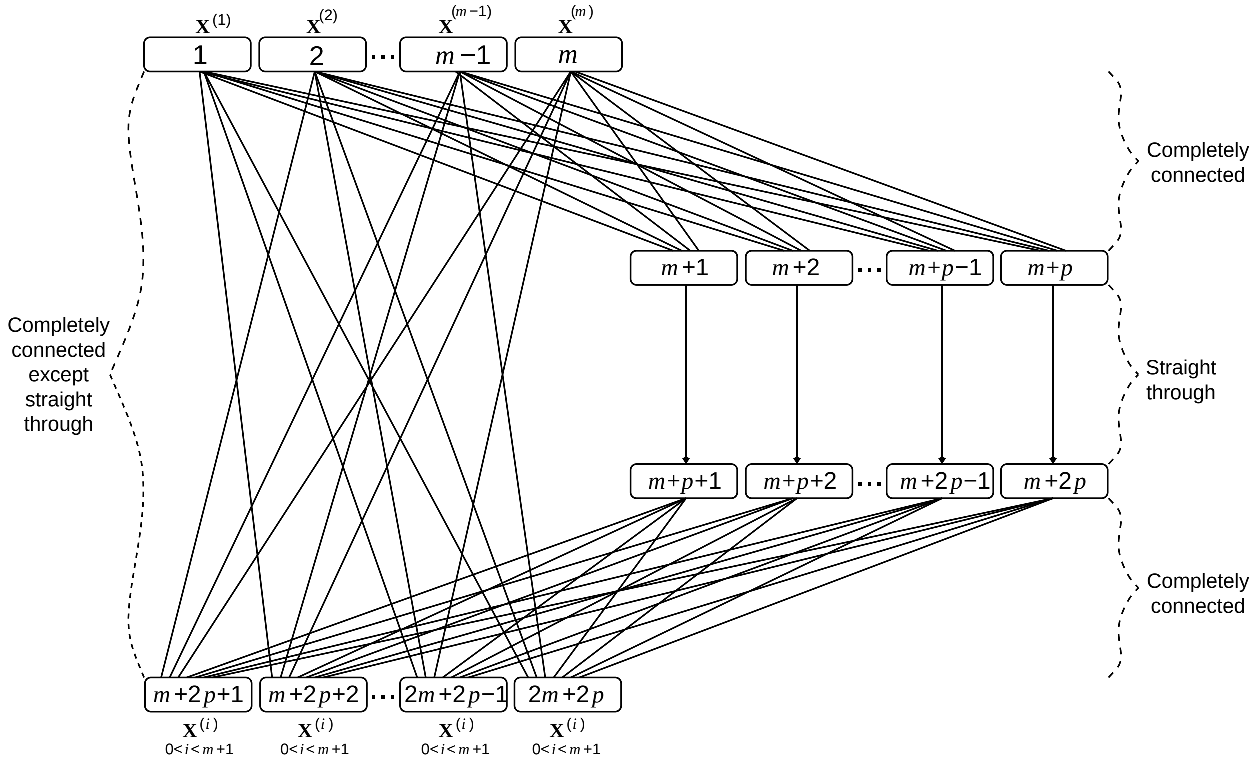

Fig. 11. A solvable network $\mathcal{N}_{10}$ that has routing capacity $r = p/m \in (0, 1]$. All edges in the network point downward.

■

In the network $\mathcal{N}_{10}$, a routing solution (with $k = n$) would require all m messages to be transmitted along the p straight through paths in the indirect portion of the network. However, for $r \in (0, 1)$ we have $p < m$, hence no routing solution exists. Thus, the network requires coding to achieve a solution. Also, note that if the network $\mathcal{N}_{10}$ is specialized to the case $m = 2$ and $p = 1$, then it becomes the network in Figure 2.

VI. Coding Capacity

This section briefly considers the coding capacity of a network, which is a generalization of the routing capacity. The coding capacity is first defined and two examples are then discussed. Finally, it is shown that the coding capacity is independent of the chosen alphabet.

