A Combinatorial Study of Linear Deterministic Relay Networks

S. M. Sadegh Tabatabaei Yazdi and Serap A. Savari

Abstract

In the last few years the so-called “linear deterministic” model of relay channels has gained popularity as a means of studying the flow of information over wireless communication networks, and this approach generalizes the model of wireline networks which is standard in network optimization. There is recent work extending the celebrated max-flow/min-cut theorem to the capacity of a unicast session over a linear deterministic relay network which is modeled by a layered directed graph. This result was first proved by a random coding scheme over large blocks of transmitted signals. We demonstrate the same result with a simple, deterministic, polynomial-time algorithm which takes as input a single transmitted signal instead of a long block of signals. Our capacity-achieving transmission scheme for a two-layer network requires the extension of a one-dimensional Rado-Hall transversal theorem on the independent subsets of rows of a row-partitioned matrix into a two-dimensional variation for block matrices. To generalize our approach to larger networks we use the submodularity of the capacity of a cut for our model and show that our complete transmission scheme can be obtained by solving a linear program over the intersection of two polymatroids. We prove that our transmission scheme can achieve the max-flow/min-cut capacity by applying a theorem of Edmonds about such linear programs. We use standard submodular function minimization techniques as part of our polynomial-time algorithm to construct our capacity-achieving transmission scheme.

1 Introduction

Network information theory [7, Ch. 15] attempts to model aspects of large communication networks such as interference, cooperation, and noise that are often overlooked in network optimization theory. Relay channels [7, §15.7] are an example of a network information problem in which there is a source, a unique destination, and at least one intermediary transmitter-receiver pair which is instrumental to the communication between the source and the destination. In this paper we focus on a simplified mathematical model for the wireless relay channel which we describe via a directed graph \( \mathcal{N}(\mathcal{V}, \mathcal{E}) \), where \( \mathcal{V} \) denotes the set consisting of the source node, the destination node, and all relay nodes. Here the source node only sends signals, the destination node only receives signals, and a relay node can both receive signals and transmit any function of its incoming messages. The transmission of signals in this network obeys two rules: 1- any signal which is sent by node \( i \) is broadcast to every node \( j \) such that \( ij \in \mathcal{E} \), and 2- the signal which is received by node \( j \) is a linear combination of all signals that are broadcast to it and an independent additive noise signal which is typically modeled by a Gaussian random variable. Finding the capacity of a wireless relay channel has long been a challenging and important open problem. Avestimehr, Diggavi, and Tse [6, 5] recently proposed a simplified linear deterministic relay network model in which the effects of broadcasting, interference,
and noise are captured by linear transformations of the transmitted signals. One motivation for the study of this model is the result in [6, 5] that the capacity of any wireless relay channel with Gaussian noise is within a constant additive factor of the capacity of a corresponding linear deterministic model. More recent work [11, 19] has connected the linear deterministic model to the approximate capacity of other relay channels and to the design of near-optimal coding schemes for them. We next summarize the model and the results of [6, 5]:

The authors of [6, 5] focus on layered directed graphs \( \mathcal{N} \) with set of nodes \( \mathcal{V} = \mathcal{O}_1 \cup \mathcal{O}_2 \cup \cdots \cup \mathcal{O}_M \). Here \( \mathcal{O}_1 = \{ \mathcal{O}_1(1) \} = \{ S \} \) and \( \mathcal{O}_M = \{ \mathcal{O}_M(1) \} = \{ D \} \) respectively denote the source and destination nodes, and \( \mathcal{O}_i = \{ \mathcal{O}_i(1), \cdots, \mathcal{O}_i(m_i) \} \) denotes the set of relay nodes in the \( i \)th layer for \( i \in \{ 2, \cdots, M - 1 \} \). Each edge in the graph is from some node in \( \mathcal{O}_i, i \in \{ 1, \cdots, M - 1 \} \), to a node in \( \mathcal{O}_{i+1} \). Observe that the study of an arbitrary directed network can also be placed into this framework if one instead studies its time-expanded representation [6, 5]. In this case \( \mathcal{O}_i \) corresponds to the possible behaviors of the network from symbol time \((i - 1)\tau\) to symbol time \(i\tau - 1\) for some positive integer \( \tau \).

