Computing linear functions by linear coding over networks

Rathinakumar Appuswamy, Massimo Franceschetti

Abstract

We consider the scenario in which a set of sources generate messages in a network and a receiver node demands an arbitrary *linear function* of these messages. We formulate an algebraic test to determine whether an arbitrary network can compute linear functions using *linear codes*. We identify a class of linear functions that can be computed using linear codes in every network that satisfies a natural cut-based condition. Conversely, for another class of linear functions, we show that the cut-based condition does not guarantee the existence of a linear coding solution. For linear functions over the binary field, the two classes are complements of each other.

I. Introduction

In many practical networks, including sensor networks and vehicular networks, receivers demand a function of the messages generated by the sources that are distributed across the network rather than the generated messages. This situation is studied in the framework of network computing [3]–[7], [10], [11]. The classical network coding model of Ahlswede, Cai, Li, and Yeung [1] can be viewed as a the special case of network computing in which the function to be computed at the receivers corresponds to a subset of the source messages and communication occurs over a network with noiseless links.

In the same noiseless set up of [1], we consider the scenario in which a set of source nodes generate messages over a finite field and a single receiver node computes a linear function of these messages. We ask whether this linear function can be computed by performing linear coding operations at the intermediate nodes.

In multiple-receiver networks, if each receiver node demands a subset of the source messages (which is an example of a linear function), then Dougherty, Freiling, and Zeger [8] showed that linear codes are not sufficient to recover the source messages. Similarly, if each receiver node demands the sum of the source messages, then Ray and Dei [4] showed that linear codes are also not sufficient to recover the source messages. In contrast, in single-receiver networks linear codes are sufficient for both the above problems and a simple cut-based condition can be used to test whether a linear solution exists.

Our contribution is as follows. We extend above results investigating if a similar cut-based condition guarantees the existence of a linear solution when the receiver node demands an arbitrary linear function of the source messages. We identify two classes of functions, one for which the cut-based condition is sufficient for solvability and the other for which it is not. These classes are complements of each other when the source messages are over the binary field. Along the way, we develop an algebraic framework to study linear codes and provide an algebraic condition to test whether a linear solution exists, similar to the one given by Koetter and Médard [2] for classical network coding.

The paper is organized as follows. We formally introduce the network computation model in Section I-A. In Section II we develop the necessary algebraic tools to study linear codes and introduce the cut-based condition. In Section III, we show the main results for the two classes of functions. Section IV concludes the paper, mentioning some open problems.

A. Network model and preliminaries

In this paper, a *network* $\mathcal{N}$ consists of a finite, directed acyclic multigraph $G = (\mathcal{V}, \mathcal{E})$, a set of *source nodes* $S = \{\sigma_1, \ldots, \sigma_s\} \subseteq \mathcal{V}$, and a *receiver* $\rho \in \mathcal{V}$. Such a network is denoted by $\mathcal{N} = (G, S, \rho)$. We use the word "graph" to mean a multigraph, and "network" to mean a single-receiver network. We assume that $\rho \notin S$, and that the graph G contains a directed path from every node in $\mathcal{V}$ to the receiver ρ. For each node $u \in \mathcal{V}$, let $\mathcal{E}_{in}(u)$ and $\mathcal{E}_{out}(u)$ denote the in-edges and out-edges of u respectively. We also assume (without loss of generality) that if a network node has no in-edges, then it is a source node. We use s to denote the number of sources $|S|$ in the network.

An *alphabet* $\mathcal{A}$ is a nonzero finite field. For any positive integer m, any vector $x \in \mathcal{A}^m$, and any i, let x_i denote the i-th component of x. For any index set $K = \{i_1, i_2, \ldots, i_q\} \subseteq \{1, 2, \ldots, m\}$ with $i_1 < i_2 < \ldots < i_q$, let x_K denote the vector $(x_{i_1}, x_{i_2}, \ldots, x_{i_q}) \in \mathcal{A}^{|K|}$.

The *network computing* problem consists of a network $\mathcal{N}$, a source alphabet $\mathcal{A}$, and a *target function*

$$f \ : \ \mathcal{A}^s \longrightarrow \mathcal{B}$$

This work was supported by the National Science Foundation award CNS 0916778 and the UCSD Center for Wireless Communications.

The authors are with the Department of Electrical and Computer Engineering, University of California, San Diego, La Jolla, CA 92093-0407. (rathnam@ucsd.edu, massimo@ece.ucsd.edu)

where $\mathcal{B}$ is the *decoding alphabet*. A target function f is *linear* if there exists a matrix T over $\mathcal{A}$ such that

$$f(x) = Tx^t, \quad \forall\ x \in \mathcal{A}^s$$

where 't' denotes matrix transposition. For linear target functions the decoding alphabet is of the form $\mathcal{A}^l$, with $1 \leq l \leq s$. Without loss of generality, we assume that T is full rank (over $\mathcal{A}$) and has no zero columns. For example, if T is the $s \times s$ identity matrix, then the receiver demands the complete set of source messages, and this corresponds to the classical network coding problem. On the other hand, if T is the row vector of 1's, then the receiver demands a sum (over $\mathcal{A}$) of the source values. Let n be a positive integer. Given a network $\mathcal{N}$ with source set S and alphabet $\mathcal{A}$, a *message generator* is a mapping

$$\alpha\ :\ S \longrightarrow \mathcal{A}^n.$$

For each source $\sigma_i \in S$, $\alpha(\sigma_i)$ is called a *message vector* and it can be viewed as an element of $\mathbb{F}_{q^n}$ (rather than as a vector).

Definition I.1. A *linear network code* in a network $\mathcal{N}$ consists of the following:

(i) Every edge $e \in \mathcal{E}$ *carries an element* of $\mathbb{F}_{q^n}$ and this element is denoted by z_e. For any node $v \in \mathcal{V} - \rho$ and any out-edge $e \in \mathcal{E}_{out}(v)$, the network code specifies an *encoding function* $h^{(e)}$ of the form:

$$h^{(e)} = \begin{cases} x_{1,e}\alpha(u) + \displaystyle\sum_{\hat{e}\in\mathcal{E}_{in}(u)} x_{\hat{e},e} z_{\hat{e}} & \text{if } u \in S \\ \displaystyle\sum_{\hat{e}\in\mathcal{E}_{in}(u)} x_{\hat{e},e} z_{\hat{e}} & \text{otherwise} \end{cases} \tag{1}$$

where $x_{\hat{e},e}, x_{1,e} \in \mathbb{F}_{q^n}$ for all $\hat{e} \in \mathcal{E}_{in}(u)$.

(ii) The *decoding function* ψ outputs a vector of length l whose j-th component is of the form:

$$\sum_{e\in\mathcal{E}_{in}(\rho)} x_{e,j} z_e \tag{2}$$

where $x_{e,j} \in \mathbb{F}_{q^n}$ for all $e \in \mathcal{E}_{in}(\rho)$. The arithmetic in (1) and (2) is performed over $\mathbb{F}_{q^n}$.

In this paper, by a *network code*, we always mean a linear network code. In the literature, the class of network codes we define here is referred to as *scalar linear codes*. These codes were introduced and studied in [2]. A more general class of linear codes over $\mathbb{F}_{q^n}$ were defined and studied in [8], [9].

Depending on the context, we may view z_e as a vector of length-n over $\mathbb{F}_q$ or as an element of $\mathbb{F}_{q^n}$. Without explicit mention, we use the fact that the addition of $a, b \in \mathbb{F}_{q^n}$ as elements of a finite field coincides with their sum as elements of a vector space over $\mathbb{F}_q$. Furthermore, we also view $\mathbb{F}_q$ as a subfield of $\mathbb{F}_{q^n}$ without explicitly stating the inclusion map. Let $z_{e_1}, z_{e_2}, \ldots, z_{e_{|\mathcal{E}_{in}(\rho)|}}$ denote the vectors carried by the in-edges of the receiver.