A (k, n) *fractional coding solution* of a network is a coding solution that uses messages with k components and edges with capacity n. If a network has a (k, n) fractional coding solution, then the rational number k/n is said to be an *achievable coding rate*. The *coding capacity* is then given by

$$\gamma = \sup \{r \in \mathbb{Q} : r \text{ is an achievable coding rate}\} .$$

If a (k, n) fractional coding solution uses only linear coding, then k/n is an *achievable linear coding rate* and we define the *linear coding capacity* to be

$$\lambda = \sup \{r \in \mathbb{Q} : r \text{ is an achievable linear coding rate}\} .$$

Note that unlike fractional routing solutions, fractional coding solutions must be considered in the context of a specific alphabet. Indeed, the linear coding capacity in general depends on the alphabet [5]. However, it will be shown in Theorem VI.5 that the coding capacity of a network is independent of the chosen alphabet.

Clearly, for a given alphabet, the coding capacity of a network is always greater than or equal to the linear coding capacity. Also, if a network is solvable (i.e., with $k = n$), then the coding capacity is greater than or equal to 1, since $k/n = k/k$ is an achievable coding rate. Similarly, if a network is linearly solvable, then the linear coding capacity is greater than or equal to 1.

The following examples illustrate the difference between the routing capacity and coding capacity of a network.

Example VI.1. The special case $\mathcal{N}_5$ of the network shown in Figure 4 has routing capacity $\epsilon = 12/25$, as discussed in the note following Example III.4. Using a cut argument, it is clear that the coding capacity of the network is upper bounded by $8/5$, since each sink demands $5k$ message components and has a total capacity of $8n$ on its incoming edges. Lemmas VI.2 and VI.3 will respectively prove that this network has a scalar linear solution for every finite field other than $GF(2)$ and has a vector linear solution for $GF(2)$. Consequently, the linear coding capacity for any finite field alphabet is at least 1, which is strictly greater than the routing capacity.

Lemma VI.2. *Network $\mathcal{N}_5$ has a scalar linear solution for* *every finite field alphabet other than $GF(2)$.*

Proof. Let $a, b, c, d,$ and e be the messages at the source. Let the alphabet be a finite field F with $|F| > 2$. Let $z \in F - \{0, 1\}$. Define the following sets (D is a multiset):

$$\begin{aligned} A &= \{a, b, c, d, e\} \\ B &= \{za + b, zb + c, zc + d, zd + e, ze + a\} \\ C &= \{a + b + c + d + e\} \\ D &= A \cup B \cup C \cup C. \end{aligned}$$

Then $|D| = 12$. Let the symbols carried on the 12 edges emanating from the source correspond to a specific permutation of the 12 elements of D. We will show that the demands of all $\binom{12}{8}$ sinks are satisfied by showing that all of the messages $a, b, c, d,$ and e can be recovered (linearly) from every multiset $S \subset D$ satisfying $|S| = 8$.

If $|S \cap A| = 5$ then the recovery is trivial.

If $|S \cap A| = 4$ then without loss of generality assume $e \notin S$. If $a + b + c + d + e \in S$ then e can clearly be recovered. If $a + b + c + d + e \notin S$ then $|S \cap B| = 4$, in which case $\{zd + e, ze + a\} \cap S \neq \emptyset$, and thus e can be recovered.

If $|S \cap A| = 1$ then $B \subset S$, so the remaining 4 elements of A can be recovered.

If $|S \cap A| = 2$ then $|B \cap S| \geq 4$, so the remaining 3 elements of A can be recovered.

If $|S \cap A| = 3$ then $|B \cap S| \geq 3$. If $|B \cap S| \geq 4$, then the remaining 2 elements of A can be recovered, so assume $|B \cap S| = 3$, in which case $a + b + c + d + e \in S$. Due to the symmetries of the elements in B, we assume without loss of generality that $A \cap S \in \{\{a, b, c\}, \{a, b, d\}\}$. First consider the case when $A \cap S = \{a, b, c\}$. Then, $d + e$ can be recovered. If $zd + e \in S$ then we can solve for d and e since $z \neq 1$. If $zd + e \notin S$ then $S \cap \{zc + d, ze + a\} \neq \emptyset$, so either d can be recovered from c and $zc + d$ or e can be recovered from a and $ze + a$. Then, the remaining term is recoverable from $d + e$. Now consider the case when $A \cap S = \{a, b, d\}$. Then, $c + e$ can be recovered. If $S \cap \{zb + c, zc + d\} \neq \emptyset$ then c can be recovered from either b and $zb + c$ or d and $zc + d$. If $S \cap \{zb + c, zc + d\} = \emptyset$ then $S \cap \{zd + e, ze + a\} \neq \emptyset$, so e can be recovered from either d and $zd + e$ or a and $ze + a$. Finally, the remaining term can be recovered from $c + e$.