In the layered network, every node \( \mathcal{O}_i(j), i \in \{ 2, \cdots, M - 1 \} \), receives a column vector \( y_i^t(j) \) at time \( t \) and transmits a column vector \( x_i^t(j) \) at time \( t \). Node \( S \) transmits vector \( x_S^t = x_1^t \) and node \( D \) receives vector \( y_M^t = y_{M+1}^t \). The elements of all transmitted and received vectors belong to some fixed finite field \( \mathbb{F}_q \). Every edge \( \mathcal{O}_i(j) \mathcal{O}_{i+1}(k) \in \mathcal{E} \) represents a communication channel which is described by a matrix \( G_i(j, k) \) with entries from \( \mathbb{F}_q \) called the transfer function from \( x_j^t(k) \) to vector \( y_{i+1}^t(j) \). Define the block matrix \( G_i \) by \( G_i = [ G_i(j, k) \] for \( j \in \{ 1, \cdots, m_i \} \) and \( k \in [ m_i ] \). Let \( x_i^t \) be the transmitted vector from layer \( i \) and \( y_{i+1}^t \) be the received vector at layer \( i + 1 \); i.e.,

\[
x_i^t = \begin{bmatrix} x_i^t(1) \\ \vdots \\ x_i^t(m_i) \end{bmatrix} \quad \text{and} \quad y_{i+1}^t = \begin{bmatrix} y_{i+1}^t(1) \\ \vdots \\ y_{i+1}^t(m_{i+1}) \end{bmatrix}.
\]

The communication channel from layer \( i \) to layer \( i + 1 \) is characterized by the following relationship:

\[
y_{i+1}^t = G_i \cdot x_i^t.
\]  

(1)

Assume the communication session begins at time \( t \) and ends at time \( \tau \) and has desired rate of transmission \( R \); i.e., node \( S \) wishes to send message \( \omega \) which is chosen randomly from a set \( \{ 1, \cdots, q^R \} \) of messages to node \( D \) at the end of the session. The communication protocol proceeds from layer to layer. Node \( \mathcal{O}_i(j) \) transmits vectors \( x_j^t(j) \), \( i \in \{ 1, \cdots, m_i \} \) to nodes in layer \( \mathcal{O}_{i+1} \), and node \( \mathcal{O}_{i+1}(k) \) transmits signals to the nodes in next layer after receiving vectors \( y_{i+1}^t(k) \), \( i \in \{ 1, \cdots, m_{i+1} \} \). At every time instant \( t \), vector \( x_S^t \) is some function of \( \omega \) and \( x_i^t(j) \) is some function of \( y_i^t(j) \). There are two natural questions about this model: First, what is the capacity or maximum rate of information in this network? Second, among capacity-achieving schemes, how can one optimize the duration \( \tau \) and the complexity of the relay functions used?

The first of these questions was initially addressed in [6, 5]. To study the capacity \( C \) of the network, we first define a cut \( \Omega \) as a subset of the nodes \( \mathcal{V} \). A cut separates \( S \) from \( D \) if \( S \in \Omega \) and \( D \in \Omega^c = \mathcal{V} \setminus \Omega \). The transfer function of the cut \( \Omega, G(\Omega) \), is defined as a block diagonal matrix with \((M - 1) \times (M - 1)\) blocks. The \( i \)th diagonal block, \( G_i(\Omega) \) is the submatrix of \( G_i \) consisting of the transfer functions from the transmitted vectors of the nodes in \( \Omega \cap \mathcal{O}_i \) to the received vectors of the nodes in \( \Omega \cap \mathcal{O}_{i+1} \) for \( i \in [ M - 1 ] \), and each off-diagonal block is an all-zero matrix. Avestimehr, Diggavi and Tse used a min-cut upper bound on the rate of transmissions [7, Thm. 15.10.1] and a random coding argument to show

