Precoding for Broadcasting with Linear Network Codes

Qiyue Zou[*], *Aria Nosratinia*[‡], *and Ali H. Sayed*[*]

$*$ Emails: $\{$eqyzou,sayed$\}$@ee.ucla.edu
Electrical Engineering Department
University of California
Los Angeles, CA 90095

‡ Email: aria@utdallas.edu
Department of Electrical Engineering
University of Texas at Dallas
Richardson, TX 75080

***Abstract*— A technique based on linear precoding is introduced for broadcasting on linear networks. The precoding allows the different message components of a broadcast message to be separated and decoded at the desired sink nodes, thus providing a systematic design methodology for broadcasting over a given network with a given linear network code. To achieve a good throughput, however, the network code itself must also be chosen judiciously. Motivated by several recent results on random network codes, we propose a combination of precoding and random linear network codes. This approach does not require a centralized coordination for network code design. One of the advantages of this approach is that by simply changing the precoding matrix (together with associated decoding strategies), different broadcast objectives can be achieved without tampering with the network code, therefore one can manage the network operation by controlling the origin and destination nodes of the network and without manipulating the network interior. Together, random network codes and linear precodings provide a simple yet powerful methodology for broadcast over linear networks.**

I. Introduction

In this work we consider the capacity of error-free networks for the purpose of broadcasting from one transmit node to multiple receive nodes. The subject of error-free networks and coding over such networks, also known as network coding, has received much attention lately. An overview of several important results in this area can be found in [1] and [2].

The multicast problem, where all sink nodes receive the same information, has been studied in a number of past works. In [3], Ahlswede *et al.* showed that with network coding a source node can multicast information to the sink nodes at a rate equal to the smallest minimum cut capacity between the source node and any sink node. Li *et al.* [4] further showed that linear network codes are sufficient for multicast. Koetter and Médard [5] presented an algebraic framework for network coding and gave an algebraic characterization of the multicast problem. Ho *et al.* [6] proposed a distributed random linear coding approach for multicast, and showed that it achieves optimal multicast capacity with probability exponentially approaching 1 with the code length.

When the information transmitted to sink nodes are not identical, the problem is referred to as *broadcast.* The broadcast problem is more general and has proved to be significantly more difficult to solve than multicast. In [7], [8], [9], the achievable rate region is derived for broadcast networks with two sink nodes. However, these works are based on graph theoretic arguments and their approach does not easily extend to general networks with more than two sink nodes.

In this paper, a systematic approach is proposed for network broadcasting, where the information messages are first precoded at the source node before being encoded by a predefined linear network code in such a way that each sink node can successfully decode its desired messages. The idea is used in Section III to produce an alternative proof of the achievability of the rate region for the broadcast networks with two sink nodes. In Section IV, the precoding technique is applied to networks with multiple sink nodes, and the design criterion for linear precoding is derived. A method based on the idea of *interference minimization* is proposed in Section V for constructing the precoding matrix. Although this method is not demonstrated to be optimal, the corresponding algorithm provides excellent results, which are demonstrated with several examples. In Section VI, we apply the proposed broadcast scheme to randomly generated linear network codes, which does not require a centralized authority for code design and can achieve different broadcast objectives by simply changing the precoding matrix at the source node and the associated decoding scheme at the sink nodes.

A. Notation and System Model

An *acyclic* network is a network without a directed cycle. Consider an acyclic communication network with a single source node that is represented by a pair (G, s), where $G = (V, E)$ is a directed graph specified by the set V of nodes and the set E of edges. Each edge $e \in E$ in the graph G represents a noiseless communication channel on which only one data unit can be transmitted per unit time. The capacity of direct transmission between two nodes is determined by the multiplicity of edges between them.

Assume there are l sink nodes t_1, t_2, …, t_l in the network, and information data are broadcast from s, the unique source node, to these sink nodes. We wish to include in this methodology both individual messages, as well as messages that are shared among arbitrary subsets of sink nodes. Let W be the power set of the set $\{t_1, t_2, \ldots, t_l\}$ of sink nodes excluding the empty set. We denote a generic subset of the sink nodes with $w \in W$. For convenience purposes, we need to index these subsets by integers j, taking values from 1 to $2^l - 1$

(the empty set is not needed). To do this, we map all the nonempty subsets of $\{t_1, t_2, \ldots, t_l\}$ to the integers from 1 to $2^l - 1$ according to the *lexicographical* order. For example, in the case of three sink nodes, we represent the nonempty subsets of $\{t_1, t_2, t_3\}$ as follows:

$$\{t_1\} \to 1, \quad \{t_2\} \to 2, \quad \{t_3\} \to 3, \quad \{t_1, t_2\} \to 4,$$
$$\{t_1, t_3\} \to 5, \quad \{t_2, t_3\} \to 6, \quad \{t_1, t_2, t_3\} \to 7.$$

With this notation, we can denote a subset of sink nodes by w_j, and the subscript indicates that the index of the subset is j.

The information message X_j, $j = 1, 2, \ldots, 2^l - 1$, with entropy rate r_j, is generated at s for the subset w_j of sink nodes. Transmission of X_j is deemed successful if all sink nodes $t \in w_j$ receive X_j.[1] The messages X_j for all $j = 1, 2, \ldots, 2^l - 1$ are mutually independent. The study of this network and its capabilities can be summarized in the following two questions. What rates $\{r_j : j = 1, 2, \ldots, 2^l - 1\}$ can this network support, and what (coding) strategy should we use to arrive at an achievable rate? This paper strives to answer these questions for the broadcast problem, using linear network codes.

We adopt the following assumptions throughout the paper:

(1) Data symbols transmitted along each edge of the graph are elements of a finite field F with size $|F|$.

(2) The entropy rate of the information messages is measured in terms of F-valued symbols. Each F-valued symbol carries $\log_2 |F|$ bits of information. For example, we can say that the entropy rate of X is three F-valued symbols per unit time.

(3) The capacity of each edge in the network is one F-valued symbol per unit time.

(4) The entropy rates r_j are integers for all $j = 1, 2, \ldots, 2^l - 1$. This assumption can be approximately achieved by choosing a proper time unit for network operations.

In this paper we use the following notation. $F^{n\times m}$ denotes the space of n-by-m matrices over a field F; dim$(\cdot)$ returns the dimension of a vector space, and span$\{\cdot\}$ is the linear subspace spanned by the vectors in its argument; rank$(\cdot)$ and $(\cdot)^T$ represent the rank and transpose of a matrix; $|\cdot|$ denotes the cardinality of a set; In(v) and Out(v) are the sets of incoming and outgoing edges of node v, respectively; mincut$\{\Psi, \Phi\}$ denotes the minimum cut capacity between the set of nodes Ψ and the set of nodes Φ. With a slight abuse of notation, we also write mincut$\{s, \Phi\}$, where s is a single node, and in that case s represents the set $\{s\}$.

II. Review of Linear Network Codes

We first give a formal definition of linear network codes, and then illustrate it by using the famous butterfly network as an example. For more detailed explanation, we refer to [4] and [2].

As seen in the famous butterfly example (see below), the advantage of a network code is that it allows the transmission of more symbols per transmission interval, compared to simple routing (or time division multiplexing). To arrive at this advantage, each node in the network combines its inputs and calculates the symbols to be sent on its outgoing edges. In linear network coding, this input-output relation is linear; therefore, all messages in the graph are linear functions of the transmitted signal by the source.