■

Lemma VI.3. *Network $\mathcal{N}_5$ has a binary linear solution for vector dimension 2.*

Proof. Consider a scalar linear solution over $GF(4)$ (which is known to exist by Lemma VI.2). The elements of $GF(4)$ can be viewed as the following four 2×2 matrices over $GF(2)$:

$$\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}.$$

Then, using the $GF(4)$ solution from Lemma VI.2 and substituting in the matrix representation yields the following 12 linear functions of dimension 2 for the second layer of the network:

$$\begin{pmatrix} a_1 \\ a_2 \end{pmatrix}, \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, \begin{pmatrix} d_1 \\ d_2 \end{pmatrix}, \begin{pmatrix} e_1 \\ e_2 \end{pmatrix},$$
$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} - \begin{pmatrix} b_1 \\ b_2 \end{pmatrix},$$
$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} - \begin{pmatrix} c_1 \\ c_2 \end{pmatrix},$$
$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} - \begin{pmatrix} d_1 \\ d_2 \end{pmatrix},$$
$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} - \begin{pmatrix} e_1 \\ e_2 \end{pmatrix},$$
$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \end{pmatrix} - \begin{pmatrix} a_1 \\ a_2 \end{pmatrix},$$
$$\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} + \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} + \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} + \begin{pmatrix} e_1 \\ e_2 \end{pmatrix},$$
$$\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} + \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} + \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} + \begin{pmatrix} e_1 \\ e_2 \end{pmatrix}.$$

It is straightforward to verify that from any 8 of these 12 vector linear functions, one can linearly obtain the 5 message vectors $\binom{a_1}{a_2}, \binom{b_1}{b_2}, \binom{c_1}{c_2}, \binom{d_1}{d_2}, \binom{e_1}{e_2}$.

■

Example VI.4. As considered in Example III.1, the network $\mathcal{N}_1$ has routing capacity $\epsilon = 3/4$. We now show that both the coding and linear coding capacities are equal to 1, which is strictly greater than the routing capacity.

Proof. Network $\mathcal{N}_1$ has a well known scalar linear solution [1] given by

$$\begin{aligned} e_{1,2} = e_{2,4} = e_{2,6} &= x \\ e_{1,3} = e_{3,4} = e_{3,7} &= y \\ e_{4,5} = e_{5,6} = e_{5,7} &= x + y. \end{aligned}$$

Thus, $\lambda \geq 1$ and $\gamma \geq 1$.

To upper bound the coding and linear coding capacities, note that each sink demands both messages but only possesses two incoming edges. Thus, we have the requirement $2k \leq 2n$, for arbitrary k and n. Hence, $\lambda \leq 1$ and $\gamma \leq 1$.

■

Theorem VI.5. *The coding capacity of any network is independent of the alphabet used.*

Proof. Suppose a network has a (k, n) fractional coding solution over an alphabet A and let B be any other alphabet of cardinality at least two. Let $\epsilon > 0$ and let

$$t = \left\lceil \frac{(k+1)\log_2 |B|}{n\epsilon \log_2 |A|} \right\rceil.$$

There is clearly a (tk, tn) fractional coding solution over the alphabet A obtained by independently applying the (k, n)

solution t times. Define the quantities

$$n' = n\left\lceil t \cdot \frac{\log_2 |A|}{\log_2 |B|}\right\rceil$$
$$k' = \left\lfloor \frac{kn'}{n} \right\rfloor - k$$

and notice by some computation that

$$|B|^{n'} \geq |A|^{tn} \quad (5)$$
$$|B|^{k'} \leq |A|^{tk} \quad (6)$$
$$\frac{k'}{n'} \geq \frac{k}{n} - \epsilon. \quad (7)$$