**Theorem 1.** For any cut \( \Omega \), \( C(\Omega) = \text{rank}(G(\Omega)) = \sum_{i=1}^{M-1} \text{rank}(G_i(\Omega)) \). Furthermore, the capacity of network \( \mathcal{N} \) as defined above is \( C = \min_{\Omega \text{ separates } S \text{ and } D} C(\Omega) \).
The achievability argument in [6, 5] is based on a linear scheme in which source node $S$ initially encodes the message $\omega$ as a vector in $\mathbb{F}_q^R$ denoted by $y_S(\omega)$. Node $S$ and relay node $O_i(j)$ respectively generate transmitted signals by the linear transformations

$$\begin{bmatrix} x_S^1 \\ \vdots \\ x_S^T \end{bmatrix} = F_S \cdot y_S(\omega) \quad \text{and} \quad \begin{bmatrix} x_i^1(j) \\ \vdots \\ x_i^T(j) \end{bmatrix} = F_i(j) \cdot \begin{bmatrix} y_i^1(j) \\ \vdots \\ y_i^T(j) \end{bmatrix}.$$ 

It is shown in [6, 5] that if the encoding matrices $F_S$ and $F_i(j)$ are chosen randomly with a uniform distribution over the space of all matrices over the field $\mathbb{F}_q$, if $\tau$ is sufficiently large, and if $R \leq C$, then the destination node $D$ will, with probability approaching 1, receive $\tau R$ linearly independent linear combinations of the message vector $y_S(\omega)$ from which it will be able to decode message $\omega$.

Since the complexity of the transmission scheme in [6, 5] is large and increases with $\tau$, we seek a deterministic, low-complexity transmission scheme that is capacity–achieving and processes only one signal $x_i^j(j)$ at a time for each $i$ and $j$, i.e., $\tau = 1$. We remark that [4] considered similar issues for transmissions over a binary field. We will discuss the approach of [4] in Section 1.2.

1.1 Our Results and Techniques

Our algorithm has two main steps. First we propose an algorithm to transmit signals from layer $O_i$, $i \in \{1, \ldots, M-1\}$, to layer $O_{i+1}$ in an optimal way. In the second step we extend our algorithm to the full network and prove that it is capacity–achieving. Since our transmission scheme manipulates only one signal $x_i^j(j)$ at a time for each $i$ and $j$, we hereafter drop the time superscript.

We define a flow of the block matrix $G_i$ as follows:

**Definition 2.** Let $d_i = (\ell_i(1), \ldots, \ell_i(m_i); \ell_{i+1}(1), \ldots, \ell_{i+1}(m_{i+1}))$ be a vector of non-negative integers that satisfies $\sum_{j=1}^{m_i} \ell_i(j) = \sum_{j=1}^{m_{i+1}} \ell_{i+1}(j) = R_d$. We say that matrix $G_i$ supports flow $d_i$ if there exists a full rank $R_d \times R_d$ submatrix $G_{d_i}$ of $G_i$ such that $G_{d_i}$ is an intersection of $\ell_i(j)$ columns of the $j$th column block of $G_i$, $j \in [m_i]$, with $\ell_{i+1}(k)$ rows of the $k$th row block of $G_i$, $k \in [m_{i+1}]$. (See Figure 1.(a).) We further say that such a submatrix $G_{d_i}$ is a solution for flow $d_i$.

For the physical interpretation of flow, suppose matrix $G_i$ supports flow $d_i$. Consider the subvector $x_{d_i}$ of $x_i$ and the subvector $y_{d_{i+1}}$ of $y_{i+1}$ which correspond to the transfer matrix $G_{d_i}$. Furthermore, let $x_{d_i}(j)$ and $y_{d_{i+1}}(k)$ respectively denote the parts of these subvectors that belong to vector $x_i(j)$ and $y_{i+1}(k)$. If the entries of $x_i$ which are not part of $x_{d_i}$ are set to zero, then $y_{d_{i+1}}$ will uniquely determine $x_{d_i}$ since $G_{d_i}$ is a full rank matrix. Hence $R_d$ units of information flow from the nodes in $O_i$ to the nodes in $O_{i+1}$ during a transmission. We next introduce the notion of the flow from $O_1$ to $O_M$ supported by network $\mathcal{N}$. For convenience we will consider a more general network $\mathcal{N}$ with an arbitrary number of nodes in the first and last layers as opposed to the single node each at the first and last layers of relay channel models.