One may therefore formulate a linear network code as follows. Assuming the source can transmit n symbols per transmission into the network, we combine these symbols into a row vector $\mathbf{c}$. All the symbols flowing in the network, at that point in time, can be considered a function of $\mathbf{c}$. In a linear network code, the signal on each edge e is a linear functional of $\mathbf{c}$, namely $\mathbf{c}\mathbf{f}_e$ (the inner product between $\mathbf{c}$ and $\mathbf{f}_e$), where $\mathbf{f}_e$ is a column vector associated with the edge. The span of the functionals on the input of each node, which we call P_v, is equivalent to the signal space observed at that node. Obviously the signals that a node can transmit must reside in its observed signal space, and thus there must be conditions between the functionals on the incoming and outgoing edges of each node.

These ideas are demonstrated in the butterfly network in Figure 2, where on the left, we see the well-known signaling that achieves the rate of 2 symbols per transmission out of the source. On the right side, we see the equivalent edge vectors $\mathbf{f}_e$. Obviously, if the only incoming edge of a node has vector $[1\ \ 0]^T$, the outgoing edge cannot have, e.g., vector $[1\ \ 1]^T$ because the corresponding signal is not available to that node. The above ideas can be formalized, using linear spaces, as follows [4].

Definition 1: Let F be a finite field and n a positive integer. An n-dimensional F-valued linear network code on an acyclic communication network is defined by assigning a vector $\mathbf{f}_e \in F^{n\times 1}$ to each edge $e \in E$ and a vector subspace $P_v \subseteq F^{n\times 1}$

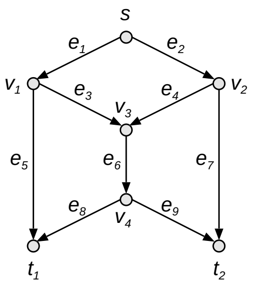

Fig. 1. Labeling of nodes and edges on the butterfly network.

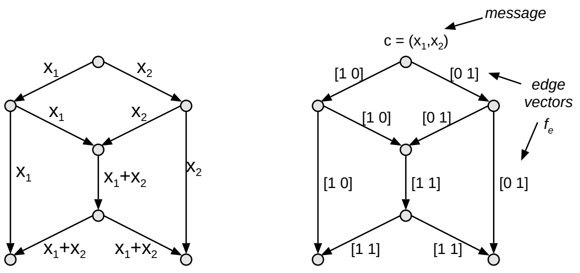

Fig. 2. Signaling and related functionals in the butterfly network.

[1] Note that we allow $r_j = 0$ in this setting.

to each node $v \in V$ such that:

(1) for the source node s, $P_s = F^{n\times 1}$;
(2) for each non-source node v, $P_v = \text{span}\{\mathbf{f}_e : e \in \text{In}(v)\}$;
(3) for each edge $e \in \text{Out}(v)$, $\mathbf{f}_e \in P_v$.

Let $\mathbf{c} \in F^{1\times n}$ be the vector of data emanating from node s. The symbol transmitted along edge e is given by the inner product $\mathbf{c}\mathbf{f}_e$. ■

Remark 1: As a consequence of the above definition, if v is a non-source node and $e \in \text{Out}(v)$, the symbol transmitted along e is a linear combination of the symbols received by v.

Example 1: For the butterfly network shown in Figures 1 and 2, the edge vectors $\mathbf{f}_{e_i}$ and the corresponding signal spaces are as follows:

$$\begin{aligned}
&\mathbf{c} = \mathbf{x} = [x_1 \;\; x_2],\\
&\mathbf{f}_{e_1} = \mathbf{f}_{e_3} = \mathbf{f}_{e_5} = [1 \;\; 0]^T,\\
&\mathbf{f}_{e_2} = \mathbf{f}_{e_4} = \mathbf{f}_{e_7} = [0 \;\; 1]^T,\\
&\mathbf{f}_{e_6} = \mathbf{f}_{e_8} = \mathbf{f}_{e_9} = [1 \;\; 1]^T,\\
&P_s = F^{2\times 1},\\
&P_{v_1} = \text{span}\big\{\mathbf{f}_{e_1}\big\} = \text{span}\big\{[1 \;\; 0]^T\big\},\\
&P_{v_2} = \text{span}\big\{\mathbf{f}_{e_2}\big\} = \text{span}\big\{[0 \;\; 1]^T\big\},\\
&P_{v_3} = \text{span}\big\{\mathbf{f}_{e_3},\, \mathbf{f}_{e_4}\big\} = \text{span}\big\{[1 \;\; 0]^T,\, [0 \;\; 1]^T\big\} = F^{2\times 1},\\
&P_{v_4} = \text{span}\big\{\mathbf{f}_{e_6}\big\} = \text{span}\big\{[1 \;\; 1]^T\big\},\\
&P_{t_1} = \text{span}\big\{\mathbf{f}_{e_5},\, \mathbf{f}_{e_8}\big\} = \text{span}\big\{[1 \;\; 0]^T,\, [1 \;\; 1]^T\big\} = F^{2\times 1},\\
&P_{t_2} = \text{span}\big\{\mathbf{f}_{e_7},\, \mathbf{f}_{e_9}\big\} = \text{span}\big\{[0 \;\; 1]^T,\, [1 \;\; 1]^T\big\} = F^{2\times 1}.
\end{aligned}$$

■

We now turn our focus to a class of network codes that have a wide diversity of edge vectors, because as is evident from the butterfly example, we would like the subspaces P_{t_i} associated with the sink nodes to have as large a dimension as possible. This property will result in a large information flow to the sink nodes. This idea can be expressed in terms of linear independence of subsets of edge vectors, as follows.

Definition 2: Consider a network in which the following property holds. For any arbitrary set of m edges e_i, where $m \le n$, and the set of their originating nodes $\{v_k\}$, where $e_k \in \text{Out}(v_k)$, if we have

$$P_{v_k} \nsubseteq \text{span}\left\{\mathbf{f}_{e_i} : i = 1, 2, \ldots, m,\, i \ne k\right\},\;\; k = 1, 2, \ldots, m,$$

it follows that the m edge vectors $\mathbf{f}_{e_i}$ are linearly independent. Then the code on this network is called a *generic* linear network code. ■

If the base field is sufficiently large, it has been shown in [2] that a generic linear network code always exists and furthermore it can be systematically constructed.

Another interesting class is the so-called *linear dispersion* codes, where the information available in any subset of nodes is directly related to the cutset between them and the source. In a manner of speaking, one may say that information transfer in linear dispersion codes (in a dimensional sense) is efficient.

Definition 3: A network code is a *linear dispersion* code if for every collection Φ of non-source nodes,

$$\dim\big(\text{span}\big\{\cup_{v\in\Phi} P_v\big\}\big) = \min\big\{n,\, \text{mincut}\{s, \Phi\}\big\}.$$

■

Theorem 1 ([2]): Every generic linear network code is a linear dispersion code. ■

III. Precoding for Network Broadcasting

This section introduces the idea of precoding for network broadcasting, and in the process provides an alternative proof of the sufficiency of the cutset condition on the rates of a two-sink-node broadcast network.[2] The proof also provides a code construction derived from any arbitrary generic linear network codes to achieve any rates satisfying the cutset condition given in the following theorem.