For each edge e let d_e and m_e respectively be the number of relevant in-edges and messages originating at the starting node of e, and for each node v let d_v and m_v respectively be the relevant number of in-edges and messages originating at v. For each edge e, denote the edge encoding function for e by

$$f_e : (A^{tn})^{d_e} \times (A^{tk})^{m_e} \to A^{tn}$$

and for each node v and each message $\mathbf{m}$ demanded by v denote the corresponding node decoding function by

$$f_{v,\mathbf{m}} : (A^{tn})^{d_v} \times (A^{tk})^{m_v} \to A^{tk}.$$

The function f_e determines the vector carried on the out-edge e of a node based upon the vectors carried on the in-edges and the message vectors originating at the same node. The function $f_{v,\mathbf{m}}$ attempts to produce the message vector $\mathbf{m}$ as a function of the vectors carried on the in-edges of the node v and the message vectors originating at v. Let $h : A^{tn} \to B^{n'}$ and $h_0 : B^{k'} \to A^{tk}$ be any injections (they exist by (5) and (6)). Define $\hat{h} : B^{n'} \to A^{tn}$ such that $\hat{h}(h(\mathbf{x})) = \mathbf{x}$ for all $\mathbf{x} \in A^{tn}$ and $\hat{h}(\mathbf{x})$ is arbitrary otherwise. Also, define $\hat{h}_0 : A^{tk} \to B^{k'}$ such that $\hat{h}_0(h_0(\mathbf{x})) = \mathbf{x}$ for all $\mathbf{x} \in B^{k'}$ and $\hat{h}_0(\mathbf{x})$ is arbitrary otherwise. Define for each edge e the mapping

$$g_e : (B^{n'})^{d_e} \times (B^{k'})^{m_e} \to B^{n'}$$

by

$$g_e(\mathbf{x}_1, \ldots, \mathbf{x}_{d_e}, \mathbf{y}_1, \ldots, \mathbf{y}_{m_e}) = h(f_e(\hat{h}(\mathbf{x}_1), \ldots, \hat{h}(\mathbf{x}_{d_e}), h_0(\mathbf{y}_1), \ldots, h_0(\mathbf{y}_{m_e})))$$

for all $\mathbf{x}_1, \ldots, \mathbf{x}_{d_e} \in B^{n'}$ and for all $\mathbf{y}_1, \ldots, \mathbf{y}_{m_e} \in B^{k'}$. Similarly, define for each node v and each message $\mathbf{m}$ demanded at v the mapping

$$g_{v,\mathbf{m}} : (B^{n'})^{d_v} \times (B^{k'})^{m_v} \to B^{k'}$$

by

$$g_{v,\mathbf{m}}(\mathbf{x}_1, \ldots, \mathbf{x}_{d_v}, \mathbf{y}_1, \ldots, \mathbf{y}_{m_v}) = \hat{h}_0(f_{v,\mathbf{m}}(\hat{h}(\mathbf{x}_1), \ldots, \hat{h}(\mathbf{x}_{d_v}), h_0(\mathbf{y}_1), \ldots, h_0(\mathbf{y}_{m_v})))$$

for all $\mathbf{x}_1, \ldots, \mathbf{x}_{d_v} \in B^{n'}$ and for all $\mathbf{y}_1, \ldots, \mathbf{y}_{m_v} \in B^{k'}$.

Now consider the (k', n') fractional network code over the alphabet B obtained by using the edge functions g_e and decoding functions $g_{v,\mathbf{m}}$. For each edge in the network, the vector carried on the edge in the (k, n) solution over the alphabet A and the vector carried on the edge in the (k', n') fractional network code over B can each be obtained from the other using h and $\hat{h}$, and likewise for the vectors obtained at sink nodes from the decoding functions for the alphabets A and B (using h_0 and $\hat{h}_0$). Thus, the set of edge functions g_e and decoding functions $g_{v,\mathbf{m}}$ gives a (k', n') fractional routing solution of the network over alphabet B, since the vector on every edge in the solution over A can be determined (using h, h_0, $\hat{h}$, and $\hat{h}_0$) from the vector on the same edge in the solution over B. The (k', n') solution achieves a rate of k'/n', which by (7) is at least $(k/n) - \epsilon$. Since ϵ was chosen as an arbitrary positive number, the supremum of achievable rates of the network over the alphabet B is at least k/n. Thus, if a coding rate is achievable by one alphabet, then that rate is a lower bound to the coding capacity for all alphabets. This implies the network coding capacity (the supremum of achievable rates) is the same for all alphabets. ■

There are numerous interesting open questions regarding coding capacity, some of which we now mention. Is the coding capacity (resp. linear coding capacity) achievable and/or rational for every network? For which networks is the linear coding capacity smaller than the coding capacity, and for which networks is the routing capacity smaller than the linear coding capacity? Do there exist algorithms for computing the coding capacity and linear coding capacity of networks?