Definition I.2. A linear network code over $\mathbb{F}_{q^n}$ is called *a linear solution for computing f in $\mathcal{N}$* (or simply a *linear solution* if f and $\mathcal{N}$ are clear from the context) if the decoding function ψ is such that for every message generator α,

$$\psi\left(z_{e_1}, \cdots, z_{e_{|\mathcal{E}_{in}(\rho)|}}\right)_j = f\Big(\alpha(\sigma_1)_j, \cdots, \alpha(\sigma_s)_j\Big) \quad \text{for all } j \in \{1, 2, \ldots, n\}. \tag{3}$$

Remark I.3. Each source generates n symbols over $\mathbb{F}_q$ (viewing $\mathbb{F}_{q^n}$ as a vector space over $\mathbb{F}_q$) and the decoder computes the target function f for each set of source symbols.

A set of edges $C \subseteq \mathcal{E}$ is said to *separate* sources $\sigma_{m_1}, \ldots, \sigma_{m_d}$ from the receiver ρ, if for each $i \in \{1, 2, \ldots, d\}$, every path from σ_{m_i} to ρ contains at least one edge in C. A set $C \in \mathcal{E}$ is said to be a *cut* if it separates at least one source from the receiver. Let $\Lambda(\mathcal{N})$ denote the set of all cuts in network $\mathcal{N}$.

For any matrix $T \in \mathbb{F}_q^{l\times s}$, let T_i denote its i-th column. For an index set $K \in \{1, 2, \ldots, s\}$, let T_K denote the $l \times |K|$ submatrix of T obtained by choosing the columns of T indexed by K. If C is a cut in a network $\mathcal{N}$, we define the set

$$K_C = \{i \in S : C \text{ disconnects } \sigma_i \text{ from } \rho\}.$$

Finally, for any network $\mathcal{N}$ and matrix T, we define

$$\text{min-cut}(\mathcal{N}, T) = \min_{C\in\Lambda(\mathcal{N})} \frac{|C|}{\text{rank}(T_{K_C})}. \tag{4}$$

II. Algebraic framework

A. An algebraic test for the existence of a linear solution

Linear solvability for the classical network coding problem was shown to be equivalent to the existence of a non-empty algebraic variety in [2]. In the following, we present an analogous characterization for computing linear functions, providing an algebraic test to determine whether a linear solution for computing a linear function exists. The reverse problem of constructing a multiple-receiver network coding (respectively, network computing) problem given an arbitrary set of polynomials, which is solvable if and only if the corresponding set of polynomials is simultaneously solvable is considered in reference [9] (respectively, [4]).

We begin by giving some definitions and stating a technical lemma, followed by the main theorem below.

For any edge $e = (u, v) \in \mathcal{E}$, let *head(e)* $= v$ and *tail(e)* $= u$. Associated with a linear code over $\mathbb{F}_{q^n}$, we define the following three types of matrices:

- For each source $\sigma_\tau \in S$, define the $1 \times |\mathcal{E}|$ matrix A_τ as follows:

$$(A_\tau)_{1,j} = \begin{cases} x_{1,e_j} & \text{if } e_j \in \mathcal{E}_{out}(\sigma_t) \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

- Similarly define the $l \times |\mathcal{E}|$ matrix B as follows:

$$B_{i,j} = \begin{cases} x_{e_j,i} & \text{if } e_j \in \mathcal{E}_{in}(\rho) \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

- Define the $|\mathcal{E}| \times |\mathcal{E}|$ matrix F as follows:

$$F_{i,j} = \begin{cases} x_{e_i,e_j} & \text{if } \textit{head}(e_i) = \textit{tail}(e_j) \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

Since the graph G associated with the network is acyclic, we can assume that the edges $e_1, e_2, \ldots$ are ordered such that the matrix F is strictly upper-triangular. Let I denote the identity matrix of suitable dimension. Consider a network $\mathcal{N}$ with alphabet $\mathbb{F}_q$ and consider a linear code over $\mathbb{F}_{q^n}$ with associated matrices $A_1, A_2, \ldots, A_s, B$ and F. For every $\tau \in \{1, 2, \ldots, s\}$, define the $1 \times l$ matrix

$$M_\tau = A_\tau (I - F)^{-1} B^t. \tag{8}$$

Now let x_A be a vector containing all the non-zero entries of the matrices $A_\tau, \tau = 1, 2, \cdots, s$, and let x_B (respectively, x_F) be a vector containing all the non-zero entries of the matrix B (respectively, F).

By abusing notation, depending on the context we may view x_{e_i,e_j}, x_{i,e_j}, $x_{e_i,j}$ as elements of $\mathbb{F}_{q^n}$ or as indeterminates. Thus, each of the matrices defined above may either be a matrix over $\mathbb{F}_{q^n}$ or a matrix over the polynomial ring $R = \mathbb{F}_{q^n}[x_A, x_F, x_B]$. The context should make it clear which of these two notions is being referred to at any given point.

Lemma II.1. *The following two statements hold:*

1) *The matrix $I - F$ has a polynomial inverse with coefficients in $\mathbb{F}_{q^n}[x_F]$, the ring of polynomials in the variables constituting x_F.*
2) *The decoding function can be written as*

$$\sum_{\tau=1}^{s} \alpha(\sigma_\tau)\, A_\tau (I - F)^{-1} B^t$$

Proof: The first assertion is a restatement of [2, Lemma 2] and the second assertion follows from [2, Theorem 3]. ∎

Definition II.2. Let R be a polynomial ring. The ideal generated by a subset $X \subset R$ and denoted by $\langle X \rangle$ is the smallest ideal in R containing X.

Let $\mathcal{N}$ be a network with alphabet $\mathbb{F}_q$. Let $R = \mathbb{F}_q[x_A, x_F, x_B]$ and $T \in \mathbb{F}_q^{l \times s}$. Consider a linear network code for computing the linear function corresponding to T in $\mathcal{N}$ and the associated matrices $M_\tau, \tau = 1, 2, \ldots, s$ over R and define

$$Z_\tau = (T_\tau)^t - M_\tau \text{ for } \tau = 1, 2, \ldots, s.$$

Let J denote the ideal generated by the elements of $Z_\tau \in R^{1 \times l}, \tau = 1, 2, \ldots, s$ in the ring R. More formally, let

$$J = \langle \{ \{ (Z_\tau)_1, (Z_\tau)_2, \ldots, (Z_\tau)_l \} : \tau = 1, 2, \ldots, s \} \rangle .$$

The polynomials $(Z_i)_j$ are referred to as the *generating polynomials* of the ideal J. We denote the Groböner basis of an ideal generated by subset $X \subset R$ of a polynomial ring R by $\mathcal{G}(X)$. The following theorem is a consequence of Hilbert Nullstellensatz (see [13, Lemma VIII.7.2] and the remark after [13, Proposition VIII.7.4]).