Theorem 2 ([7], [8]): For networks with one source and two sink nodes, any rates $\{r_1, r_2, r_3\}$ satisfying the following bounds are achievable:

$$\begin{aligned}
\text{mincut}\big\{s, \{t_1\}\big\} &\ge r_1 + r_3\\
\text{mincut}\big\{s, \{t_2\}\big\} &\ge r_2 + r_3\\
\text{mincut}\big\{s, \{t_1, t_2\}\big\} &\ge r_1 + r_2 + r_3.
\end{aligned} \tag{1}$$

■

To prove the theorem, we first establish the following lemma.

Lemma 1: (The Precoding Lemma) For an arbitrary linear network code over a single-source, two-sink network, define:

$$\begin{aligned}
c_1 &= \dim(P_1) - \dim(P_1 \cap P_2)\\
c_2 &= \dim(P_2) - \dim(P_1 \cap P_2)\\
c_3 &= \dim(P_1 \cap P_2),
\end{aligned}$$

where we adopt the notation that $P_i = P_{t_i}$. Then rate c_1 can be privately transmitted to node t_1, rate c_2 can be privately transmitted to node t_2, and rate c_3 can be multicast to nodes t_1 and t_2. ■

Proof: Consider the space $S = \text{span}\{P_1 \cup P_2\}$ which is the entire signal space under consideration at this point. We wish to identify subspaces corresponding to private messages and common messages (via the corresponding bases). The basis vectors for private messages are called $\mathbf{a}_{1,k}$ and $\mathbf{a}_{2,k}$, and those for common messages are $\mathbf{a}_{3,k}$. We follow an intuitive design and choose each of the private signal spaces to avoid interference with the other node. For example, we choose $\mathbf{a}_{1,k}$ from $P_2^\perp$, where here $P_2^\perp$ denotes the orthogonal complement space to P_2 in S, i.e., $S = P_2 \oplus P_2^\perp$ (and, hence, $\dim(P_2^\perp) = c_1$). But we also want to allow maximum possible rate, which translates into larger spans, and so we choose $\mathbf{a}_{1,k}$ such that, together with any basis of P_2, will result in a basis for $\text{span}\{P_1 \cup P_2\}$. The common message is sent through a subspace that is visible to both nodes, i.e., it can be a subspace of $P_1 \cap P_2$. To summarize:

(1) Find c_1 vectors $\mathbf{a}_{1,k}$ in $P_2^\perp$ such that, together with a basis for P_2, result in a basis for $\text{span}\{P_1 \cup P_2\}$.
(2) Find c_2 vectors $\mathbf{a}_{2,k}$ in $P_1^\perp$ such that, together with a basis for P_1, result in a basis for $\text{span}\{P_1 \cup P_2\}$.
(3) Find c_3 vectors $\mathbf{a}_{3,k}$ to form a basis for the subspace $P_1 \cap P_2$.

[2] The original proof appeared in two independent works [7] and [8] using graph-theoretic arguments.

Now let the code vector $\mathbf{c}$ generated at s be given by

$$\mathbf{c} = \sum_{k=1}^{c_1} x_{1,k}\mathbf{a}_{1,k}^T + \sum_{k=1}^{c_2} x_{2,k}\mathbf{a}_{2,k}^T + \sum_{k=1}^{c_3} x_{3,k}\mathbf{a}_{3,k}^T, \qquad (2)$$

where $x_{1,k}$ and $x_{2,k}$ are data intended for t_1 and t_2, respectively, and $x_{3,k}$ are data intended for both t_1 and t_2. For convenience, we concatenate all data into one vector and all basis elements into one matrix, that is

$$\mathbf{x} = [\mathbf{x}_1 \;\; \mathbf{x}_2 \;\; \mathbf{x}_3]$$

where $\mathbf{x}_i = [x_{i,1} \;\; x_{i,2} \;\; \ldots \;\; x_{i,c_i}]$, and

$$\mathbf{Q} = [\mathbf{Q}_1 \;\; \mathbf{Q}_2 \;\; \mathbf{Q}_3]^T$$

where $\mathbf{Q}_i = [\mathbf{a}_{i,1} \;\; \mathbf{a}_{i,2} \;\; \ldots \;\; \mathbf{a}_{i,c_i}]$. Using matrix notation, Equation (2) can be rewritten as

$$\mathbf{c} = \mathbf{x}\,\mathbf{Q}.$$

Here $\mathbf{Q}$ is a matrix that maps our "segmented" data vector $\mathbf{x}$ to the generic data vector $\mathbf{c}$ at the source node. Thus we consider $\mathbf{Q}$ as a pre-coder. We now show that each of the receive nodes can decode its intended data.

At the receiver side, t_1 receives the following signal on one of its incoming edges e:

$$\begin{aligned} y_e &= \mathbf{c}\,\mathbf{f}_e = \mathbf{x}\mathbf{Q}\,\mathbf{f}_e \\ &= [\mathbf{x}_1 \; \mathbf{x}_3] \begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_3^T \end{bmatrix} \; \mathbf{f}_e \end{aligned}$$

since $\mathbf{f}_e \in P_1$ and, hence, $\mathbf{Q}_2^T\mathbf{f}_e = 0$. We now concatenate all incoming signals to t_1 into one vector $\mathbf{y}_1$, and all corresponding edge vectors $\mathbf{f}_e$ are collected into a matrix $\mathbf{F}_1$. Then we have

$$\mathbf{y}_1 = [\mathbf{x}_1 \; \mathbf{x}_3] \begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_3^T \end{bmatrix} \; \mathbf{F}_1$$

Since the matrix multiplying $[\mathbf{x}_1 \;\; \mathbf{x}_3]$ has full rank (see Appendix B for a proof), the system of equations above has a unique solution and the destination node can recover the data that is intended for it. A similar argument holds for t_2. Consequently, we can send $x_{1,k}$, to t_1, $x_{2,k}$ to t_2, and $x_{3,k}$ to both t_1 and t_2, simultaneously. This concludes the proof of the precoding Lemma. ■

Proof: [Theorem 2]

Recall that messages X_1, X_2 and X_3 are the private and common messages to the two nodes. We wish to consider all possible ways of messaging; therefore, we note the following: we do not care if nodes receive each other's private messages, and thus a private message may, in principle, be transmitted using both private and common parts of the available rate (although using common rate is wasteful). In a similar manner, the common message can be transmitted by using both private and common components of available rates, although using private rates requires duplicating the message and is wasteful.

By Theorem 1, if $n \geq \text{mincut}\{s, \{t_1, t_2\}\}$, we can construct an n-dimensional generic linear network code such that

$$\begin{aligned} \dim(P_1) &= \text{mincut}\{s, \{t_1\}\} = c_1 + c_3 \\ \dim(P_2) &= \text{mincut}\{s, \{t_2\}\} = c_2 + c_3 \\ \dim(\text{span}\{P_1 \cup P_2\}) &= \text{mincut}\{s, \{t_1, t_2\}\} = c_1 + c_2 + c_3. \end{aligned}$$

For the generic linear network code, if (1) holds, i.e.,

$$\begin{aligned} c_1 + c_3 &\geq r_1 + r_3 \\ c_2 + c_3 &\geq r_2 + r_3 \\ c_1 + c_2 + c_3 &\geq r_1 + r_2 + r_3 \end{aligned}$$

then we can encode the information messages X_1, X_2 and X_3 as follows. If $r_i \leq c_i$, it means X_i can be transmitted completely with rate c_i or less. If any $r_i > c_i$, it means transmission of X_i needs to "borrow" rate from other components, namely, X_1 and/or X_2 borrow from c_3, or X_3 borrows from c_1 and c_2. The above inequalities guarantee that enough rates are always available for successful transmission. The details are relegated to the appendix. ■

IV. Networks with Multiple Sink Nodes

In general, the cutset condition given in Lemma 3 (see Appendix A) is not sufficient for networks with more than two sink nodes. That is, if condition (3) is satisfied for some rates $\{r_j\}$, it does not follow that these rates are achievable. We can demonstrate this with an example.