VII. Conclusions

This paper formally defined the concept of the routing capacity of a network and proved a variety of related properties. When fractional routing is used to solve a network, the dimension of the messages need not be the same as the capacity of the edges. The routing capacity provides an indication of the largest possible fractional usage of the edges for which a fractional routing solution exits. A variety of sample networks were considered to illustrate the notion of the routing capacity. Through a constructive procedure, the routing capacity of any network was shown to be achievable and rational. Furthermore, it was demonstrated that every rational number in $(0, 1]$ is the routing capacity of some solvable network. Finally, the coding capacity of a network was also defined and was proven to be independent of the alphabet used.

The results in this paper straightforwardly generalize to (not necessarily acyclic) undirected networks and to directed networks with cycles as well. Also, the results can be generalized to networks with nonuniform (but rational) edge capacities; in such case, some extra coefficients are required in the network inequalities. An interesting future problem would be to find a more efficient algorithm for computing the routing capacity of a network.

VIII. Acknowledgment

The authors thank Emina Soljanin and Raymond Yeung for providing helpful references.

References

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, July 2000.

[2] C. Chekuri, C. Fragouli, E. Soljanin "On average throughput benefits and alphabet size in network coding," *IEEE Transactions on Information Theory* (submitted March 2005).

[3] P. A. Chou, Y. Wu, and K. Jain, "Practical network coding," *Allerton Conference on Communication, Control, and Computing*, Monticello, IL, October 2003.

[4] R. Dougherty, C. Freiling, and K. Zeger, "Linearity and solvability in multicast networks," *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2243–2256, October 2004.

[5] R. Dougherty, C. Freiling, and K. Zeger, "Insufficiency of linear coding in network information flow," *IEEE Transactions on Information Theory*, vol. 51, no. 8, pp. 2745–2759, August 2005.

[6] R. Dougherty, C. Freiling, and K. Zeger, "Unachievability of network coding capacity", joint issue of *IEEE Transactions on Information Theory & IEEE/ACM Transactions on Networking*, June 2006, (to appear).

[7] M. Feder, D. Ron, and A. Tavory, "Bounds on linear codes for network multicast," *Electronic Colloquium on Computational Complexity (ECCC)*, Report 33, pp. 1–9, 2003.

[8] T. Ho, D. Karger, M. Médard, and R. Koetter, "Network coding from a network flow perspective," *IEEE International Symposium on Information Theory*, Yokohama, Japan, p. 441, June 2003.

[9] F. K. Hwang, D. S. Richards, and P. Winter, *The Steiner Tree Problem, Volume 53 of the Annals of Discrete Mathematics*. Amsterdam, Netherlands: Elsevier Science Publishers (North-Holland), 1992.

[10] S. Jaggi, P. Sanders, P. A. Chou, M. Effros, S. Egner, K. Jain, and L. Tolhuizen, "Polynomial time algorithms for multicast network code construction," *IEEE Transactions on Information Theory*, vol. 51, no. 6, pp. 1973–1982, June 2005.

[11] K. Jain, M. Mahdian, and M. Salavatipour, "Packing Steiner trees," *14th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 266–274, Baltimore, MD, January 2003.

[12] R. Koetter and M. Médard, "An algebraic approach to network coding," *IEEE/ACM Transactions on Networking*, vol. 11, no. 5, pp. 782–795, October 2003.

[13] Z. Li and B. Li, "Network coding in undirected networks," *38th Annual Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, paper #603, March 2004.

[14] Z. Li, B. Li, D. Jiang, and L. C. Lau, "On achieving optimal end-to-end throughput in data networks: Theoretical and empirical studies," ECE Technical Report, Department of Electrical and Computer Engineering, University of Toronto, (available on-line at http://www.eecg.toronto.edu/∼bli/research.html), February 2004.

[15] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Transactions on Information Theory*, vol. 49, no. 2, pp. 371–381, February 2003.