Theorem II.3. *Consider a network $\mathcal{N}$ with alphabet $\mathbb{F}_q$ and the linear target function f corresponding to a matrix $T \in \mathcal{A}^{l\times s}$. There exists an $n > 0$ and a linear solution over $\mathbb{F}_{q^n}$ for computing f in $\mathcal{N}$ if and only if $\mathcal{G}(J) \neq \{1\}$.*

Proof: From Lemma II.1, the vector computed at the receiver can be written as

$$\psi\left(z_{e_1},\cdots,z_{e_{|\mathcal{E}_{in}(\rho)|}}\right) = \begin{pmatrix} M_1^t & M_2^t & \cdots & M_s^t \end{pmatrix} \begin{pmatrix} \alpha(\sigma_1) \\ \alpha(\sigma_2) \\ \vdots \\ \alpha(\sigma_s) \end{pmatrix}. \tag{9}$$

On the other hand, to compute the linear function corresponding to T, the decoding function must satisfy

$$\psi\left(z_{e_1},\cdots,z_{e_{|\mathcal{E}_{in}(\rho)|}}\right) = T \begin{pmatrix} \alpha(\sigma_1) \\ \alpha(\sigma_2) \\ \vdots \\ \alpha(\sigma_s) \end{pmatrix}. \qquad \text{[from (3)]} \tag{10}$$

It follows that the encoding coefficients in a linear solution must be such that

$$(T_\tau)^t - M_\tau = 0 \text{ for } \tau = 1, 2, \ldots, s. \qquad \text{[from (9) and (10)]} \tag{11}$$

If we view the coding coefficients as variables, then it follows that a solution must simultaneously solve the generating polynomials of the corresponding ideal J. By [13, Lemma VIII.7.2], such a solution exists over the algebraic closure $\bar{\mathbb{F}}_q$ of $\mathbb{F}_q$ if and only if $J \neq \mathbb{F}_q[x_A, x_F, x_B]$. Furthermore, $J \neq \mathbb{F}_q[x_A, x_F, x_B]$ if and only if $\mathcal{G}(J) \neq \{1\}$. Moreover, a solution exists over the algebraic closure $\bar{\mathbb{F}}_q$ of $\mathbb{F}_q$ if and only if it exists over some extension field $\mathbb{F}_{q^n}$ of $\mathbb{F}_q$ and the proof is now complete. ■

B. Minimum cut condition

It is clear that the set of linear functions that can be solved in a network depends on the network topology. It is easily seen that a linear solution for computing a linear target function corresponding to $T \in \mathbb{F}_q^{l\times s}$ exists only if the network $\mathcal{N}$ is such that for every $C \in \Lambda(\mathcal{N})$, the value of the cut $|C|$ is at least the rank of the submatrix T_{K_C} (recall that K_C is the index set of the sources separated by the cut C). This observation is stated in the following lemma which is an immediate consequence of the cut-based bound in [10, Theorem 2.1].

Lemma II.4. *For a network $\mathcal{N}$, a necessary condition for the existence of a linear solution for computing the target function corresponding to $T \in \mathbb{F}_q^{l\times s}$ is*

$$\textit{min-cut}(\mathcal{N}, T) \geq 1.$$

We now consider two special cases. First, consider the case in which the receiver demands all the source messages. The corresponding T is given by the $s \times s$ identity matrix I and the condition min-cut$(\mathcal{N}, T) \geq 1$ reduces to

$$\frac{|C|}{|K_C|} \geq 1 \quad \forall\ C \in \Lambda(\mathcal{N})$$

i.e., the number of edges in the cut be at least equal to the number of sources separated by the cut. Second, consider the case in which the receiver demands the sum of the source messages. The corresponding matrix T is an $1 \times s$ row vector and the requirement that min-cut$(\mathcal{N}, T) \geq 1$ reduces to

$$|C| \geq 1 \quad \forall\ C \in \Lambda(\mathcal{N})$$

i.e., all the sources have a directed path to the receiver. For both of the above cases, the cut condition in Lemma II.4 is also sufficient for the existence of a solution. This is shown in [10, Theorem 3.1 and Theorem 3.2] and is reported in the following Lemma:

Lemma II.5. *Let $l \in \{1, s\}$. For a network $\mathcal{N}$ with the linear target function f corresponding to a matrix $T \in \mathcal{A}^{l\times s}$, a linear solution exists if and only if min-cut$(\mathcal{N}, T) \geq 1$.*

The focus in the rest of the paper is to extend above results to the case $l \notin \{1, s\}$ by using the algebraic test of Theorem II.3.

III. Computing linear functions

In the following, we first define an equivalence relation among matrices and then use it to identify a set of functions that are linearly solvable in every network satisfying the condition min-cut$(\mathcal{N}, T) \geq 1$. We then construct a linear function outside this set, and a corresponding network with min-cut$(\mathcal{N}, T) \geq 1$, on which such a function cannot be computed with linear codes. Finally, we use this example as a building block to identify a set of linear functions for which there exist networks satisfying the min-cut condition and on which these functions are not solvable.

Notice that for a linear function with matrix $T \in \mathbb{F}_q^{l \times s}$, each column of T corresponds to a single source node. Hence, for every $s \times s$ permutation matrix Π, computing Tx is equivalent to computing $T\Pi x$ after appropriately renaming the source nodes. Furthermore, for every $l \times l$ full rank matrix Q over $\mathbb{F}_q$, computing Tx is equivalent to computing QTx. These observations motivate the following definition:

Definition III.1. Let $T \in \mathbb{F}_2^{l \times s}$ and $T' \in \mathbb{F}_2^{l \times s}$. We say $T \sim T'$ if there exist an invertible matrix Q of size $l \times l$ and a permutation matrix Π of size $s \times s$ such that $T = QT'\Pi$, and $T \nsim T'$ if such Q and Π do not exist.

Since T is assumed to be a full rank matrix, Π can be chosen such that the first l columns of $T\Pi$ are linearly independent. Let $\hat{T}$ denote the first l columns of $T\Pi$. By choosing $Q = \hat{T}^{-1}$, we have $T \sim QT\Pi = (I \ P)$ where P is an $l \times s - l$ matrix. So for an arbitrary linear target function f and an associated matrix T, there exists an $l \times s - l$ matrix P such that $T \sim (I \ P)$. Without loss of generality, we assume that each column of T associated with a target function is non-zero.

Theorem III.2. *Consider a network $\mathcal{N}$ with a linear target function corresponding to a matrix $T \in \mathbb{F}_q^{(s-1) \times s}$ (i.e., $l = s - 1$). If*

$$T \sim (I \ u)$$

where u is a column vector of units, then a necessary and sufficient condition for the existence of a linear solution is min-cut$(\mathcal{N}, T) \geq 1$.

Proof: Let $T = (I \ u)$. The 'necessary' part is clear from Lemma II.4. We now focus on the 'sufficiency' part. Notice that for each $\tau = 1, 2, \ldots, s$, the matrix M_τ (computed as in (8)) is a row vector of length $s - 1$. Stack these s row vectors to form an $s \times (s-1)$ matrix M as follows,

$$M = \begin{pmatrix} M_1 \\ M_2 \\ \vdots \\ M_s \end{pmatrix}.$$

Let $M_{(i)}$ denote the $(s-1) \times (s-1)$ submatrix of M obtained by deleting its i-th row.
Claim 1: The matrix

$$\prod_{i=1}^{s} M_{(i)}$$

has a non-zero determinant over the ring $R = \mathbb{F}_q[x_A, x_F, x_B]$.
Claim 2: For each $i = 1, 2, \ldots, s-1$, we have $\left(A_s(I-F)^{-1}B^t M_{(s)}^{-1}\right)_i \neq 0$.
By Claim 1 and the sparse zeros lemma [2], [12], it follows that that there exists some $n > 0$ such that the variables $x_{e',e}, x_{e,l}$ can be assigned values over $\mathbb{F}_{q^n}$ so that the $s \times (s-1)$ matrix

$$M = \begin{pmatrix} A_1(I-F)^{-1}B^t \\ A_2(I-F)^{-1}B^t \\ \vdots \\ A_s(I-F)^{-1}B^t \end{pmatrix}$$

is such that any of its $(s-1) \times (s-1)$ submatrices $M_{(i)}, i = 1, 2, \ldots, s$ obtained by deleting the i-th row in M, is full rank over $\mathbb{F}_{q^n}$. Define two $s-1 \times s-1$ diagonal matrices U and D such that for $i \in \{1, 2, \cdots, s-1\}$

$$\begin{aligned} U_{i,i} &= u_i \\ D_{i,i} &= \left(A_s(I-F)^{-1}B^t M_{(s)}^{-1}\right)_i. \end{aligned} \tag{12}$$