Example 2: A network with three sink nodes is shown in Figure 3. Recall that for three sink nodes, we can have a combination of seven different types of private and common messages, which are denoted by X_1 through X_7. Two messages X_3 and X_7 are generated at s with $r_3 = r_7 = 1$, and the objective is to transmit X_3 to t_3 and X_7 to all t_1, t_2 and t_3. It is easy to verify that the necessary condition given in Lemma 3 holds, i.e.,

$$\begin{aligned} \text{mincut}\{s, \{t_1\}\} &= 1 \geq r_7 = 1, \\ \text{mincut}\{s, \{t_2\}\} &= 1 \geq r_7 = 1, \\ \text{mincut}\{s, \{t_3\}\} &= 2 \geq r_3 + r_7 = 2, \\ \text{mincut}\{s, \{t_1, t_2\}\} &= 2 \geq r_7 = 1, \\ \text{mincut}\{s, \{t_1, t_3\}\} &= 2 \geq r_3 + r_7 = 2, \\ \text{mincut}\{s, \{t_2, t_3\}\} &= 2 \geq r_3 + r_7 = 2, \\ \text{mincut}\{s, \{t_1, t_2, t_3\}\} &= 2 \geq r_3 + r_7 = 2. \end{aligned}$$

However, the rates $r_3 = r_7 = 1$ in this example cannot be achieved: both t_1 and t_2 demand X_7 from s, which exhausts the capacity of the links from s to t_1 and t_2. Consequently, it is impossible to send X_3 to t_3. ■

In this section, we generalize the precoding method of the last section to networks with multiple receive nodes. Similar to the last section, we take $\mathbf{x}_j$ to be the j-th component of transmit data meant for subset w_j of receive nodes. Recall that each component $\mathbf{x}_j$ can be a private message, common to a subset of receive nodes, or common to all receive nodes. We concatenate all $\mathbf{x}_j$ into one large vector $\mathbf{x}$. Then,

$$\mathbf{c} = \mathbf{x}\mathbf{Q}$$

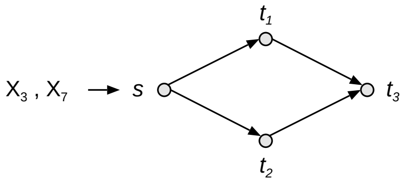

Fig. 3. Necessary condition of Lemma 3 is not always sufficient.

where $\mathbf{Q}$ is the precoding matrix. The precoding matrix must be chosen such that, at each receive node, all data meant for that node can be decoded. We collect all edge vectors at receive node t_i into one matrix $\mathbf{F}_i$, and all the receive signals at these edges into vector $\mathbf{y}_i$. Then,

$$\mathbf{y}_i = \mathbf{c}\mathbf{F}_i = \mathbf{x}\mathbf{Q}\mathbf{F}_i$$

We now segment the information vector $\mathbf{x}$ into $[\mathbf{z}_i \;\; \mathbf{z}_i']$, where $\mathbf{z}_i$ is relevant to node t_i and $\mathbf{z}_i'$ is the remaining data (some re-arrangement of elements may be necessary). We can correspondingly segment the precoding matrix $\mathbf{Q}$ into $[\mathbf{A}_i^T \;\; \mathbf{B}_i^T]^T$ (with the same re-arrangement of columns as for $\mathbf{x}$). Then,

$$\mathbf{y}_i = (\mathbf{z}_i\mathbf{A}_i + \mathbf{z}_i'\mathbf{B}_i)\mathbf{F}_i$$

We wish to recover $\mathbf{z}_i$ while $\mathbf{z}_i'$ essentially acts as interference and must be removed at node t_i. We now establish conditions on precoding matrix components $\mathbf{A}_i, \mathbf{B}_i$ to guarantee that this is possible.

Lemma 2: Consider $\mathbf{N}_i$ to be the projection matrix onto the null space of $\mathbf{B}_i\mathbf{F}_i$. Then the information intended for node t_i, namely $\mathbf{z}_i$, can be correctly recovered if, and only if, rank$(\mathbf{A}_i\mathbf{F}_i\mathbf{N}_i^T)$ is equal to the dimension of $\mathbf{z}_i$. $\blacksquare$

Proof: To show sufficiency, assume rank$(\mathbf{A}_i\mathbf{F}_i\mathbf{N}_i^T) =$ dim$(\mathbf{z}_i)$. Then, we can recover $\mathbf{z}_i$ from $\mathbf{y}_i$ in the following manner. Multiply $\mathbf{y}_i$ by $\mathbf{N}_i^T$ to find:

$$\begin{aligned}\mathbf{y}_i\mathbf{N}_i^T &= \mathbf{z}_i\mathbf{A}_i\mathbf{F}_i\mathbf{N}_i^T + \mathbf{z}_i'\mathbf{B}_i\mathbf{F}_i\mathbf{N}_i^T \\ &= \mathbf{z}_i\mathbf{A}_i\mathbf{F}_i\mathbf{N}_i^T\end{aligned}$$

Because of the rank condition, this system of equations can be solved to yield $\mathbf{z}_i$.

Conversely, assume[3] rank$(\mathbf{A}_i\mathbf{F}_i\mathbf{N}_i^T) <$ dim$(\mathbf{z}_i)$. Then there exists a nonzero vector $\mathbf{g}$ such that

$$\mathbf{g}\mathbf{A}_i\mathbf{F}_i\mathbf{N}_i^T = 0.$$

Hence the vector $\mathbf{g}\mathbf{A}_i\mathbf{F}_i$ is orthogonal to the null space of $\mathbf{B}_i\mathbf{F}_i$, i.e., it is in the row space of $\mathbf{B}_i\mathbf{F}_i$. This implies that there also exists a vector $\mathbf{g}'$ such that

$$\mathbf{g}\mathbf{A}_i\mathbf{F}_i = \mathbf{g}'\mathbf{B}_i\mathbf{F}_i.$$

It then follows that the information vector $[\mathbf{z}_i \;\; \mathbf{z}_i'] = [\mathbf{g} \;\; 0]$ cannot be distinguished from the information vector $[\mathbf{z}_i \;\; \mathbf{z}_i'] = [0 \;\; \mathbf{g}']$ at t_i, and perfect decoding is impossible. $\blacksquare$

Each of the receive nodes provides one set of constraints as seen above. The network code needs to satisfy all such conditions simultaneously.

Remark 2: Based on the above discussion, the proposed broadcasting scheme consists of the following four stages:

(1) *Construct a linear network code.* In this stage, a code vector is selected for each edge in the network according to Definition 1. In this paper, we will not discuss how to design a linear network code to optimize network broadcasting by assuming that the linear network code is either predefined or randomly generated. An optimal linear network code may be able to obtain a larger achievable rate region than a predefined or randomly generated linear network code, but it requires more centralized coordination and incurs the overhead of designing and deploying a new network code every time when the broadcast requirement is changed. In Section VI, a broadcasting scheme is proposed based on randomly generated linear network codes.