**Definition 3.** Suppose non-negative integers $\ell_1(1), \ldots, \ell_1(m_1)$, and $\ell_M(1), \ldots, \ell_M(m_M)$ satisfy $\sum_{j=1}^{m_1} \ell_1(j) = \sum_{j=1}^{m_M} \ell_M(j) = R$. We say that vector $d = (\ell_1(1), \ldots, \ell_1(m_1); \ell_M(1), \ldots, \ell_M(m_M))$ is a rate-$R$ flow supported by network $\mathcal{N}$ if for every $i \in \{2, \ldots, M-1\}$ there exists non-negative integers $\ell_i(1), \ldots, \ell_i(m_i)$ such that vector $d_j = (\ell_j(1), \ldots, \ell_j(m_j); \ell_{j+1}(1), \ldots, \ell_{j+1}(m_{j+1}))$, $j \in [M-1]$, is a rate-$R$ flow supported by matrix $G_j$. (See Figure 1.(b).)

Every flow for network $\mathcal{N}$ is determined by the submatrices $G_{d_i}$ and the corresponding row and column indices of $G_i$. Let us return to the case where $\mathcal{N}$ has a single node each in the first and last layers. Suppose that network $\mathcal{N}$ supports a rate-$R$ flow $d = (R; R)$. Then given $G_{d_i}, i \in [M-1]$, a simple coding scheme that achieves rate $R$ can be defined as follows:
Figure 1: (a) An example of a matrix flow for vector \( \mathbf{d}_1 = (2, 2; 1, 3) \) in a matrix \( G_i \) with four blocks. Each small square is an entry of \( G_i \) and matrix \( G_{\mathbf{d}_1} \) is the intersection of dashed rows with dashed columns. (b) An example of network flow \( \mathbf{d} = (3; 3) \) in a network with four layers. The solid part of vector \( y_i(j) \) denotes the entries of \( y_{\mathbf{d}_1}(j) \) and the dashed part of vector \( x_i(j) \) denotes the entries of \( x_{\mathbf{d}_1}(j) \). The flow vectors are \( \mathbf{d}_1 = (3 ; 1, 2) \), \( \mathbf{d}_2 = (1, 2; 2, 1) \), and \( \mathbf{d}_3 = (2, 1; 3) \).

Transmission Scheme:

Given the length-\( R \) encoded vector \( y_S(\omega) \), node \( S \) generates vector \( x_S = x_1 \) by setting \( x_{\mathbf{d}_1} \) to the vector \( y_S(\omega) \) and by setting the other entries of \( x_S \) to zero. The transformation at every relay node \( O_i(j) \) is similar: after receiving vector \( y_i(j) \), node \( O_i(j) \) extracts the subvector \( y_{\mathbf{d}_1}(j) \) with length \( \ell_i(j) \) and sets \( x_{\mathbf{d}_1}(j) = y_{\mathbf{d}_1}(j) \). The remaining entries of \( x_i(j) \) are set to zero. Finally node \( D \) first decodes subvector \( y_{d_M} \) from the received vector \( y_D = y_M \) and then extracts the encoded message \( y_S(\omega) \). Observe that for every \( i \), \( x_{\mathbf{d}_1} = G^{-1}_{\mathbf{d}_1} \cdot y_{d_{i+1}} \) and \( x_{\mathbf{d}_1} = y_{\mathbf{d}_1} \). These imply that \( y_S(\omega) = G^{-1}_{\mathbf{d}_1} G^{-1}_{\mathbf{d}_2} \cdots G^{-1}_{d_{M-1}} y_{d_M} \). Since the matrices \( G_{\mathbf{d}_1} \) are nonsingular, the decoding operation is well defined.

Our main technical result in Section 2 is the following theorem providing the necessary and sufficient conditions for matrix \( G_i \) to support a flow \( \mathbf{d}_1 \):

**Theorem 4.** For any subsets \( U \subseteq [m_i] \) and \( W \subseteq [m_{i+1}] \) define the block matrix \( G_i(W, U) \) as the intersection of the row blocks of \( G_i \) having indices in \( W \) with the column blocks of \( G_i \) having indices in \( U \). Matrix \( G_i \) supports a flow \( \mathbf{d}_1 \) if and only if for all \( U \subseteq [m_i] \) and for all \( W \subseteq [m_{i+1}] \)

\[
\text{rank}(G_i(W, U)) \geq \sum_{j \in U} \ell_i(j) + \sum_{k \in W} \ell_{i+1}(k) - R_{\mathbf{d}_1}.
\]

This combinatorial property of matrices is, to our knowledge, the first two-dimensional result of this type and may be of independent interest in the theory of matrices. Theorem 4 holds for matrices with entries from an arbitrary field and is therefore more general than its application for this relay problem. We prove the necessity of Theorem 4 by examining the relationships of the ranks of various submatrices of block matrix \( G_{\mathbf{d}_1} \).