Now define the following matrices over $\mathbb{F}_{q^n}$:

$$\begin{aligned} \bar{B} &= D^{-1}U(M_{(s)}^t)^{-1}B \\ \bar{A}_i &= u_i^{-1}\left(A_s(I-F)^{-1}\bar{B}^t\right)_i A_i \quad i = 1, 2, \ldots, s-1 \\ \bar{A}_s &= A_s. \end{aligned} \tag{13}$$

By by Claim 2 it follows that D^{-1} exists. If the matrices $\bar{A}_\tau, F$, and $\bar{B}$ define a linear network code, then by Lemma II.1, the vector received by ρ can be written as,

$$\bar{M}^t \begin{pmatrix} \alpha(\sigma_1) \\ \alpha(\sigma_2) \\ \vdots \\ \alpha(\sigma_s) \end{pmatrix} \tag{14}$$

where,

$$\bar{M} = \begin{pmatrix} \bar{A}_1(I-F)^{-1}\bar{B}^t \\ \bar{A}_2(I-F)^{-1}\bar{B}^t \\ \vdots \\ \bar{A}_s(I-F)^{-1}\bar{B}^t \end{pmatrix}. \tag{15}$$

We have

$$\begin{aligned}
\begin{pmatrix} A_1(I-F)^{-1}\bar{B}^t \\ A_2(I-F)^{-1}\bar{B}^t \\ \vdots \\ A_s(I-F)^{-1}\bar{B}^t \end{pmatrix} &= \begin{pmatrix} A_1(I-F)^{-1}(D^{-1}U(M_{(s)}^t)^{-1}B)^t \\ A_2(I-F)^{-1}(D^{-1}U(M_{(s)}^t)^{-1}B)^t \\ \vdots \\ A_s(I-F)^{-1}(D^{-1}U(M_{(s)}^t)^{-1}B)^t \end{pmatrix} && [\text{from } \bar{B} = D^{-1}U(M_{(s)}^t)^{-1}B] \\
&= \begin{pmatrix} A_1(I-F)^{-1}B^tM_{(s)}^{-1} \\ A_2(I-F)^{-1}B^tM_{(s)}^{-1} \\ \vdots \\ A_s(I-F)^{-1}B^tM_{(s)}^{-1} \end{pmatrix} D^{-1}U && [\text{from } \left((M_{(s)}^t)^{-1}\right)^t = M_{(s)}^{-1}] \\
&= \begin{pmatrix} I \\ A_s(I-F)^{-1}B^tM_{(s)}^{-1} \end{pmatrix} D^{-1}U && [\text{from construction of } M_{(s)}] \qquad (16) \\
\begin{pmatrix} \bar{A}_1(I-F)^{-1}\bar{B}^t \\ \bar{A}_2(I-F)^{-1}\bar{B}^t \\ \vdots \\ \bar{A}_s(I-F)^{-1}\bar{B}^t \end{pmatrix} &= \begin{pmatrix} U^{-1}D \\ A_s(I-F)^{-1}B^tM_{(s)}^{-1} \end{pmatrix} D^{-1}U && [\text{from (13) and (16) }] \\
&= \begin{pmatrix} U^{-1} \\ \mathbf{1}^t \end{pmatrix} U && [\text{from (12)}] \\
&= \begin{pmatrix} I \\ \mathbf{1}^tU \end{pmatrix} \\
&= \begin{pmatrix} I \\ u^t \end{pmatrix} && (17) \\
\bar{M}^t &= \begin{pmatrix} I & u \end{pmatrix}. && [\text{from (15) and (17)}] \qquad (18)
\end{aligned}$$

By substituting (18) in (14), we conclude that the receiver computes the desired linear function by employing the network code defined by the encoding matrices $\{\bar{A}_i, i = 1, 2, \ldots, s\}$, $\bar{B}$, and F.

The proof of the theorem is now complete for the case when $T = (I\ u)$. If $T \sim (I\ u)$, then there exists a full-rank matrix Q and a column vector u' of non-zero elements over $\mathbb{F}_q$ such that

$$T = Q\ (I\ u'). \qquad [\text{from From Lemma A.1 in the Appendix}]$$

Since a full-rank linear operator preserves linear-independence among vectors, for every such full-rank matrix Q, we have

$$\text{rank}(T_{K_C}) = \text{ rank}\big((Q^{-1}T)_{K_C}\big) \quad \forall\ C \in \Lambda(\mathcal{N}). \tag{19}$$

Equation (19) implies that min-cut$(\mathcal{N}, T)$ = min-cut$\big(\mathcal{N}, Q^{-1}T\big)$. Since $Q^{-1}T = (I\ u')$, from the first part of the proof, there exist an $n > 0$ and coding matrices $A_\tau, \tau = 1, 2, \cdots, s$, F, and B over $\mathbb{F}_{q^n}$ such that the receiver can compute the linear target function corresponding to $(I\ u')$ if and only if min-cut$(\mathcal{N}, T) \geq 1$. It immediately follows that by utilizing a code corresponding to the coding matrices $A_\tau, \tau = 1, 2, \cdots, s$, F, and QB, the receiver can compute the target function corresponding to $Q(I\ u') = T$.

All that remains to be done is to provide proofs of claims 1 and 2.

Proof of Claim 1: If a cut C is such that $|K_C| \leq s-1$, then

$$\begin{aligned} |C| &\geq \text{ rank}(T_{K_C}) && \text{[from min-cut}(\mathcal{N}, T) \geq 1 \text{ and (4)]} \\ &= |K_C|\,. && \text{[from } T = (I\ u)\text{]} \end{aligned}$$

Thus by [10, Theorem 3.1], there exists a routing solution to compute the identity function of the sources $\{\sigma_i, i \in K_C\}$ at the receiver. Let $|K_C| = s-1$ and let $K_C = \{1, 2, \ldots, j-1, j+1, \ldots, s\}$ for some (arbitrary) j. By Lemma II.1, after fixing $\alpha(\sigma_j) = 0$, the vector received by ρ can be written as

$$M_{(j)}^t \begin{pmatrix} \alpha(\sigma_1) \\ \alpha(\sigma_2) \\ \vdots \\ \alpha(\sigma_{j-1}) \\ \alpha(\sigma_{j+1}) \\ \vdots \\ \alpha(\sigma_s) \end{pmatrix}.$$

The existence of a routing solution for computing the identity function guarantees that there exist $x_{e',e}, x_{e,l} \in \{0,1\}$ such that the matrix $M_{(j)}$ has a non-zero determinant over $\mathbb{F}_q$. It follows that the determinant of $M_{(j)}$ is non-zero over $\mathbb{F}_q[x_A, x_F, x_B]$. Since $j \in \{1, 2, \ldots, s\}$ was arbitrary in the above argument, it follows that the determinant of each $M_{(j)}, j = 1, 2, \ldots, s$ is non-zero over $\mathbb{F}_q[x_A, x_F, x_B]$ and the claim follows.

