Network Coding from a Network Flow Perspective

Tracey Ho, David R. Karger, Muriel Médard and Ralf Koetter

***Abstract*— The algebraic framework introduced in [4] gives an algebraic condition for the feasibility of a set of multicast connections in a network, that is equivalent to the max-flow min-cut condition of [1]. The algebraic condition also checks the validity of a linear coding solution to a given multicast connection problem. We present two alternative formulations of this condition that are closely tied to network flow, providing further insights and connections with combinatorial optimization. They are also easier to work with in many cases as they do not involve matrix products and inversions. From these results we derive a substantially tighter upper bound on the coding field size required for a given connection problem than that in [4].**

I. Introduction

The algebraic framework introduced in [4] gives an algebraic condition, in terms of transfer matrix determinant polynomials, for checking the feasibility of a set of multicast connections in a network, that is equivalent to the max-flow min-cut condition of [1]. The algebraic condition also checks the validity of a linear coding solution to a given multicast connection problem.

We present two alternative formulations of the condition that are related to network flow. The first is in terms of network flows to individual receivers. The second is in terms of the Edmonds matrix formulation for checking if a bipartite graph has a perfect matching, which is a classical reduction of the problem of checking the feasibility of an $s - t$ flow [3]. This latter problem is a special case of network coding, restricted to the binary field and to transmission of only one input signal on any output link. It is thus interesting to find that the two formulations are equivalent for the more general case of coding in higher order fields.

These combinatorial formulations make it easier to deduce various characteristics of transfer matrix determinant polynomials, without involving matrix products and inversions. Such characteristics include the maximum exponent of a variable, the total degree of the polynomial, and its form for networks with linearly correlated sources, or with some fixed and some randomized code coefficients. These lead to new results on randomized distributed transmission and compression of information in networks, presented in our companion paper [2], and an upper bound on required coding field size that substantially tightens the bound given in [4].

The paper is organized as follows: Section II describes our network model, Section III gives the main results, Section IV gives proofs and ancillary results, and Section V concludes the paper with a summary of the results and a discussion of further work.

Tracey Ho and Muriel Médard are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139, e-mail: {trace, medard}@mit.edu

David R. Karger is with the Laboratory for Computer Science, Massachusetts Institute of Technology, MA 02139, e-mail: karger@lcs.mit.edu

Ralf Koetter is with the Coordinated Science Laboratory, University of Illinois, Urbana, IL 61801, e-mail: koetter@csl.uiuc.edu

II. Model

We adopt the model of [4], which represents a network as a directed graph $\mathcal{G}$. Discrete independent random processes $X_1, \ldots, X_r$ are observable at one or more source nodes, and there are $d \geq 1$ receiver nodes. The output processes at a receiver node β are denoted $Z(\beta, i)$. The *multicast* connection problem is to transmit all the source processes to each of the receiver nodes.

There are ν links in the network. Link l is an *incident outgoing link* of node v if $v = \text{tail}(l)$, and an *incident incoming link* of v if $v = \text{head}(l)$. We call an incident outgoing link of a source node a *source link* and an incident incoming link of a receiver node a *terminal link*. Edge l carries the random process $Y(l)$.

The time unit is chosen such that the capacity of each link is one bit per unit time, and the random processes X_i have a constant entropy rate of one bit per unit time. Edges with larger capacities are modelled as parallel edges, and sources of larger entropy rate are modelled as multiple sources at the same node.

The processes X_i, $Y(l)$, $Z(\beta, i)$ generate binary sequences. We assume that information is transmitted as vectors of bits which are of equal length u, represented as elements in the finite field $\mathbb{F}_{2^u}$. The length of the vectors is equal in all transmissions and all links are assumed to be synchronized with respect to the symbol timing.