(2) *Precode the information messages using a precoding matrix* $\mathbf{Q}$. The information vector $\mathbf{x}$ is mapped to the data vector $\mathbf{c}$ by $\mathbf{c} = \mathbf{x}\mathbf{Q}$. The conditions on $\mathbf{Q}$ are given in Lemma 2, which can be used to construct algorithms for finding $\mathbf{Q}$.

(3) *Encode the data vector* $\mathbf{c}$ *using the linear network code.* For each link $e \in E$, the symbol transmitted along e is given by $\mathbf{c}\mathbf{f}_e$.

(4) *Decode via interference cancellation.* Use the methodology introduced in Lemma 2 to null the interference and then solve linearly for desired messages.

In the following, we give an example to illustrate how linear precoding can be used to achieve broadcasting.

Example 3: Consider the network in Figure 4, a modified butterfly network, where message X_4 with $r_4 = 1$ is intended for nodes t_1 and t_2, and message X_7 with $r_7 = 1$ is intended for nodes t_1, t_2 and t_3.

Assume that a generic linear network code over the network is given by

$$\begin{aligned}&\mathbf{f}_{e_1} = \mathbf{f}_{e_3} = \mathbf{f}_{e_5} = [1 \;\; 1]^T, \\ &\mathbf{f}_{e_2} = \mathbf{f}_{e_4} = \mathbf{f}_{e_7} = \mathbf{f}_{e_{10}} = [1 \;\; 2]^T, \\ &\mathbf{f}_{e_6} = \mathbf{f}_{e_8} = \mathbf{f}_{e_9} = [2 \;\; 3]^T.\end{aligned}$$

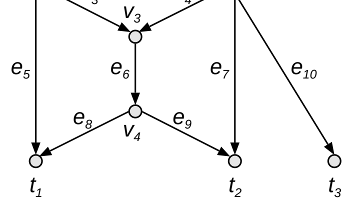

Fig. 4. A modified butterfly network.

[3] Because of the dimension of $\mathbf{A}_i$, rank$(\mathbf{A}_i\mathbf{F}_i\mathbf{N}_i^T)$ cannot be greater than dim$(\mathbf{z}_i)$.

Then,

$$\mathbf{F}_1 = [\mathbf{f}_{e_5} \;\; \mathbf{f}_{e_8}] = \begin{bmatrix} 1 & 2 \\ 1 & 3 \end{bmatrix},$$

$$\mathbf{F}_2 = [\mathbf{f}_{e_7} \;\; \mathbf{f}_{e_9}] = \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix},$$

$$\mathbf{F}_3 = \mathbf{f}_{e_{10}} = [1 \;\; 2]^T.$$

Let

$$\mathbf{x} = [x_4 \;\; x_7] \quad \text{and} \quad \mathbf{Q} = \begin{bmatrix} 2 & -1 \\ 1 & 2 \end{bmatrix}.$$

At t_1, we have

$$\mathbf{y}_1 = \mathbf{x}\mathbf{Q}\mathbf{F}_1 = \mathbf{x} \begin{bmatrix} 2 & -1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 1 & 3 \end{bmatrix} = \mathbf{x} \begin{bmatrix} 1 & 1 \\ 3 & 8 \end{bmatrix},$$

from which

$$\mathbf{x} = \mathbf{y}_1 \begin{bmatrix} 1 & 1 \\ 3 & 8 \end{bmatrix}^{-1} = \mathbf{y}_1 \begin{bmatrix} \frac{8}{5} & -\frac{1}{5} \\ -\frac{3}{5} & \frac{1}{5} \end{bmatrix}.$$

At t_2, we have

$$\mathbf{y}_2 = \mathbf{x}\mathbf{Q}\mathbf{F}_2 = \mathbf{x} \begin{bmatrix} 2 & -1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix} = \mathbf{x} \begin{bmatrix} 0 & 1 \\ 5 & 8 \end{bmatrix},$$

from which

$$\mathbf{x} = \mathbf{y}_2 \begin{bmatrix} 0 & 1 \\ 5 & 8 \end{bmatrix}^{-1} = \mathbf{y}_2 \begin{bmatrix} -\frac{8}{5} & \frac{1}{5} \\ 1 & 0 \end{bmatrix}.$$

At t_3, we have

$$y_3 = \mathbf{x}\mathbf{Q}\mathbf{F}_3 = \mathbf{x} \begin{bmatrix} 2 & -1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \mathbf{x} \begin{bmatrix} 0 \\ 5 \end{bmatrix} = 5x_7,$$

from which

$$x_7 = \frac{1}{5} y_3.$$

Therefore, symbol x_4 can be transmitted to t_1 and t_2, and symbol x_7 can be transmitted to t_1, t_2 and t_3. ■

V. Designing the Precoding Matrix $\mathbf{Q}$

In the last section, a technique based on linear precoding is presented for network broadcasting. However, how to construct the precoding matrix $\mathbf{Q}$ has not been addressed yet. In this section, we propose an efficient method for constructing $\mathbf{Q}$. We must note that this method, which is based on interference cancellation, is not guaranteed to arrive at the optimal $\mathbf{Q}$; however, it is computationally efficient and generates good results.

Consider the index set $\{1, \ldots, 2^\ell - 1\}$ representing the single-cast and multi-cast messages. The precoder is designed in a multi-step greedy fashion. We start with an empty precoder, and in each step add one of the single-cast or multi-cast components to the precoder.

During the execution of the algorithm, we start with the set $U = \{j : r_j > 0\}$ of single-cast and multi-cast messaging components with nonzero rates. We denote with $O = \{i : t_i \in w_j \text{ for some } j \in U\}$ the indices of receive nodes represented in U. As the algorithm progresses we take care of messaging components one-by-one, and thus U gets successively smaller and the algorithm stops when $U = \emptyset$.

We introduce the set U' for the message components that have already been included in the precoder. At the beginning, $U' = \emptyset$. Sometimes the rates requested of the algorithm cannot be supported, and therefore we also have a set V, initialized to $V = \emptyset$, as the components that cannot be transmitted at the requested rate.

Let

$$\mathbf{F}_i^{(0)} = \mathbf{F}_i, \;\; \forall i \in O.$$

Then, execute the following steps iteratively for $m = 0, 1, \ldots$, until U is empty:

Step 1) Randomly choose an index j from U. Then, r_j vectors $\mathbf{a}_{j,k}$, $k = 1, 2, \ldots, r_j$, are selected such that

a) $\text{rank}\{\mathbf{A}^T\mathbf{F}_i^{(m)}\} = r_j$ for all i such that $t_i \in w_j$;

b) $\max_{i \in O : t_i \notin w_j} \text{rank}\{\mathbf{A}^T\mathbf{F}_i\}$ is minimized;

where

$$\mathbf{A} = [\mathbf{a}_{j,1} \;\; \mathbf{a}_{j,2} \;\; \ldots \;\; \mathbf{a}_{j,r_j}].$$

Step 2)

```
IF Step 1 is successful AND Lemma 2 satisfied (messages can be recovered) THEN
    Go to Step 3.
ELSE
    $U \leftarrow U - \{j\}$
    $V \leftarrow V \cup \{j\}$
    $O \leftarrow \{$indices of receive nodes still participating in $U \cup U'\}$
    Go to Step 1.
ENDIF
```

Step 3) Recall that $\mathbf{A}$ denotes the precoder components in this iteration. Let $\mathbf{P}_F$ be the projection matrix onto the range space of $\mathbf{F}_i^{(m)}$, and $(\mathbf{P}_{\mathbf{P}_F\mathbf{A}})^\perp$ be the projection matrix onto the orthogonal complement of the range space of $\mathbf{P}_F\mathbf{A}$. Then

$$\mathbf{F}_i^{(m+1)} \leftarrow (\mathbf{P}_{\mathbf{P}_F\mathbf{A}})^\perp \mathbf{F}_i^{(m)} \qquad \forall i \in O$$

$$U \leftarrow U - \{j\}$$

$$U' \leftarrow U' \cup \{j\}$$

$$m \leftarrow m + 1 \;\; \text{(increment the algorithm counter)}$$

Remark 3:

(1) The vectors $\mathbf{a}_{j,k}$ are used to construct the precoding matrix $\mathbf{Q}$.