Proof of Claim 2: We have

$$\begin{aligned} M\ M_{(s)}^{-1} &= \begin{pmatrix} A_1(I-F)^{-1}B^t \\ A_2(I-F)^{-1}B^t \\ \vdots \\ A_s(I-F)^{-1}B^t \end{pmatrix} \quad M_{(s)}^{-1} \\ &\overset{(a)}{=} \begin{pmatrix} I \\ A_s(I-F)^{-1}B^t M_{(s)}^{-1} \end{pmatrix} \end{aligned} \tag{20}$$

where, (a) follows from the definition of $M_{(s)}^{-1}$. By contraction, assume that there exists an $i \in \{1, 2, \ldots, s-1\}$ such that $\left(A_s(I-F)^{-1}\bar{B}^t\right)_i = 0$. It then follows that

$$A_s(I-F)^{-1}B^t M_{(s)}^{-1} = \sum_{j=1}^{s-2} \left(A_s(I-F)^{-1}B^t M_{(s)}^{-1}\right)_{i_j} (A_{i_j}(I-F)^{-1}B^t M_{(s)}^{-1}) \qquad \text{[from (20)]} \tag{21}$$

for some choice of $i_j \in \{1, 2, \ldots, s-1\}, j = 1, 2, \ldots, s-2$ and

$$\left(A_s(I-F)^{-1}B^t - \sum_{j=1}^{s-2} \left(A_s(I-F)^{-1}B^t M_{(s)}^{-1}\right)_{i_j} (A_{i_j}(I-F)^{-1}B)^t\right) M_{(s)}^{-1} = 0 \qquad \text{[from (21)]}$$

$$\left(A_s(I-F)^{-1}B^t - \sum_{j=1}^{s-2} \left(A_s(I-F)^{-1}B^t M_{(s)}^{-1}\right)_{i_j} (A_{i_j}(I-F)^{-1}B)^t\right) = 0. \qquad \text{[from } M_{(s)}^{-1} \text{ is full rank]} \tag{22}$$

Equation (22) implies a linear dependence among $s-1$ rows of the matrix M. This contradicts the fact that for each $i = 1, 2, \ldots, s$, $M_{(i)}$ is full rank. Thus $\left(A_s(I-F)^{-1}B^t M_{(s)}^{-1}\right)_i \neq 0$ for $i = 1, 2, \ldots, s-1$ and the claim follows.

■

Remark III.3. We provide the following communication-theoretic interpretation of our method of proof above. We may view the computation problem as a MIMO (multiple input multiple output) channel where the multiple input is given by the vector of symbols generated by the sources, the output is the vector decoded by the receiver, and the channel is given by the network topology and the network code. Our objective is to choose a channel to guarantee the desired output, by way of code design subject to the constraints imposed by network topology. The channel gain from source σ_i to the receiver is given by the vector M_i of length $s-1$. The first part of the proof utilizes the sparse zeros lemma to establish that there exists a choice of channels such that the channel between every set of $s-1$ sources and the receiver is invertible. This is similar to the proof of the multicast theorem in [2]. In the second part of the proof, we recognize that the interference from different sources must also be "aligned" at the output for the receiver to be able to compute the desired function. Accordingly, we have modified the code construction to provide such alignment.

We now show the existence of a linear function that cannot be computed on a network satisfying the min-cut condition. This network will then be used as a building block to show an analogous result for a larger class of functions. Let T_1 denote the matrix

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \tag{23}$$

and let f_1 denote the corresponding linear function. It is possible to show with some algebra that $T_1 \not\sim (I\ u)$, for any column vector u of units, so that the conclusion of Theorem III.2 does not hold. Indeed, for the function f_1 the opposite conclusion is true, namely f_1 cannot be computed over $\mathcal{N}_1$ using linear codes. This is shown by the following Lemma.

Lemma III.4. *Let $\mathcal{N}_1$ be the network shown in Figure 1 with alphabet $\mathbb{F}_q$. We have*

1) *min-cut$(\mathcal{N}_1, T_1) = 1$.*
2) *There does not exist a linear solution for computing f_1 in $\mathcal{N}_1$.*

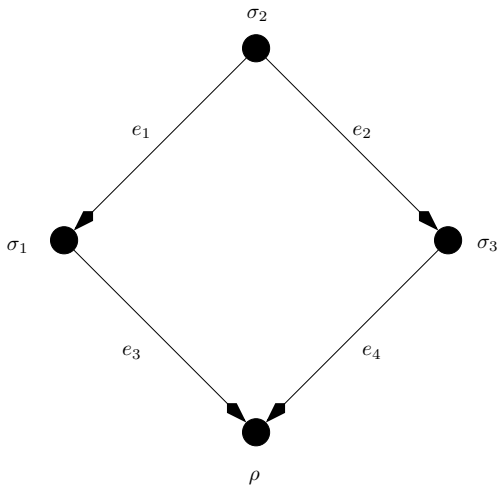

Fig. 1. Network on which there is no linear solution for computing f_1.

Proof: That min-cut$(\mathcal{N}_1, T_1) = 1$ is easily verified by considering the cut $C = \{e_3, e_4\}$ which attains the minimum. We now proceed to show, using Theorem II.3, that a linear solution does not exist.

We may assume, without loss of generality, that the node σ_2 sends its message directly to nodes σ_1 and σ_3 (i.e., $x_{1,e_1} = x_{1,e_2} = 1$). The matrices Z_1, Z_2, and Z_3 over R can then be written as

$$\begin{aligned}
(T_1)^t - M_1 &= \begin{pmatrix} (1 - x_{1,e_3}x_{e_3,1}) & (0 - x_{1,e_3}x_{e_3,2}) \end{pmatrix} \\
(T_2)^t - M_2 &= \begin{pmatrix} 0 - x_{e_1,e_3}x_{e_3,1} - x_{e_2,e_4}x_{e_4,1} \\ 1 - x_{e_1,e_3}x_{e_3,2} - x_{e_2,e_4}x_{e_4,2} \end{pmatrix}^t \\
(T_3)^t - M_3 &= \begin{pmatrix} (1 - x_{1,e_4}x_{e_4,1}) & (0 - x_{1,e_4}x_{e_4,2}) \end{pmatrix}.
\end{aligned}$$

Consequently, the ideal J is given by

$$\begin{aligned}
J = \big\langle &(1 - x_{1,e_3}x_{e_3,1}),\ (0 - x_{1,e_3}x_{e_3,2}), \\
&(0 - x_{e_1,e_3}x_{e_3,1} - x_{e_2,e_4}x_{e_4,1}), \\
&(1 - x_{e_1,e_3}x_{e_3,2} - x_{e_2,e_4}x_{e_4,2}), \\
&(1 - x_{1,e_4}x_{e_4,1}),\ (0 - x_{1,e_4}x_{e_4,2}) \big\rangle.
\end{aligned}$$

We have

$$\begin{aligned}
1 = &(1 - x_{e_1,e_3}x_{e_3,2} - x_{e_2,e_4}x_{e_4,2}) \\
&+ x_{e_1,e_3}x_{e_3,2}(1 - x_{1,e_3}x_{e_3,1}) \\
&- x_{e_1,e_3}x_{e_3,1}(0 - x_{1,e_3}x_{e_3,2}) \\
&+ x_{e_2,e_4}x_{e_4,2}(1 - x_{1,e_4}x_{e_4,1}) \\
&- x_{e_2,e_4}x_{e_4,1}(0 - x_{1,e_4}x_{e_4,2}) \ \in\ J.
\end{aligned}$$

Thus, it follows that $\mathcal{G}(J) = \{1\}$. By Theorem II.3, a linear solution does not exist for computing f_1 in $\mathcal{N}_1$. ■

We now identify a much larger class of linear functions for which there exist networks satisfying the min-cut condition but for which linear solutions do not exist. Let P be an $l \times s - l$ matrix with at least one zero element and $T \sim (I\ P)$. For each T in this equivalence class we show that there exist a network $\mathcal{N}$ that does not have a solution for computing the linear target function corresponding to T but satisfies the cut condition in Lemma II.4. The main idea of the proof is to establish that a solution for computing such a function in network $\mathcal{N}$ implies a solution for computing the function corresponding to T_1 in $\mathcal{N}_1$, and then to use Lemma III.4.