Our sufficiency argument is more technical and involves a divide-and-conquer procedure and an inductive argument to prove the existence of \( G_{\mathbf{d}_1} \). The basis of our inductive argument will be the Rado–Hall transversal theorem [25, Ch. 7], which states the necessary and sufficient conditions for the existence of independent structures in a collection of subsets of elements of a matroid. In the special case of matrices we have:
Theorem 5. (The Rado–Hall Theorem) Let $H$ be a matrix with column blocks labeled $1, \ldots, m$. For $U \subseteq [m]$ let $H(U)$ denote the submatrix of $H$ which is formed by the column blocks with indices in $U$. Given non-negative integers $\ell_1, \ldots, \ell_m$, there are $\ell_1 + \cdots + \ell_m$ linearly independent columns of $H$ with exactly $\ell_i$ of the columns from $H(\{i\})$, $i \in [m]$, if and only if for every $U$ we have $\text{rank}(H(U)) \geq \sum_{i \in U} \ell_i$.

In the study of matrices Theorem 4 can be viewed as an extension of the Rado–Hall Theorem. The Rado–Hall Theorem has many variations for matroids with different constraints on the cardinalities of the independent sets [18]. However, the rank function of a matroid is, by definition, both submodular and nondecreasing. The columns of a matrix form a ground set for a matroid, and the rank function of a subset of columns is defined as the dimension of the subspace spanned by these columns; this is also true if the columns are replaced by the rows of the matrix. We next define a natural extension of the rank function on the union of all rows and columns of a matrix which is motivated by the cuts of a network. Let $P$ and $Q$ respectively denote the set of rows and columns of the matrix. Define the rank function of a subset $T \subseteq P \cup Q$ where $T = P \cup Q$ with $P \subseteq P$ and $Q \subseteq Q$, to be the dimension of the spanning subspace of the submatrix which is formed by the intersection of the rows in $P$ and the columns which are not in $Q$. In Lemma 18 we study this rank function for the block matrices of network $N$ and prove that it is submodular. In that discussion the nodes of the two consecutive layers of the network respectively represent the column and row blocks of a matrix. We comment that the rank function arising from the study of cuts in our model has an important difference from the rank function in earlier transversal theorems. Observe that since the rank of both sets $\emptyset$ and $P \cup Q$ is zero, the rank function we introduce is not monotone. Korte and Lovász [15] initiated the study of a generalization of matroids known as greedoids with rank functions which are monotone but not necessarily submodular. They have also derived a transversal theory of greedoids similar to the Rado–Hall Theorem [15]. Theorem 4 appears to be the first extension of the Rado–Hall Theorem in which the rank function is submodular but not monotone. This is interesting because monotonicity plays a central role in the proofs of the previous results.

In Section 3, we prove the following extension of Theorem 4:

Theorem 6. A network $N$ with capacity $C$ supports flow $d = (R; R)$ if and only if $R \leq C$.

Edmonds and Giles [10] introduced a generalization of classical network flow known as submodular flow, where the classical flow conservation constraints are replaced by submodular flow constraints on certain subsets of nodes. We prove a submodularity property of the cut function in Lemma 19, and it is possible to show that our notion of flow for network $N$ is a special case of submodular flow. The Edmonds and Giles theory of submodular flow does not directly imply variations of the max–flow min–cut theorems. We therefore study our flow using an earlier result of Edmonds [9]. We prove in Section 3 that the maximum flow in our setting is a linear programming optimization over the intersection of two polymatroids. We apply a corollary of the polymatroid intersection theorem [9] to show that the maximum rate of a flow is the capacity $C$ of the network and that the corresponding flow can be achieved by non-negative integer vectors $d_i$.