In this paper we consider linear coding[1] on acyclic

[1]which is sufficient for multicast [5]

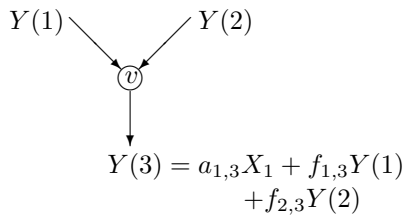

Fig. 1. Illustration of linear coding at a node.

delay-free networks[2]. In a linear code, the signal $Y(j)$ on a link j is a linear combination of processes X_i generated at node $v = \text{tail}(j)$ and signals $Y(l)$ on incident incoming links l (ref Figure 1):

$$Y(j) = \sum_{\{i\,:\,X_i \text{ generated at } v\}} a_{i,j}X_i + \sum_{\{l\ :\ \text{head}(l)\,=\,v\}} f_{l,j}Y(l)$$

and an output process $Z(\beta, i)$ at receiver node β is a linear combination of signals on its terminal links:

$$Z(\beta, i) = \sum_{\{l\,:\,\text{head}(l)=\beta\}} b_{\beta_{i,l}}Y(l)$$

The coefficients $\{a_{i,j}, f_{l,j}, b_{\beta_{i,l}} \in \mathbb{F}_{2^u}\}$ can be collected into matrices $r \times \nu$ matrices $A = (a_{i,j})$ and $B_\beta = (b_{\beta_{i,l}})$, and the $\nu \times \nu$ matrix $F = (f_{l,j})$, whose structure is constrained by the network. A triple (A, F, B), where

$$B = \left[\begin{array}{c} B_1 \\ \hline : \\ \hline B_d \end{array}\right]$$

specifies the behavior of the network, and represents a *linear network code*.

We use the following notation:

- $G = (I - F)^{-1}$
- $G_{\mathcal{H}}$ is the submatrix consisting of columns of G corresponding to links in set $\mathcal{H}$
- $\underline{a}_j$ is column j of A
- $\underline{c}_j$ is column j of AG
- $\underline{b}_j$ is column j of B

We denote a path traversing links $\{e_1, \ldots, e_k\}$ by the ordered set $\{e_1, \ldots, e_k\}$, where $e_1 < \ldots < e_k$. For any set $\mathcal{E} = \{e_1, \ldots, e_k\}$, $e_1 < \ldots < e_k$, we define

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

which for a path $\mathcal{E}$ is the product of gains along the path.

[2]this algebraic framework can be extended to networks with cycles and delay by working in fields of rational functions in a delay variable [4]

III. Main results

Reference [4] gives the following necessary and sufficient condition for a multicast connection problem to be feasible (or for a particular network code (A, F, B) to be a valid solution): that for each receiver β, the transfer matrix $A(I - F)^{-1}B_\beta^T = AGB_\beta^T$ has nonzero determinant.

Our first result is an alternative statement of this condition in terms of flows.

Theorem 1: A multicast connection problem is feasible (or a particular (A, F) can be part of a valid solution) if and only if each receiver β has a set $\mathcal{H}_\beta$ of r incident incoming links for which

$$P_{\mathcal{H}_\beta} = \sum_{\substack{\{\text{disjoint paths } \mathcal{E}_1, \ldots, \mathcal{E}_r\,:\\ \mathcal{E}_i \text{ from outgoing link}\\ l_i \text{ of source } i \text{ to } h_i \in \mathcal{H}_\beta\}}} \left|A_{\{l_1,\ldots,l_r\}}\right| \prod_{j=1}^{r} g(\mathcal{E}_j) \neq 0$$

where $A_{\{l_1,\ldots,l_r\}}$ is the submatrix of A consisting of columns corresponding to links $\{l_1, \ldots, l_r\}$. The sum is over all flows that transmit all source processes to links in $\mathcal{H}_\beta$, each flow being a set of r disjoint paths each connecting a different source to a different link in $\mathcal{H}_\beta$. □

This formulation in terms of network flows is useful in proving our second result: the equivalence of the transfer matrix formulation of [4] and the Edmonds matrix formulation for checking if a bipartite graph has a perfect matching.

Theorem 2: The determinant of the transfer matrix $M_1 = A(I - F)^{-1}B_\beta^T$ for receiver β in a network code (A, F, B) can be calculated as

$$|M_1| = (-1)^{r(\nu+1)}\,|M_2|$$

where $M_2 = \begin{bmatrix} A & 0 \\ I - F & B_\beta^T \end{bmatrix}$ is the corresponding Edmonds matrix. □

These combinatorial formulations lead to new results presented in [2], and to the following bound on the size of the finite field needed for coding, which determines the complexity of a linear multicast code.

Theorem 3 (Required field size): For a feasible multicast connection problem with d receivers, there exists a solution in a finite field $\mathbb{F}_q$ where $q > d$, or in a finite field $\mathbb{F}_{2^u}$ where $u \leq \lceil \log_2(d + 1) \rceil$. □

This result tightens the upper bound of $u \leq \lceil \log_2(rd + 1) \rceil$ given in [4], where r is the number of processes being transmitted in the network.