Theorem III.5. *Consider a linear target function f corresponding to a matrix $T \in \mathbb{F}_q^{l\times s}$. If $T \sim (I\ P)$ such that at least one element of P is zero, then there exists a network $\mathcal{N}$ such that*

1) *min-cut$(\mathcal{N}, T) = 1$.*
2) *There does not exist a linear solution for computing f in $\mathcal{N}$.*

Proof: Let $\hat{T} = (I\ P)$ and let $\hat{f}$ denote the corresponding linear target function. It is enough to show that there exists a network $\mathcal{N}_P$ such that min-cut$(\mathcal{N}_P, f) = 1$ but $\mathcal{N}_P$ does not have a linear solution for computing $\hat{f}$. This is because a network $\mathcal{N}$ that does not have a solution for computing T is easily obtained by renaming the sources in $\mathcal{N}_P$ as follows: Since $T \sim (I\ P)$, there exist Q and Π such that $T = Q(I\ P)\Pi$. Let κ denote the permutation function on the set $\{1, 2, \ldots, s\}$ defined by the permutation matrix Π^{-1}. Obtain the network $\mathcal{N}$ by relabeling source σ_i in $\mathcal{N}_P$ as $\sigma_{\kappa(i)}$. To see that there does not exist a solution for computing f in $\mathcal{N}$, assume to the contrary that a solution exists. By using the same network code in $\mathcal{N}_P$, the receiver computes

$$Q(I\ P)\Pi\ (x_{\kappa(1)}, x_{\kappa(2)}, \ldots, x_{\kappa(s)})^t = Q(I\ P)\ (x_1, x_2, \ldots, x_s)^t.$$

Thus the receiver in $\mathcal{N}_P$ can compute $\hat{T}x^t$, which is a contradiction.

Now we construct the network $\mathcal{N}_P$ as claimed. Since P has at least once zero element, there exists a $\tau \in \{l+1, l+2, \ldots, s\}$ such that $\hat{T}$ has a zero in τ-th column. Define

$$K = \left\{ i \in \{1, 2, \ldots, l\} : \hat{T}_{i,\tau} = 1 \right\}$$

Denote the elements of K by

$$\left\{ j_1, j_2, \ldots, j_{|K|} \right\}.$$

Let p be an element of $\{1, 2, \ldots, l\} - K$ (such a p exists from the fact that the τ-th column contains at least one zero) and define

$$\bar{K} = \{1, 2, \ldots, s\} - K - \{\tau, p\}$$

and denote the elements of $\bar{K}$ by

$$\left\{ j_{|K|+1}, j_{|K|+2}, \ldots, j_{s-|K|-2} \right\}.$$

Since $\hat{T}$ does not contain an all-zero column, $|K| > 0$. Now, let $\mathcal{N}_P$ denote the network shown in Figure 2 where, v denotes a relay node. It follows from the construction that

$$\begin{pmatrix} \hat{T}_{j_1,j_1} & \hat{T}_{j_1,p} & \hat{T}_{j_1,\tau} \\ \hat{T}_{p,j_1} & \hat{T}_{p,p} & \hat{T}_{p,\tau} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \tag{24}$$

which is equal to the transfer matrix T_1 defined in (23).

Notice that in the special case when $K = \{j_1\}$ and $\left|\bar{K}\right| = 0$, the network shown in Figure 2 reduces to the network shown in Figure 3 which is equivalent to the network $\mathcal{N}_1$ in Figure 1 with target function f_1. Since $\mathcal{N}_1$ does not have a solution for computing f_1 by Lemma III.4, we conclude that $\mathcal{N}_1$ cannot have a solution either.

Similarly, we now show that in the general case, if the network $\mathcal{N}_P$ has a solution for computing $\hat{f}$, then such a solution induces a solution for computing f_1 in network $\mathcal{N}_1$, contradicting Lemma III.4. Let there exist an $n > 0$ for which there is a linear solution for computing $\hat{f}$ over $\mathcal{N}_P$ using an alphabet over $\mathbb{F}_{q^n}$. In any such solution, for each $j \in K - \{j_1\}$, the encoding function on the edge (σ_j, ρ) must be of the form

$$\beta_{1,j}\alpha(\sigma_j) + \beta_{2,j}\alpha(\sigma_\tau) \tag{25}$$

for some $\beta_{1,j}, \beta_{2,j} \in \mathbb{F}_{q^n}$. Since (σ_j, ρ) is the only path from source σ_j to the receiver, it is obvious that $\beta_{1,j} \neq 0$.

We define the map α as follows. Let $\alpha(\sigma_{j_1}), \alpha(\sigma_p), \alpha(\sigma_\tau)$ be arbitrary elements of $\mathbb{F}_{q^n}$ and let

$$\alpha(\sigma_j) = \begin{cases} 0 & \text{for } j \in \bar{K} \\ -(\beta_{1,j})^{-1}\beta_{2,j}\alpha(\sigma_\tau) & \text{for } j \in K - \{j_1\}. \end{cases} \tag{26}$$

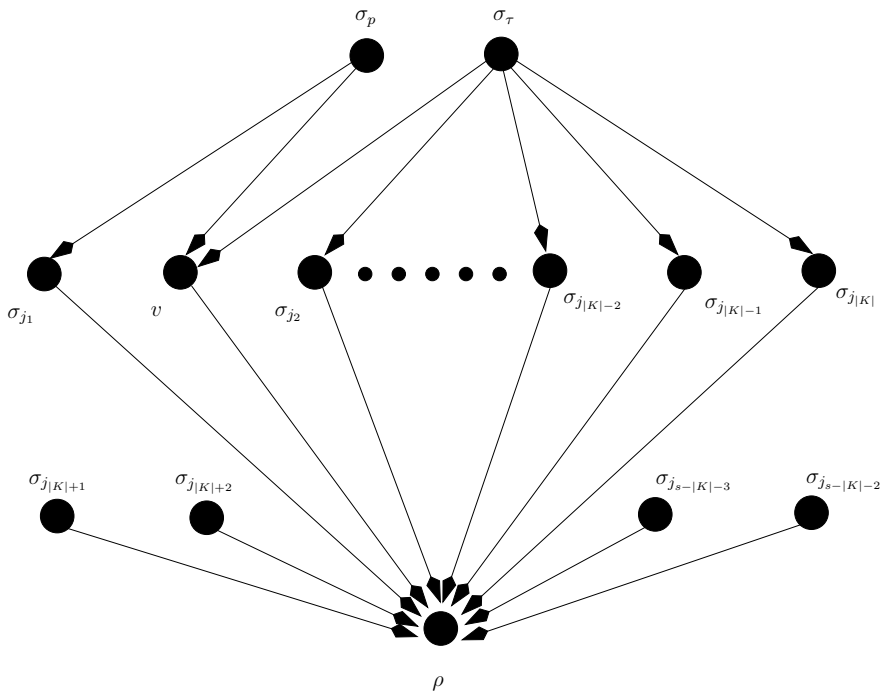

Fig. 2. Network $\mathcal{N}_P$ with min-cut 1 that does not have an $\mathbb{F}_q$-linear solution for computing $(I\ P)$.