(2) In *Step 1*, Criterion a) ensures that message j can be decoded by all $t_i \in w_j$, while Criterion b) tries to minimize the "interference" caused by message j.

(3) In *Step 3*, $\mathbf{F}_i^{(m+1)}$ represents the unused degree of freedom that can be exploited to encode other message components.

(4) The set O labels all the sink nodes that intend to receive at least one message from the source node. It is updated in each iteration to exclude the sink nodes that do not receive any messages in the whole encoding process.

Assume that the precoding vectors have been obtained by the proposed algorithm for $X_{j(1)}$, $X_{j(2)}$, …, $X_{j(M)}$, i.e., the final $U' = \{j(1), j(2), \ldots, j(M)\}$. Let

$$\mathbf{x} = [\mathbf{x}_{j(1)} \;\; \mathbf{x}_{j(2)} \;\; \ldots \;\; \mathbf{x}_{j(M)}],$$

where $\mathbf{x}_j = [x_{j,1}\ x_{j,2}\ \ldots\ x_{j,r_j}]$. The associated precoding matrix $\mathbf{Q}$ is given by

$$\mathbf{Q} = [\mathbf{Q}_{j(1)}\ \ \mathbf{Q}_{j(2)}\ \ \ldots\ \ \mathbf{Q}_{j(M)}]^T,$$

where $\mathbf{Q}_j = [\mathbf{a}_{j,1}\ \mathbf{a}_{j,2}\ \ldots\ \mathbf{a}_{j,r_j}]$. As verified by Lemma 2 in *Step 2*, the construction of $\mathbf{Q}$ guarantees that the messages X_j can be decoded by all $t_i \in w_j$ for all $j = j(1), j(2), \ldots, j(M)$.

A. Construction of $\mathbf{Q}$

Example 4: Consider the broadcast network in Figure 5. We use the proposed algorithm to construct a precoding matrix for this network. Two messages X_{14} and X_{15} are required to be broadcast to the sink nodes $\{t_2, t_3, t_4\}$ and $\{t_1, t_2, t_3, t_4\}$, respectively. The entropy rates are $r_{14} = 1$ and $r_{15} = 1$. A linear network code on the network is given by

$$\begin{aligned}
&\mathbf{f}_{e_1} = \mathbf{f}_{e_4} = \mathbf{f}_{e_5} = \mathbf{f}_{e_6} = [1\ \ 0\ \ 0]^T,\\
&\mathbf{f}_{e_2} = \mathbf{f}_{e_7} = \mathbf{f}_{e_8} = [0\ \ 1\ \ 0]^T,\\
&\mathbf{f}_{e_3} = \mathbf{f}_{e_9} = \mathbf{f}_{e_{10}} = [0\ \ 0\ \ 1]^T.
\end{aligned}$$

Hence,

$$\begin{aligned}
&\mathbf{F}_1 = \mathbf{f}_{e_4} = [1\ \ 0\ \ 0]^T,\\
&\mathbf{F}_2 = [\mathbf{f}_{e_5}\ \ \mathbf{f}_{e_7}] = \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}^T,\\
&\mathbf{F}_3 = [\mathbf{f}_{e_6}\ \ \mathbf{f}_{e_9}] = \begin{bmatrix}1 & 0 & 0\\ 0 & 0 & 1\end{bmatrix}^T,\\
&\mathbf{F}_4 = [\mathbf{f}_{e_8}\ \ \mathbf{f}_{e_{10}}] = \begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}^T.
\end{aligned}$$

To begin with, we perform the following initialization:

$$U = \{14, 15\},\ U' = \emptyset,\ V = \emptyset,\ O = \{1, 2, 3, 4\},$$

and

$$\begin{aligned}
&\mathbf{F}_1^{(0)} = \mathbf{F}_1 = [1\ \ 0\ \ 0]^T,\\
&\mathbf{F}_2^{(0)} = \mathbf{F}_2 = \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}^T,\\
&\mathbf{F}_3^{(0)} = \mathbf{F}_3 = \begin{bmatrix}1 & 0 & 0\\ 0 & 0 & 1\end{bmatrix}^T,\\
&\mathbf{F}_4^{(0)} = \mathbf{F}_4 = \begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}^T.
\end{aligned}$$

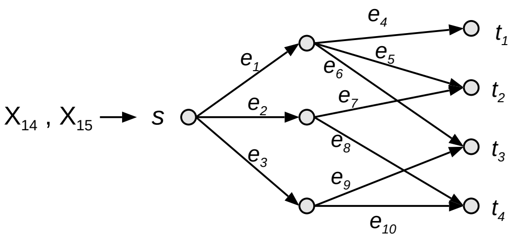

Fig. 5. A broadcast network with four receive nodes.

Starting with $j = 14$, we choose

$$\mathbf{a}_{14,1} = [0\ \ 1\ \ 1]^T,$$

because

$$\begin{aligned}
&\text{rank}\big(\mathbf{a}_{14,1}^T\mathbf{F}_2^{(0)}\big) = 1,\ \ \text{rank}\big(\mathbf{a}_{14,1}^T\mathbf{F}_3^{(0)}\big) = 1,\\
&\text{rank}\big(\mathbf{a}_{14,1}^T\mathbf{F}_4^{(0)}\big) = 1,
\end{aligned}$$

and

$$\text{rank}\big(\mathbf{a}_{14,1}^T\mathbf{F}_1\big) = 0.$$

Step 3 obtains

$$\begin{aligned}
&\mathbf{F}_1^{(1)} = [1\ \ 0\ \ 0]^T,\\
&\mathbf{F}_2^{(1)} = [1\ \ 0\ \ 0]^T,\\
&\mathbf{F}_3^{(1)} = [1\ \ 0\ \ 0]^T,\\
&\mathbf{F}_4^{(1)} = [0\ \ 1\ \ -1]^T,
\end{aligned}$$

and

$$U = \{15\},\ U' = \{14\},\ V = \emptyset,\ O = \{1, 2, 3, 4\}.$$

Let $j = 15$, and we choose

$$\mathbf{a}_{15,1} = [1\ \ 1\ \ 0]^T,$$

because

$$\begin{aligned}
&\text{rank}\big(\mathbf{a}_{15,1}^T\mathbf{F}_1^{(1)}\big) = 1,\ \text{rank}\big(\mathbf{a}_{15,1}^T\mathbf{F}_2^{(1)}\big) = 1,\\
&\text{rank}\big(\mathbf{a}_{15,1}^T\mathbf{F}_3^{(1)}\big) = 1,\ \text{rank}\big(\mathbf{a}_{15,1}^T\mathbf{F}_4^{(1)}\big) = 1.
\end{aligned}$$