IV. Proofs and ancillary results

We prove the following slightly more general form of Theorem 1, since this is useful for proving Theorem 2.

Lemma 1: Let A be an arbitrary $r \times \nu$ matrix and F an arbitrary upper triangular $\nu \times \nu$ matrix with zeros on the main diagonal. For $1 \leq h' \leq h \leq \nu$, let $S_{h',h}$ be the set of all sets of integers $\{e_1, e_2, \ldots, e_k\}$ such that $h' = e_1 < e_2 < \ldots < e_k = h$. Let $\mathcal{H} = \{h_1, \ldots, h_r\}$, where $1 \leq h_1 < \ldots < h_r \leq \nu$. Then $|AG_{\mathcal{H}}| =$

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

Proof: It follows from the definitions of transfer matrices A and $G = I + F + F^2 + \ldots$ that $\underline{c}_h$ can be computed recursively as follows:

$$\underline{c}_1 = \underline{a}_1 \quad (1)$$

$$\underline{c}_h = \sum_{i=1}^{h-1} \underline{c}_i f_{i,h} + \underline{a}_h, \quad h = 2, 3, \ldots, \nu \quad (2)$$

Using the expression

$$\underline{c}_h = \sum_{i=1}^{h} \underline{a}_i \sum_{\mathcal{E}\in S_{i,h}} g(\mathcal{E})$$

for each column of $AG_{\mathcal{H}}$ and expanding the determinant linearly in all columns, we obtain

$$|AG_{\mathcal{H}}| = \left| \begin{array}{ccc} | & & | \\ \underline{c}_{h_1} & \cdots & \underline{c}_{h_r} \\ | & & | \end{array} \right|$$

$$= \sum_{\substack{\{(h'_1,\ldots,h'_r):\\ 1\leq h'_j\leq h_j\\ h'_i\neq h'_j\ \forall\, i\neq j\}}} \left| \begin{array}{ccc} | & & | \\ \underline{a}_{h'_1} & \cdots & \underline{a}_{h'_r} \\ | & & | \end{array} \right| \prod_{i=1}^{r} \sum_{\mathcal{E}\in S_{h'_i,h_i}} g(\mathcal{E})$$

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

The above expansion does not take into account dependencies among the columns $\underline{c}_h$. We can obtain an equivalent expression with fewer terms by using the following alternative sequence of expansions which takes the dependencies into account. We start by expanding the determinant of $AG_{\mathcal{H}}$ linearly in the $h_r{}^{th}$ column using Equation 2:

$$|AG_{\mathcal{H}}| = \left| \begin{array}{ccc} | & & | \\ \underline{c}_{h_1} & \cdots & \underline{c}_{h_r} \\ | & & | \end{array} \right|$$

$$= \sum_{\substack{\{i\ :\ 1\leq i<h_r,\\ i\neq h_1,\ldots,h_{r-1}\}}} \left| \begin{array}{ccccc} | & & | & & | \\ \underline{c}_{h_1} & \cdots & \underline{c}_{h_{r-1}} & & \underline{c}_i \\ | & & | & & | \end{array} \right| f_{i,h_r} + \left| \begin{array}{ccccc} | & & | & & | \\ \underline{c}_{h_1} & \cdots & \underline{c}_{h_{r-1}} & & \underline{a}_{h_r} \\ | & & | & & | \end{array} \right|$$

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

Proof of Theorem 1: The result follows from Lemma 1 by noting that each set $\mathcal{E} = \{e_1, e_2, \ldots, e_k\}$ such that $g(\mathcal{E}) \neq 0$ corresponds to a network path consisting of links $e_1, \ldots, e_k$; that the condition $\mathcal{E}_j \cap \mathcal{E}_k = \emptyset$ for all $j \neq k,\ 1 \leq j, k \leq r$ implies that the corresponding paths $\mathcal{E}_1, \ldots, \mathcal{E}_r$ are disjoint; and that $\left|\underline{a}_{h'_1} \ldots \underline{a}_{h'_r}\right|$ is nonzero only when links $h_{j'}$ are source links of r different sources and carry r independent signals. ■