Note that α has been chosen such that for any choice of $\alpha(\sigma_{j_1}), \alpha(\sigma_p)$, and $\alpha(\sigma_\tau)$, every edge $e \in \mathcal{E}_{in}(\rho) - \{(\sigma_{i_1}, \rho), (v, \rho)\}$ carries the zero vector. Furthermore, for the above choice of α, the target function associated with $\hat{T}$ reduces to

$$\left(\alpha(\sigma_1) + \hat{T}_{1,\tau}\alpha(\sigma_\tau), \alpha(\sigma_2) + \hat{T}_{2,\tau}\alpha(\sigma_\tau), \ldots, \alpha(\sigma_l) + \hat{T}_{l,\tau}\alpha(\sigma_\tau)\right). \tag{27}$$

Substituting $\hat{T}_{j_1,\tau} = 1$ and $\hat{T}_{p,\tau} = 0$ in (27), it follows that the receiver can compute

$$(\alpha(\sigma_{j_1}) + \alpha(\sigma_\tau), \alpha(\sigma_p))$$

from the vectors received on edges (σ_{i_1}, ρ) and (v, ρ). Consequently, it follows that there exist a linear solution over $\mathbb{F}_{q^n}$ for computing the linear target function associated with the transfer matrix

$$\begin{pmatrix} \hat{T}_{j_1,j_1} & \hat{T}_{j_1,p} & \hat{T}_{j_1,\tau} \\ \hat{T}_{p,j_1} & \hat{T}_{p,p} & \hat{T}_{p,\tau} \end{pmatrix}$$

in the network shown in Figure 3. It is easy to see that the existence of such a code implies a scalar linear solution for

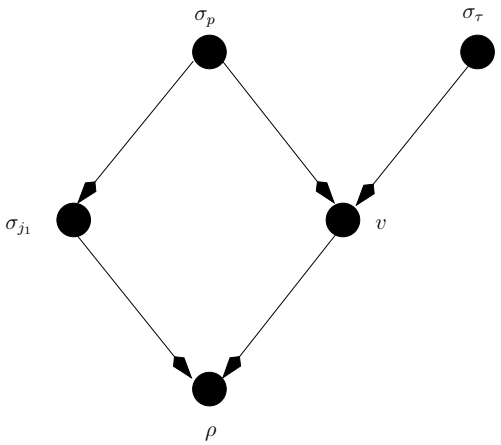

Fig. 3. Subnetwork of $\mathcal{N}_P$ used to show the equivalence between solving network $\mathcal{N}_P$ and solving network $\mathcal{N}_1$.

computing f_1 in $\mathcal{N}_1$. This establishes the desired contradiction.

Finally, we show that min-cut$(\mathcal{N}, T) = 1$. Let $C \in \Lambda(\mathcal{N})$ be a cut such that $K_C \subset K \cup \{p, \tau\}$ (i.e, C separates sources from only the top and middle rows in the network $\mathcal{N}_P$). We have the following two cases:

1) If $\sigma_\tau \notin K_C$, then it is easy to see that $|C| \geq |K_C|$. Similarly, if $\sigma_\tau \in K_C$ and $\sigma_p \notin K_C$, then again $|C| \geq |K_C|$. Consequently, we have

$$\begin{aligned} \frac{|C|}{\text{rank}(T_{K_C})} &\geq \frac{|C|}{|K_C|} && [\text{from } \text{rank}(T_{K_C}) \leq |K_C|] \\ &\geq 1. && [\text{from } |C| \geq |K_C|] \end{aligned} \tag{28}$$

2) If $\sigma_\tau \in K_C$ and $\sigma_p \in K_C$, then from Figure 3, $|C| = |K| + 1$ and $K_C = K \cup \{p, \tau\}$. Moreover, the index set K was constructed such that

$$\hat{T}_\tau = \sum_{i \in K} \hat{T}_{i,\tau} \hat{T}_i. \tag{29}$$

Consequently, we have

$$\begin{aligned} \text{rank}(T_{K_C}) &= \text{rank}\big(T_{K \cup \{p,\tau\}}\big) && [\text{from } K_C = K \cup \{p, \tau\}] \\ &\leq |K| + 1 && [\text{from (29)}] \\ &= |C|\,. \end{aligned} \tag{30}$$

From (28) and (30), we conclude that if $K_C \subset K \cup \{p, \tau\}$, then

$$\frac{|C|}{\text{rank}(T_{K_C})} \geq 1. \tag{31}$$

For an arbitrary cut $C \in \Lambda(\mathcal{N})$, let $c_{\bar{K}}$ denote the number of sources in $\bar{K}$ that are separated from the receiver by C (i.e, $c_{\bar{K}} = \big|K_C \cap \bar{K}\big|$). We have

$$\begin{aligned} \frac{|C|}{\text{rank}(T_{K_C})} &= \frac{|C| - c_{\bar{K}} + c_{\bar{K}}}{\text{rank}(T_{K_C})} \\ &\geq \frac{|C| - c_{\bar{K}} + c_{\bar{K}}}{\text{rank}\big(T_{K_C - \bar{K}}\big) + c_{\bar{K}}} \end{aligned} \tag{32}$$

Since each source in $\bar{K}$ is directly connected to the receiver, $|C| - c_{\bar{K}}$ is equal to the number of edges in C separating the sources in $K_C - \bar{K}$ from the receiver. Consequently, from (31), it follows that

$$\frac{|C| - c_{\bar{K}}}{\text{rank}\big(T_{K_C - \bar{K}}\big)} \geq 1. \tag{33}$$

Substituting (33) in (32), we conclude that for all $C \in \Lambda(\mathcal{N})$

$$\text{min-cut}(\mathcal{N}, T) \geq 1.$$

Since the edge $(\sigma_{j_{|K|+1}}, \rho)$ disconnects the source $\sigma_{j_{|K|+1}}$ from the receiver, min-cut$(\mathcal{N}, T) \leq 1$ is immediate and the proof of the theorem is now complete. ∎

We now consider the case in which the source alphabet is over the binary field. In this case, we have that the two function classes identified by Theorems III.2 and III.5 are complements of each other, namely either $T \sim (I\ \mathbf{1})$ or $T \sim (I\ P)$ with P containing at least one zero element.

Theorem III.6. *Let $l \notin \{1, s\}$ and let $T \in \mathbb{F}_2^{l \times s}$. If $T \nsim (I\ \mathbf{1})$, then there exists an $l \times (s-l)$ matrix P such that P has at least one zero element and $T \sim (I\ P)$.*

Proof: Since T is assumed to have a full row rank, $T \sim (I\ \bar{P})$ for some $l \times (s-l)$ matrix $(I\ \bar{P})$ over $\mathbb{F}_2$. If $\bar{P}$ has 0's, then we are done. Assume to the contrary that $\bar{P}$ is a matrix of non-zero elements. We only need to consider the case when $(s-l) > 1$ (since $T \nsim (I\ \mathbf{1})$). For $i = 1, 2, \ldots, l-1$, let $\phi^{(i)}$ denote the i-th column vector of the $l \times l$ identity matrix. Define $Q = (\phi^{(1)}\phi^{(2)} \cdots \phi^{(l-1)}\ \mathbf{1})$ and let Π be a permutation matrix that interchanges the l-th and $(l+1)$-th columns and leaves the remaining columns unchanged. It is now easy to verify that

$$\begin{aligned} Q\ (I\ \bar{P})\ \Pi &= (Q\ Q\bar{P})\ \Pi \\ &= (I\ P) \end{aligned} \tag{34}$$

where P is an $l \times s - l$ matrix with at least one 0 element: for $i \in \{1, 2, \cdots, l-1\}$

$$\begin{aligned} P_{i,2} &= (Q\bar{P})_{i,2} \\ &= (Q\mathbf{1})_i \\ &= 1 + 1 \\ &= 0. \end{aligned}$$

Thus, $(I\ \bar{P}) \sim (I\ P)$ and by transitivity we conclude that $T \sim (I\ P)$ which proves the claim. ■

IV. Conclusion

We wish to mention the following open problems arising from this work.