Then, *Step 3* gives

$$U = \emptyset,\ U' = \{14, 15\},\ V = \emptyset,\ \ O = \{1, 2, 3, 4\}.$$

Let

$$\mathbf{x} = [x_{14,1}\ \ x_{15,1}],$$

and

$$\mathbf{Q} = [\mathbf{a}_{14,1}\ \ \mathbf{a}_{15,1}]^T = \begin{bmatrix}0 & 1 & 1\\ 1 & 1 & 0\end{bmatrix}.$$

At sink node t_1,

$$\begin{aligned}
y_1 = \mathbf{x}\mathbf{Q}\mathbf{F}_1 &= x_{14,1}\mathbf{a}_{14,1}^T\mathbf{F}_1 + x_{15,1}\mathbf{a}_{15,1}^T\mathbf{F}_1\\
&= x_{15,1}\mathbf{a}_{15,1}^T\mathbf{F}_1 = x_{15,1},
\end{aligned}$$

and hence

$$x_{15,1} = y_1.$$

At sink node t_2,

$$\begin{aligned}
\mathbf{y}_2 = \mathbf{x}\mathbf{Q}\mathbf{F}_2 &= [x_{14,1}\ \ x_{15,1}]\begin{bmatrix}0 & 1 & 1\\ 1 & 1 & 0\end{bmatrix}\begin{bmatrix}1 & 0\\ 0 & 1\\ 0 & 0\end{bmatrix}\\
&= [x_{14,1}\ \ x_{15,1}]\begin{bmatrix}0 & 1\\ 1 & 1\end{bmatrix},
\end{aligned}$$

and hence

$$[x_{14,1}\ \ x_{15,1}] = \mathbf{y}_2\begin{bmatrix}-1 & 1\\ 1 & 0\end{bmatrix}.$$

At sink node t_3,

$$\begin{aligned}\mathbf{y}_3 = \mathbf{x}\mathbf{Q}\mathbf{F}_3 &= [x_{14,1} \;\; x_{15,1}] \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \\ &= [x_{14,1} \;\; x_{15,1}] \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},\end{aligned}$$

and hence

$$[x_{14,1} \;\; x_{15,1}] = \mathbf{y}_3 \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

At sink node t_4,

$$\begin{aligned}\mathbf{y}_4 = \mathbf{x}\mathbf{Q}\mathbf{F}_4 &= [x_{14,1} \;\; x_{15,1}] \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \\ &= [x_{14,1} \;\; x_{15,1}] \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix},\end{aligned}$$

and hence

$$[x_{14,1} \;\; x_{15,1}] = \mathbf{y}_4 \begin{bmatrix} 0 & 1 \\ 1 & -1 \end{bmatrix}.$$

Therefore, the broadcast requirement can be achieved. The symbol transmitted along each edge is given by

$$\begin{aligned}&e_1 : \mathbf{x}\mathbf{Q}\mathbf{f}_{e_1} = x_{15,1},\; e_4 : \mathbf{x}\mathbf{Q}\mathbf{f}_{e_4} = x_{15,1},\; e_5 : \mathbf{x}\mathbf{Q}\mathbf{f}_{e_5} = x_{15,1}, \\ &e_6 : \mathbf{x}\mathbf{Q}\mathbf{f}_{e_6} = x_{15,1},\; e_2 : \mathbf{x}\mathbf{Q}\mathbf{f}_{e_2} = x_{14,1} + x_{15,1}, \\ &e_7 : \mathbf{x}\mathbf{Q}\mathbf{f}_{e_7} = x_{14,1} + x_{15,1},\; e_8 : \mathbf{x}\mathbf{Q}\mathbf{f}_{e_8} = x_{14,1} + x_{15,1}, \\ &e_3 : \mathbf{x}\mathbf{Q}\mathbf{f}_{e_3} = x_{14,1},\; e_9 : \mathbf{x}\mathbf{Q}\mathbf{f}_{e_9} = x_{14,1}, \\ &e_{10} : \mathbf{x}\mathbf{Q}\mathbf{f}_{e_{10}} = x_{14,1}.\end{aligned}$$

■

B. *Precoding for a Given Achievable Rate Vector*

The rate vector for a network may lie anywhere in the achievable region. In general, a separate code needs to be designed for each of the rate vectors. One of the advantages of the precoding approach is that we can fix the network code, and only vary the precoder to arrive at various points in the achievable rate region. We demonstrate this via an example.

Example 5: Consider the network shown in Figure 6. A predefined linear network code is given by

$$\begin{aligned}&\mathbf{f}_{e_1} = \mathbf{f}_{e_2} = [1 \;\; 0]^T, \\ &\mathbf{f}_{e_3} = \mathbf{f}_{e_4} = [0 \;\; 1]^T, \\ &\mathbf{f}_{e_5} = [1 \;\; 1]^T.\end{aligned}$$

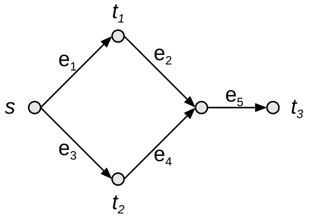

Fig. 6. A 5-node network. We demonstrate various achievable rates on this network via precoding matrices.

Hence,

$$\mathbf{F}_1 = \mathbf{f}_{e_1},\; \mathbf{F}_2 = \mathbf{f}_{e_3},\; \mathbf{F}_3 = \mathbf{f}_{e_5}.$$

Different broadcast objectives can be achieved as follows:

(1) If $\mathbf{x} = [x_{1,1} \; x_{2,1}]$, let

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Then, $x_{1,1}$ can be transmitted to t_1 and $x_{2,1}$ can be transmitted to t_2.

(2) If $\mathbf{x} = [x_{1,1} \; x_{3,1}]$, let

$$\mathbf{Q} = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}.$$

Then, $x_{1,1}$ can be transmitted to t_1 and $x_{3,1}$ can be transmitted to t_3.

(3) If $\mathbf{x} = [x_{2,1} \; x_{3,1}]$, let

$$\mathbf{Q} = \begin{bmatrix} -1 & 1 \\ 1 & 0 \end{bmatrix}.$$

Then, $x_{2,1}$ can be transmitted to t_2 and $x_{3,1}$ can be transmitted to t_3.

■

VI. Completing the Circle: Precoding in Random Networks

So far, we have concentrated on the design of precoders for a given network code. However, we have not addressed the question of designing the network code itself (namely, the edge vectors). In this section we outline the methodology with which a complete network code may be designed, although the details are outside the scope of this paper and are a subject for future work. There are many works in the literature that concern themselves with the design of linear network codes, and the issue is in general not a simple one. However, it has been recently discovered that randomly selected network codes can asymptotically achieve the multicast capacity [6].

Motivated by this result, we propose to use precoding with random network codes in a straightforward manner. We propose that for each instance of a random network code, an appropriate precoding matrix can be constructed. Then, data will be transmitted using the precoder, and will be decoded at each receive node according to the precoder structure. This requires a certain communication overhead: the signal subspaces at destination nodes must be known for precoder design, and the precoder must be known at the source and destination nodes. One may argue that if enough codewords are transmitted at each random realization of the network code, then the overhead rate may be made arbitrarily small. The details of the communication process for the overhead is the subject of system design, which is beyond the scope of this paper.