Proof of Theorem 2: We prove that this result holds for any set of matrices (A, F, B)[3] where A and B are arbitrary $r \times \nu$ matrices, and F is an arbitrary upper triangular $\nu \times \nu$ matrix with zeros on the main diagonal.

$$|M_1| = \left| \left[\begin{array}{ccc} | & & | \\ \underline{c}_1 & \cdots & \underline{c}_\nu \\ | & & | \end{array} \right] \left[\begin{array}{c} -\underline{b}_1^T- \\ \vdots \\ -\underline{b}_\nu^T- \end{array} \right] \right|$$

$$= \left| \begin{array}{ccc} & | & \\ \sum_{i=1}^{\nu} \underline{c}_i b_{1,i} & \cdots & \sum_{i=1}^{\nu} \underline{c}_i b_{r,i} \\ & | & \end{array} \right|$$

[3]We drop the subscript β from B_β for notational simplicity.

Expanding $|M_1|$ linearly in each column and using the determinant expansion

$$\det M = \sum_{\text{all permutations } p} (\text{sign } p) M(p(1),1) \dots M(p(r),r)$$

we have

$$\begin{aligned}
&|M_1| \\
&= \sum_{\substack{\{(h_1,\dots,h_r):\\ 1\le h_1<h_2\\ \dots<h_r\le\nu\}}} \sum_{\text{all permutations } p} \left| \begin{array}{ccc} | & & | \\ \underline{c}_{h_{p(1)}} & \dots & \underline{c}_{h_{p(r)}} \\ | & & | \end{array} b_{1,h_{p(1)}} \dots b_{r,h_{p(r)}} \right| \\
&= \sum_{\substack{\{(h_1,\dots,h_r):\\ 1\le h_1<h_2\\ \dots<h_r\le\nu\}}} \sum_{\text{all permutations } p} (\text{sign } p^{-1}) \\
&\qquad \left| \begin{array}{ccc} | & & | \\ \underline{c}_{h_1} & \dots & \underline{c}_{h_r} \\ | & & | \end{array} b_{p^{-1}(1),h_1} \dots b_{p^{-1}(r),h_r} \right| \\
&= \sum_{\substack{\{(h_1,\dots,h_r):\\ 1\le h_1<h_2\\ \dots<h_r\le\nu\}}} \sum_{\text{all permutations } p} (\text{sign } p^{-1}) \\
&\qquad \sum_{\text{all permutations } \tilde{p}} (\text{sign } \tilde{p}) c_{\tilde{p}(1),h_1} b_{p^{-1}(1),h_1} \dots c_{\tilde{p}(r),h_r} b_{p^{-1}(r),h_r} \\
&= \sum_{\substack{\{(h_1,\dots,h_r):\\ 1\le h_1<h_2\\ \dots<h_r\le\nu\}}} \sum_{\text{all permutations } \tilde{p}} (\text{sign } \tilde{p}) c_{\tilde{p}(1),h_1} \dots c_{\tilde{p}(r),h_r} \\
&\qquad \sum_{\text{all permutations } p} (\text{sign } p^{-1}) b_{p^{-1}(1),h_1} \dots b_{p^{-1}(r),h_r} \\
&= \sum_{\substack{\{(h_1,\dots,h_r):\\ 1\le h_1<h_2\\ \dots<h_r\le\nu\}}} \left| \begin{array}{ccc} | & & | \\ \underline{c}_{h_1} & \dots & \underline{c}_{h_r} \\ | & & | \end{array} \right| \left| \begin{array}{c} - \ \underline{b}_{h_1}^T \ - \\ \vdots \\ - \ \underline{b}_{h_r}^T \ - \end{array} \right| \qquad (3)
\end{aligned}$$

By Lemma 1,

$$\begin{aligned}
\left| \begin{array}{ccc} | & & | \\ \underline{c}_{h_1} & \dots & \underline{c}_{h_r} \\ | & & | \end{array} \right| &= \sum_{\substack{\{h'_1,\dots,h'_r:\\ 1\le h'_j\le h_j,\\ h'_i\ne h'_j\ \forall\, i\ne j\}}} \left| \begin{array}{ccc} | & & | \\ \underline{a}_{h'_1} & \dots & \underline{a}_{h'_r} \\ | & & | \end{array} \right| \\
&\qquad \sum_{\substack{\{\mathcal{E}_1,\dots,\mathcal{E}_r:\\ \mathcal{E}_j\in S_{h'_j,h_j},\\ \mathcal{E}_j\cap\mathcal{E}_k=\emptyset\\ \forall\, j\ne k\}}} \prod_{j=1}^{r} g(\mathcal{E}_j) \qquad (4)
\end{aligned}$$

First we show that there is a bijective correspondence between terms of $|M_1|$ and terms of $|M_2|$.