- Is there a graph-theoretic condition that allows to determine whether a given network is solvable with reference to a given linear function? We have provided an algebraic test in terms of the Groböner basis of a corresponding ideal, but we wish to know whether there is there an algorithmically more efficient test.
- We showed that min-cut$(\mathcal{N}, T) = 1$ is not sufficient to guarantee solvability for a certain class of linear functions. A possible direction of future research is to ask whether there is a constant c such that min-cut$(\mathcal{N}, T) \geq c$ guarantees solvability. Alternatively, for every constant c, does there exist a network $\mathcal{N}$ and a matrix T such that min-cut$(\mathcal{N}, T) \geq c$ and $\mathcal{N}$ does not have a linear solution for computing the linear target function associated with T?

APPENDIX

Lemma A.1. *Let $T \in \mathbb{F}_q^{l\times s}$. If $u \in \mathbb{F}_q^{s-1}$ is a column vector of non-zero elements and $T \sim (I\ u)$, then there exists a full rank matrix Q and a column vector u' of non-zero elements over $\mathbb{F}_q$ such that $T = Q\ (I\ u')$.*

Proof: Let Q denote the matrix obtained by collecting the first $(s-1)$ columns of T. We will first show that the matrix Q is full-rank. After factoring out Q, we then prove that the last column must have non-zero entries.

Since $T \sim (I\ u)$, there exists a full-rank matrix $\bar{Q}$ and a permutation matrix $\bar{\Pi}$ such that

$$\begin{aligned} T &= \bar{Q}\ (I\ u)\ \bar{\Pi} \\ &= (\bar{Q}\ \bar{Q}u)\ \bar{\Pi}. \end{aligned} \tag{35}$$

From (35), the columns of Q are constituted by the columns of $\bar{Q}$ in which case Q is full-rank, or columns of Q contains $(s-2)$ columns of $\bar{Q}$ and $\bar{Q}u$. We will now show that the vector $\bar{Q}u$ cannot be written as a linear combination of any set of $s-2$ column vectors of $\bar{Q}$. Assume to the contrary that there exist $a_j \in \mathbb{F}_q$ for $j \in \{1, 2, s-2\}$ such that

$$\bar{Q}u = \sum_{j=1}^{s-2} a_j \bar{Q}_j \tag{36}$$

where $\bar{Q}_j$ denotes the j-th column of $\bar{Q}$. Let a denote the vector such that $a_j = a_j, j = 1, 2, \ldots s-2$, and $a_{s-1} = 0$. We have

$$\begin{aligned} u - a &\neq 0 && [\text{from } u_{s-1} \neq 0 \text{ and } a_{s-1} = 0] \\ \bar{Q}(u-a) &= 0 && [\text{from (36)}]. \end{aligned} \tag{37}$$

(37) contradicts the fact that $\bar{Q}$ is full-rank. Hence a_i's satisfying (36) do not exist and consequently, Q is a full-rank matrix. We now have

$$T = Q(I\ u')$$

where $u' = Q^{-1}T_s$ and hence $T \sim (I\ u')$. Furthermore, $T \sim (I\ u)$ and $T \sim (I\ u')$ implies that $(I\ u) \sim (I\ u')$. Thus, there exists a full-rank matrix P and a permutation matrix Π such that

$$\begin{aligned} (I\ u) &= P\ (I\ u')\ \Pi \\ &= (P\ Pu')\ \Pi. \end{aligned} \tag{38}$$

Let $\phi^{(i)}$ denote the i-th column of I. It follows from (38) that either (a) $Pu' = u$ and P itself is an $(s-1)\times(s-1)$ permutation matrix, or (b) For some $j \in \{1, 2, \ldots, s-1\}$, j-th column of P is u, and the remaining columns must constitute the $s-2$ columns $\phi^{(1)}, \phi^{(2)}, \ldots, \phi^{(\tau-1)}, \phi^{(\tau+1)}, \phi^{(s-1)}$ of I for some τ. If (a) is true, then $u' = P^{-1}u$ and the elements of u' are non-zero since P^{-1} is another permutation matrix. If (b) is true, then $Pu' = \phi^{(\tau)}$ and it must be that $u'_j \neq 0$ (if $u'_j = 0$, then $(Pu')_\tau = 0$ which contradicts $Pu' = \phi^{(\tau)}$). Let $L = \{i : i \neq j, \text{ and } u'_i \neq 0\}$. We must have

$$\phi^{(\tau)} = u'_j u + \sum_{i\in D} u'_i\ \phi^{(j_i)}. \tag{39}$$

If we denote the number of non-zero entries in a vector u by $|u|$, then we have

$$\begin{aligned} 1 &= \left|\phi^{(\tau)}\right| \\ &\geq \left|u'_j u\right| - |D| && [\text{from (39)}] \\ &= (s-1) - |D| \\ &\geq 1 && [\text{from } |D| \leq s-2] \end{aligned} \tag{40}$$

From (40), it follows that $|D| = s-2$ and consequently that every element of u' is non-zero. The proof of the lemma is now complete. ■

REFERENCES

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

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

[3] H. Kowshik and P. R. Kumar, "Zero error function computation in sensor networks", *Proceedings of the IEEE Conference on Decision and Control*, 2009.

[4] B. K. Rai, and B. K. Dey, "Sum-networks: System of polynomial equations, unachievability of coding capacity, reversibility, insufficiency of linear network coding," http://arxiv.org/abs/0906.0695, 2009.

[5] A. Ramamoorthy, "Communicating the sum of sources over a network," *Proceedings of the IEEE International Symposium on Information Theory*, Toronto, Canada, 2008.

[6] N. Ma, P. Ishwar, and P. Gupta, "Information-theoretic bounds for multiround function computation in collocated networks," *Proceedings of the IEEE International Symposium on Information Theory*, pp. 2306–2310, 2009.

[7] B. Nazer and M. Gastpar, "Computing over multiple-access channels," *IEEE Transactions on Information Theory*, vol. 53, pp. 3498–3516, Oct. 2007.

[8] 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.
[9] R. Dougherty, C. Freiling, and K. Zeger, "Linear network codes and systems of polynomial equations", *IEEE Transactions on Information Theory* vol. 54, no. 5, pp. 2303-2316, May 2008.
[10] R. Appuswamy, M. Franceschetti, N. Karamchandani, and K. Zeger, "Network coding for computing: Cut-set bounds", *to appear in IEEE Transactions on Information Theory*, Feb. 2011.
[11] R. Appuswamy, M. Franceschetti, N. Karamchandani, and K. Zeger, "Network coding for computing: Linear codes", *submitted to IEEE Transactions on Information Theory*, 2010.
[12] J. T. Schwartz, "Fast probabilistic algorithms for verification of polynomial identities", *J. ACM.*, vol. 27, pp. 701-717, 1980.
[13] T.W. Hungerford, "Algebra", *Springer-Verlag*, 1997.