VII. Conclusions

In the paper, the network broadcast problem is studied in the context of linear network codes. A systematic way based on linear precoding is proposed to achieve broadcasting, and the criterion for choosing an appropriate precoding matrix is derived. The scheme can be used with random linear network codes to perform broadcasting, and different broadcast requirements can be achieved by simply changing the precoding matrix and its associated decoding scheme. Nevertheless, how to characterize the achievable rate region for a network with multiple sinks and how to generalize the work to nonlinear codes are still open research problems.

Acknowledgment

This work was supported in part by NSF grants ECS-0601266 and CNS-0435429.

Appendix

A. A Necessary Condition for Broadcast Achievability

In this appendix, we use the cutset bound to deduce a necessary condition for achievable rates under broadcast conditions. To do so, for each subset of sink nodes, we generate a dummy node absorbing all their information. We then apply the cutset bound on that dummy node.

Lemma 3: Consider a network with a single source node s and l sink nodes t_i, $i = 1, 2, \ldots, l$. If rates r_j, $j = 1, 2, \ldots, 2^l - 1$, are achievable, then the network satisfies

$$\text{mincut}\{s,\, w_j\} \geq \sum_{w_k \cap w_j \neq \emptyset} r_k \tag{3}$$

■

Proof: Since the rates are achievable, there exists a coding scheme such that each node t_i can receive or deduce its desired message sets $\{X_k : t_i \in w_k\}$. Consider the subset of sink nodes w_j. We construct a new network G' by adding a new node z and connecting every $t_i \in w_j$ to z with infinite-capacity edges. Then, by extending the original coding scheme, z can receive the message set $\{X_k : w_k \cap w_j \neq \emptyset\}$. Information transmission from s to z is essentially a single-source-node single-sink-node problem. The Max-flow Min-cut theorem for single-source networks implies [3]:

$$\text{mincut}\,\{s,\, \{z\}\} \geq \sum_{w_k \cap w_j \neq \emptyset} r_k.$$

Since each newly added edge has an infinite capacity, we have

$$\text{mincut}\,\{s,\, \{z\}\} = \text{mincut}\,\{s,\, w_j\}\,,$$

which leads to (3). ■

In a network with exactly two sink nodes, the above condition is both necessary and sufficient, and the rates can be achieved by linear network codes (see Section III). With more than two sink nodes; however, the condition is not sufficient (see Section IV).

B. Completion of the Proof of Lemma 1

Proof: [Lemma 1] Assume the following matrix is not full rank, i.e.,

$$\text{rank}\left\{ \begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_3^T \end{bmatrix} \mathbf{F}_1 \right\} < c_1 + c_3.$$

Then there exists a nonzero vector $\mathbf{z}$ such that

$$\mathbf{z}^T \begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_3^T \end{bmatrix} \mathbf{F}_1 = [\mathbf{z}_1^T \;\; \mathbf{z}_3^T] \begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_3^T \end{bmatrix} \mathbf{F}_1 = 0.$$

Since P_1 is the range space of $\mathbf{F}_1$, the vector $\mathbf{Q}_1\mathbf{z}_1 + \mathbf{Q}_3\mathbf{z}_3$ is orthogonal to P_1 and thus in the subspace spanned by the $\{\mathbf{a}_{2,k}\}$. This implies that the vectors $\mathbf{a}_{1,k}$, $\mathbf{a}_{2,k}$ and $\mathbf{a}_{3,k}$ are linearly dependent, i.e., there exists a vector $\mathbf{z}_2$ such that

$$\mathbf{Q}_1\mathbf{z}_1 + \mathbf{Q}_2\mathbf{z}_2 + \mathbf{Q}_3\mathbf{z}_3 = 0.$$

But the range spaces of $\mathbf{Q}_1$ and $\mathbf{Q}_2$ are orthogonal to the range space of $\mathbf{Q}_3$. Hence,

$$\mathbf{Q}_1\mathbf{z}_1 + \mathbf{Q}_2\mathbf{z}_2 = 0, \quad \mathbf{Q}_3\mathbf{z}_3 = 0,$$

which implies $\mathbf{z}_3 = 0$ because $\mathbf{Q}_3$ is full rank. From $\mathbf{Q}_1\mathbf{z}_1 + \mathbf{Q}_2\mathbf{z}_2 = 0$, we have $\mathbf{Q}_1\mathbf{z}_1 = -\mathbf{Q}_2\mathbf{z}_2$. Since $\mathbf{Q}_2\mathbf{z}_2$ is orthogonal to P_1, then $\mathbf{Q}_1\mathbf{z}_1$ is orthogonal to both P_1 and P_2. However, $\mathbf{Q}_1\mathbf{z}_1 \in \text{span}\{P_1 \cup P_2\}$. This implies that $\mathbf{z}_1 = 0$ and $\mathbf{z}_2 = 0$ because both $\mathbf{Q}_1$ and $\mathbf{Q}_2$ are full rank. This shows that the vectors $\mathbf{a}_{1,k}$, $\mathbf{a}_{2,k}$ and $\mathbf{a}_{3,k}$ cannot be linearly dependent, which shows that $\begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_3^T \end{bmatrix} \mathbf{F}_1$ is full rank.

■

C. Proof of Theorem 2

Proof: [Theorem 2] The details of rate arithmetic for this theorem are as follows:

(1) If $c_3 \leq r_3$, X_3 is split into two independent parts $X_{3,1}$ and $X_{3,2}$ with rates c_3 and $r_3 - c_3$, respectively. The private rates to t_1 are used by r_1 data symbols from X_1 and $r_3 - c_3$ data symbols from $X_{3,2}$, the private rates to t_2 are used by r_2 data symbols from X_2 and $r_3 - c_3$ data symbols from $X_{3,2}$, and the common rates to t_1 and t_2 are used by c_3 data symbols from $X_{3,1}$.

(2) If $c_3 > r_3$, the private rates to t_1 are used by $\min\{r_1,\, c_1\}$ data symbols from X_1, the private rates to t_2 are used by $\min\{r_2,\, c_2\}$ data symbols from X_2, and the common rates to t_1 and t_2 are used by the rest $r_1 - \min\{r_1,\, c_1\}$ data symbols from X_1, the rest $r_2 - \min\{r_2,\, c_2\}$ data symbols from X_2, and r_3 data symbols from X_3.

■

References

[1] R. W. Yeung, *A First Course in Information Theory*, New York: Springer, 2002.

[2] R. W. Yeung, S.-Y. R. Li, N. Cai, and Z. Zhang, "Theory of network coding," submitted to *Foundation and Trends in Communications and Information Theory*. Available online at: http://iest2.ie.cuhk.edu.hk/~whyeung/publications/preprint.html.

[3] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Information Theory*, vol. 46, pp. 304–316, Jul. 2000.

[4] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Trans. Information Theory*, vol. 49, pp. 371–381, Feb. 2003.

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

[6] T. Ho, M. Médard, R. Koetter, D. R. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," submitted to *IEEE Trans. Information Theory*. Available online at: `http://www.its.caltech.edu/~tho/itrandom-revision.pdf`.

[7] E. Erez and M. Feder, "Capacity region and network codes for two receivers multicast with private and common data," in *Proc. Workshop on Coding, Cryptography and Combinatorics*, China, 2003.

[8] C. K. Ngai and R. W. Yeung, "Multisource network coding with two sinks," in *Proc. International Conference on Communications, Circuits and Systems (ICCCAS)*, vol. 1, pp. 34–37, Jun. 2004.

[9] A. Ramamoorthy and R. D. Wesel, "The single source two terminal network with network coding," in *Proc. 9th Canadian Workshop on Information Theory*, Montreal, Jun. 2005.