Each product term of $|M_1|$ is of the form

$$\prod_{i=1}^{r} a_{p(i),h'_i} f_{e^i_1,e^i_2} \dots f_{e^i_{k_i-1},e^i_{k_i}} b_{p_b(i),h_i}$$

where p and p_b are permutations acting on $\{1,\dots,r\}$, $\{e^i_1,\dots,e^i_{k_i}\} \in S_{h'_i,h_i}$, and $\{e^i_1,\dots,e^i_{k_i}\} \cap \{e^j_1,\dots,e^j_{k_j}\} = \emptyset\ \forall\ i\ne j$. Let $\mathcal{C}_1$ be the set of variables $\{a_{p(i),h'_i}, f_{e^i_1,e^i_2},\dots,f_{e^i_{k_i-1},e^i_{k_i}}, b_{p_b(i),h_i} : i = 1,\dots,r\}$ in a given product term. The conditions on $\{h'_i, e^i_1,\dots,e^i_{k_i}, h_i : i = 1,\dots,r\}$ imply that no two variables in $\mathcal{C}_1$ appear in the same row or the same column in matrix M_2, and that if a column z in M_2 does not contain any of these variables, then $z \notin \bigcup_{i=1}^r \mathcal{E}_i$, so row $r+z$ also does not contain any of these variables. For such z, we append to $\mathcal{C}_1$ $M_2(r+z,z) = 1$. Then $\mathcal{C}_1$ comprises $r+\nu$ variables each occupying a different row and column of the $(r+\nu)\times(r+\nu)$ matrix M_2. The variables in $\mathcal{C}_1$ are thus variables of a product term in $|M_2|$.

Conversely, a product term of $|M_2|$ comprises $m+r$ entries of M_2, each from a different row and column of M_2. For any such nonzero product term, the variables form a set $\mathcal{C}_2 = \mathcal{A}\cup\mathcal{B}\cup\mathcal{F}\cup\mathcal{I}$, where

$$\begin{aligned}
\mathcal{A} &= \{a_{p_a(j),\hat{h}_j} : j = 1,\dots,r;\ \hat{h}_1 < \dots, \hat{h}_r\} \\
\mathcal{B} &= \{b_{p_b(i),h_i} : i = 1,\dots,r;\ h_1 < \dots < h_r\} \\
\mathcal{F} &= \{f_{x_l,y_l} : l = 1,\dots,|\mathcal{F}|;\ y_1 < \dots < y_{|\mathcal{F}|}\} \\
\mathcal{I} &= \{M_2(r+z_l,z_l) = 1 : l = 1,\dots,|\mathcal{I}|\},
\end{aligned}$$

p_a and p_b are permutations acting on $\{1,\dots,r\}$, and $\{h_1,\dots,h_r,x_1,\dots,x_{|\mathcal{F}|},z_1,\dots,z_{|\mathcal{I}|}\}$ and $\{\hat{h}_1,\dots,\hat{h}_r,y_1,\dots,y_{|\mathcal{F}|},z_1,\dots,z_{|\mathcal{I}|}\}$ are permutations of $\{1,\dots,\nu\}$.

For $i = 1,\dots,r$, h_i is equal to some y_l or some $\hat{h}_j$. Let $e^i_0 = h_i$. If $e^i_0 = y_l$, set $e^i_{-1} = x_l$; e^i_{-1} is in turn equal to some y_l or some $\hat{h}_j$. We proceed in this way, defining $e^i_{-2}, e^i_{-3}, \dots$, until we reach an index e^i_{-k} that is equal to some $\hat{h}_j$. We then set $k_i = k+1$, $h'_i = \hat{h}_j$, and $e^i_{k_i-l} = e^i_{-l}\ \forall\, l = 0,\dots,k_i-1$. With these definitions, $\mathcal{C}_2$ becomes $\{a_{p'_a(i),h'_i}, f_{e^i_1,e^i_2},\dots,f_{e^i_{k_i-1},e^i_{k_i}}, b_{p_b(i),h_i} : i = 1,\dots,r\} \cup \{M_2(r+z,z) = 1 : z \notin \bigcup_{i=1}^r \{e^i_1,\dots,e^i_{k_i}\}\}$, where the set of indices $\{h_i, h'_i, e^i_1,\dots,e^i_{k_i} : i = 1,\dots,r\}$ so defined satisfies $h'_i = e^i_1 < \dots < e^i_{k_i} = h_i$ and $\{e^i_1,\dots,e^i_{k_i}\} \cap \{e^j_1,\dots,e^j_{k_j}\} = \emptyset\ \forall\ i \ne j$. From Equations 3 and 4, we can see that $\mathcal{C}_2$ comprises variables of some product term in $|M_1|$.