[16] M. Médard, M. Effros, T. Ho, D. Karger, "On coding for non-multicast networks," *41st Annual Allerton Conference on Communication Control and Computing*, Monticello, IL, October 2003.

[17] C. K. Ngai and R. W. Yeung, "Network coding gain of combination networks," *IEEE Information Theory Workshop (ITW)*, San Antonio, TX, January 2005.

[18] A. Rasala Lehman and E. Lehman, "Complexity classification of network information flow problems," *41st Annual Allerton Conference on Communication Control and Computing*, Monticello, IL, October 2003.

[19] A. Rasala Lehman and E. Lehman, "Network information flow: Does the model need tuning?," *Symposium on Discrete Algorithms (SODA)*, pp. 499–504, Vancouver, British Columbia, Canada, January 2005.

[20] S. Riis, "Linear versus non-linear boolean functions in network flow," *38th Annual Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, March 2004.

[21] P. Sanders, S. Egner, L. Tolhuizen, "Polynomial time algorithms for network information flow", *15th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pp. 286–294, San Diego, CA, June 2003.

[22] F. Shahrokhi and D. W. Matula, "The maximum concurrent flow problem," *Journal of the ACM*, vol. 37, no. 2, pp. 318–334, 1990.

[23] Y. Wu, P. A. Chou, and K. Jain, "A comparison of network coding and tree packing," *2004 IEEE International Symposium on Information Theory (ISIT)*, p. 145, Chicago, IL, June 2004.

[24] R. W. Yeung, *A First Course in Information Theory*. New York, NY: Kluwer, 2002.

Jillian Cannons (S'99) received the B.Sc. degree in computer engineering with distinction from the University of Manitoba, Winnipeg, Canada, in 2000, and the M.S. degree in electrical engineering from the University of Illinois at Urbana-Champaign, Urbana, IL, in 2002.

She is currently pursuing the Ph.D. degree at the University of California, San Diego, where she is a member of the Information Coding Laboratory. Her current research focus is the area of network coding.

Randall Dougherty was born in 1961. He received the A.B. and Ph.D. degrees in mathematics from the University of California (Berkeley) in 1981 and 1985, respectively. After postdoctoral work at Caltech and UCLA, he joined the mathematics faculty of The Ohio State University in 1989, becoming an Associate Professor in 1992 and a Professor in 1998. In 2000 he left Ohio State to work for the software company LizardTech in Seattle; in 2002 he started to work full time at the Center for Communications Research (CCR) in La Jolla, California. He also spent shorter periods working as a consultant in various organizations, such as CCR and the Los Alamos National Laboratory. He is a three-time International Mathematical Olympiad medalist and a three-time Putnam Fellow. He received an NSF Presidential Young Investigator Award in 1991 and a Sloan Research Fellowship in 1992. His mathematical research interests range from foundations (logic and set theory) to other pure mathematics (real analysis, group theory, combinatorics) to applied work (numerical methods, network design, computational geometry, etc.).

Christopher Freiling was born in 1954, received his Ph.D. from UCLA under the direction of Donald Martin. He has been on the faculty at California State University at San Bernardino since 1983, interrupted by two visiting appointments at UCLA. His mathematical interests include Foundations and Real Analysis. He is the recipient of the 2003 Andy Award in Real Analysis.

Kenneth Zeger (S'85-M'90-SM'95-F'00) was born in Boston in 1963. He received both the S.B. and S.M. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology in 1984, and both the M.A. degree in mathematics and the Ph.D. in electrical engineering at the University of California, Santa Barbara, in 1989 and 1990, respectively. He was an Assistant Professor of Electrical Engineering at the University of Hawaii from 1990 to 1992. He was in the Department of Electrical and Computer Engineering and the Coordinated Science Laboratory at the University of Illinois at Urbana-Champaign, as an Assistant Professor from 1992 to 1995, and as an Associate Professor from 1995 to 1996. He has been in the Department of Electrical and Computer Engineering at the University of California at San Diego, as an Associate Professor from 1996 to 1998, and as a Professor from 1998 to present. He received an NSF Presidential Young Investigator Award in 1991. He served as Associate Editor At-Large for the IEEE TRANSACTIONS ON INFORMATION THEORY during 1995-1998, as a member of the Board of Governors of the IEEE Information Theory Society during 1998-2000 and 2005-2007, and is an IEEE Fellow.