It remains to show equality of the sign of the product term in $|M_1|$ and that in $|M_2|$ whose variables constitute the same set $\mathcal{C} = \mathcal{A}\cup\mathcal{B}\cup\mathcal{F}\cup\mathcal{I}$, where

$$\begin{aligned}
\mathcal{A} &= \{a_{p(i),h'_i} : i = 1,\dots,r\} \\
&= \{a_{p_a(j),\hat{h}_j} : j = 1,\dots,r;\ \hat{h}_1 < \dots, \hat{h}_r\} \\
\mathcal{B} &= \{b_{p_b(i),h_i} : i = 1,\dots,r;\ h_1 < \dots < h_r\} \\
\mathcal{F} &= \{f_{e^i_1,e^i_2},\dots,f_{e^i_{k_i-1},e^i_{k_i}} : i = 1,\dots,r\}
\end{aligned}$$

$$
\begin{aligned}
&= \{f_{x_l,y_l} : l = 1,\ldots,|\mathcal{F}|;\ y_1 < \ldots < y_{|\mathcal{F}|}\} \\
\mathcal{I} &= \{M_2(r+z_l, z_l) = 1 : l = 1,\ldots,|\mathcal{I}|\}
\end{aligned}
$$

Let the corresponding product terms in $|M_1|$ and $|M_2|$ be $\sigma_1^{\mathcal{C}}\pi_{\mathcal{C}}$ and $\sigma_2^{\mathcal{C}}\pi_{\mathcal{C}}$ respectively, where $\pi_{\mathcal{C}} = \prod_{\xi\in\mathcal{C}}\xi$, and $\sigma_1^{\mathcal{C}}$, $\sigma_2^{\mathcal{C}}$ are the respective signs, equal to 1 or -1.

From Equations 3 and 4, we see that $\sigma_1^{\mathcal{C}} = (\text{sign } p)(\text{sign } p_b)$. Let p' be the permutation that sorts $\{h'_1,\ldots,h'_r\}$ in ascending order, i.e. $h'_{p'(1)} < \ldots < h'_{p'(r)}$. Then $p(p'(i)) = p_a(i)$. So sign $p = (\text{sign } p')(\text{sign } p_a)$, and $\sigma_1^{\mathcal{C}} = (\text{sign } p_a)(\text{sign } p_b)(\text{sign } p')$.

Consider the following procedure for determining sign p', consisting of a series of steps in which the variables $f_{x_l,y_l} \in \mathcal{F}$ are considered one by one in ascending order of index l. We maintain an ordered set $\mathcal{S} = \{s_1,\ldots,s_r\}$ of distinct elements from $[1,\nu]$, and an ordered set $\mathcal{Q}$ of the indices of elements of $\mathcal{S}$ sorted in ascending order, i.e. $\mathcal{Q} = \{q_1,\ldots,q_r\}$ where $s_{q_1} < \ldots < s_{q_r}$. Each s_i is initialized to h'_i. We carry out the following procedure for $l = 1,\ldots,|\mathcal{F}|$ in order. At each step l, $x_l = s_{\gamma_l}$ for some index $\gamma_l \in [1,r]$. We set $s_{\gamma_l} = y_l$, and let n_l be the number of indices i for which $x_l < s_i < y_l$. If $n_l \geq 1$, the change in $\mathcal{Q}$ is a cyclic permutation p_l of n_l+1 elements. If $n_l = 0$, there is no change in $\mathcal{Q}$, and p_l is the identity permutation. We continue in this manner, noting that at every step, for the index y_l under consideration, all indices less than y_l are either equal to some s_i or some x_k where $y_k < y_l$. Since all the x_k's are distinct, x_l must equal some s_i.

At the end of the procedure, $s_i = h_i \forall i$, so the elements of $\mathcal{S}$ are in ascending order and $\mathcal{Q} = \{1,\ldots,r\}$. Permutation p' is equal to the composition of the cyclic permutations p_l, $l = 1,\ldots,|\mathcal{F}|$. Since sign $p_l = (-1)^{n_l}$, sign $p' = (-1)^{\sum_{l=1}^{|\mathcal{F}|} n_l}$.

Next we determine $\sigma_2^{\mathcal{C}}$. Let $M_{\mathcal{C}}, A_{\mathcal{C}}$ and $B_{\mathcal{C}}$ be the matrices obtained from M_2, A and B respectively by setting to 0 all entries involving variables not in $\mathcal{C}$, and let ξ_j be the nonzero entry in column j of $M_{\mathcal{C}}$. Let λ be the number of inversions in $M_{\mathcal{C}}$, where an inversion is a pair of nonzero entries at positions (q_1^i, q_1^j), (q_2^i, q_2^j) such that $q_1^i < q_2^i$ and $q_2^j < q_1^j$. Then $\sigma_2^{\mathcal{C}} = (-1)^{\lambda+|\mathcal{F}|}$, since each entry involving a variable in $|\mathcal{F}|$ has a negative sign in M_2.

For each j, let u_j be the number of inversions involving ξ_j and entries ξ_k, $k > j$. Then $\sigma_2^{\mathcal{C}} = (-1)^{\sum_{j=1}^{\nu+r} u_j + |\mathcal{F}|} = (-1)^{U_a+U_b+U_f+U_i+|\mathcal{F}|}$, where $U_a = \sum_{j:\xi_j\in\mathcal{A}} u_j$, $U_b = \sum_{j:\xi_j\in\mathcal{B}} u_j$, $U_f = \sum_{j:\xi_j\in\mathcal{F}} u_j$ and $U_i = \sum_{j:\xi_j\in\mathcal{I}} u_j$.

If $\xi_j \in \mathcal{A}$ is involved in an inversion with ξ_k, $k > j$, then ξ_k must be a term in $\mathcal{A}$ as it is in a smaller-numbered row. Thus, the number of inversions involving entries in $\mathcal{A}$ is equal to the number of inversions in the $r\times r$ submatrix of $A_{\mathcal{C}}$ consisting of columns j for which $\xi_j \in \mathcal{A}$. So $(-1)^{U_a} = \text{sign } p_a$. Similarly, if $\xi_j \in \mathcal{B}$ is involved in an inversion with ξ_k, $k > j$, then ξ_k must be a term in $\mathcal{B}$ as it is in a larger-numbered column. So $(-1)^{U_b} = \text{sign } p_b$.

For j such that $\xi_j \in \mathcal{F}$, carry out the procedure described earlier that considers the terms $\xi_j = f_{x_l,y_l}$ one by one in ascending order of index l. At each step we compute $\mathcal{S}$, $\mathcal{Q}$ and n_l as before, noting that $x_l = s_{\gamma_l}$ for some $\gamma_l \in [1,r]$, and that the entries ξ_k, $k > y_l$, with which f_{x_l,y_l} is involved in inversions are $\{\xi_k : k = s_i > y_l\} \cup \{f_{x_g,y_g} : x_g = s_i < x_l, y_g = k > y_l\} \cup \{b_{k-\nu,g} : g = s_i < x_l\}$. These are in bijective correspondence with the elements of the set $\{s_i : s_i < x_l = s_{\gamma_j} \text{ or } s_i > y_l\}$. Thus, $u_j = r - 1 - n_l$.

For j such that $\xi_j \in \mathcal{I}$, there are exactly $j-1$ entries in columns $1,\ldots,j-1$, and exactly $\nu-j$ entries in rows $j+r+1,\ldots,\nu+r$ which are not involved in inversions with ξ_j. Thus, $u_j = \nu + r - 1 - (j-1+\nu-j) = r$.

Combining these expressions, and noting that $\mathcal{I} = \nu - r - |\mathcal{F}|$, we have

$$
\begin{aligned}
U_f + U_i &= |\mathcal{F}|(r-1) - \sum_{l=1}^{|\mathcal{F}|} n_l + (\nu - r - |\mathcal{F}|)r \\
&= (\nu-r)r - |\mathcal{F}| - \sum_{l=1}^{|\mathcal{F}|} n_l \\
\sigma_2^{\mathcal{C}} &= (-1)^{U_a}(-1)^{U_b}(-1)^{U_f+U_i+|\mathcal{F}|} \\
&= (\text{sign } p_a)(\text{sign } p_b)(-1)^{(\nu-r)r - \sum_{l=1}^{|\mathcal{F}|} n_l}
\end{aligned}
$$

If r is even, then $(\nu-r)r$ is even, and

$$
\begin{aligned}
\sigma_2^{\mathcal{C}} &= (\text{sign } p_a)(\text{sign } p_b)(-1)^{-\sum_{l=1}^{|\mathcal{F}|} n_l} \\
&= (\text{sign } p_a)(\text{sign } p_b)(-1)^{\sum_{l=1}^{|\mathcal{F}|} n_l} \\
&= \sigma_1^{\mathcal{C}}
\end{aligned}
$$

If r is odd, then $(\nu-r)r$ is even if ν is odd, and odd if ν is even. So $\sigma_2^{\mathcal{C}} = \sigma_1^{\mathcal{C}}$ if ν is odd, and $\sigma_2^{\mathcal{C}} = -\sigma_1^{\mathcal{C}}$ if ν is even. ■

By similar reasoning, we can also show that for M_3 defined as $\begin{bmatrix} A & 0 \\ -I+F & B^T \end{bmatrix}$,

$$|M_1| = (-1)^{\nu(r+1)}\,|M_3|$$

Proof of Theorem 3: By Theorem 2, the form of the transfer matrix determinant $|AGB_\beta^T|$ for any receiver β matches the form of the Edmonds matrix determinant $\begin{vmatrix} A & 0 \\ I-F & B_\beta^T \end{vmatrix}$. Since no variable $a_{x,j}$, $f_{i,j}$ or $b_{\beta_{i,j}}$ appears in more than one entry of the Edmonds matrix, no product term in the determinant polynomial contains a variable $a_{x,j}$, $f_{i,j}$ or $b_{\beta_{i,j}}$ raised to an exponent greater

than 1. Thus, the largest exponent of any variable in a product of d such determinants is at most d.

The desired bound follows from the theorem, given in [4], that there exists a solution in $\mathbb{F}_q$ if q is greater than or equal to the maximal degree of any variable in the product of the determinant polynomials for all receivers. ■

V. Conclusions and further work

We have presented two alternative formulations of the algebraic condition of [4] for checking the feasibility of a multicast connection problem, or the validity of a linear network code. The first is in terms of network flows, and the second in terms of the Edmonds matrix determinant condition for checking if a bipartite graph has a perfect matching. These offer combinatorial insights relating network coding to network flows, and have led to new results on randomized distributed transmission and compression of information, presented in our companion paper [2]. We have also given an upper bound on network coding complexity that substantially tightens the bound given in [4], a result which follows easily from the Edmonds matrix formulation.

There is much further work to be done in deepening our understanding of the fundamental connections between network coding and network flow. We suspect that it may be possible to bring into a similar framework other network connection problems for which the max-flow min-cut condition is necessary and sufficient for feasibility, e.g. connection problems where sets of processes transmitted to different receivers are mutually disjoint, or problems with two receivers where one receiver receives a subset of processes transmitted to the other [4]. It would be interesting to see whether there are more results and insights in the rich field of network flow optimization that can be applied to network coding.

References

[1] R. Ahlswede, N. Cai, S.-Y.R. Li and R.W. Yeung, "Network Information Flow", IEEE-IT, vol. 46, pp. 1204-1216, 2000.

[2] T. Ho, R. Koetter, M. Médard, D. R. Karger and M. Effros, "The Benefits of Coding over Routing in a Randomized Setting", Submitted to the 2003 IEEE International Symposium on Information Theory.

[3] R. M. Karp, E. Upfal and A. Wigderson, "Constructing a Perfect Matching is in Random NC", Combinatorica 6 (1) (1986), pp 35-48.

[4] R. Koetter and M. Médard, "Beyond Routing: An Algebraic Approach to Network Coding", Proceedings of the 2002 IEEE Infocom, 2002.

[5] S.-Y.R. Li and R.W. Yeung, "Linear Network Coding", preprint, 1999.