In the appendix, we demonstrate an algorithm to construct a capacity–achieving code for network $N$ which is strongly polynomial time in the size of the graph and in the size of the matrices $G_i$. In first step of the analysis we show that:

Theorem 7. Given matrix $G_i$ and an achievable flow vector $d_i$, the submatrix $G_{d_i}$ can be computed in polynomial time.

We can in principle use the divide–and–conquer argument for the proof of Theorem 4 to obtain a recursive algorithm for this problem, but since the analysis is difficult we take a different approach.
We will demonstrate that testing the conditions of Theorem 4 for matrix $G_i$ and flow $d_i$ is equivalent to showing that a certain submodular function has a non-negative minimum. It is well known (see, e.g., [23, 14]) that there are polynomial-time algorithms to find the minimum of a submodular function. Our algorithm to construct $G_{d_i}$ checks which rows of $G_i$ can be removed without violating the conditions of Theorem 4 and then removes them one by one. The same procedure is next repeated for the columns. The final part of this step is to establish that the remaining matrix is a valid choice for $G_{d_i}$.

The second step establishes the following result.

**Theorem 8.** The capacity $C$ of a deterministic relay network $N$ can be computed in polynomial time. Given the network flow vector $d = (R; R)$ with $R \leq C$, the flow vectors $d_i$ for every matrix $G_i$, $i \in \{1, \cdots, M - 1\}$, can be computed in polynomial time.

We prove this theorem in the appendix by applying the algorithm in [17] for optimizing a linear function over the intersection of two polytopes. Theorems 7 and 8 together imply a polynomial-time algorithm for finding a transmission scheme for network $N$.

### 1.2 Related Work

Earlier work [3], [22] obtained capacity results for a different type of deterministic relay network in which the nodes broadcast data but the signals are received without interference. The paper [4] considers the same problem we address here, but restricts $\mathbb{F}_q$ to a binary field. The approach of [4] is based on a path augmentation argument similar to the Ford-Fulkerson algorithm (see, e.g., [12]) and involves a new network in which every node $O_i(j)$ is replaced by several nodes with each corresponding to a different entry of vector $x_i(j)$ or $y_i(j)$. In the new network there is an edge between a pair of nodes if the analogous entry in the transfer function $G_i(k, j)$ is equal to one. For an edge $e$, we respectively denote its tail and head by $x(e)$ and $y(e)$. Edges $e_1, \cdots, e_K$ are said to be independent if the transfer function from the vector $(x(e_1), \cdots, x(e_K))$ to the vector $(y(e_1), \cdots, y(e_K))$ has full rank. The scheme in [4] finds $C$ disjoint paths in the new network, where every cut that separates $S$ from $D$ intersects $C$ independent edges of these paths. There does not appear to be a natural way to extend the approach of [4] to arbitrary finite fields. We use a different procedure to construct the full rank submatrices needed for our transmission scheme.

**Remark.** The missing proofs in the body of the paper can be found in the appendix.

### 2 Proof of Theorem 4

At several steps of our proof we remove rows or columns from matrix $G_i$ to find a suitable submatrix $G_{d_i}$. Unless otherwise stated, assume that each such intermediate submatrix $T$ of $G_i$ maintains the same partition of row blocks and column blocks as original matrix $G_i$. In other words, each row (column) of $T$ consists of a subset of the entries of some row (column) of $G_i$, and the two rows (columns) have the same row (column) block index in their respective matrices. For $U \subseteq [m_i]$ and $W \subseteq [m_{i+1}]$, let the block submatrix $T(W, U)$ denote the intersection of the row blocks of $T$ having indices in $W$ with the column blocks of $T$ having indices in $U$. Notice that some row and/or column blocks of $T$ may be empty.

#### 2.1 Proof of Necessity

For $U \subseteq [m_i]$ and $W \subseteq [m_{i+1}]$, $G_{d_i}(W, U)$ is a submatrix of $G_i(W, U)$. Therefore, $\text{rank}(G_i(W, U)) \geq \text{rank}(G_{d_i}(W, U))$. From the submodularity of the rank function we have:
\[ \text{rank}(G_{d_i}([m_{i+1}], U)) \leq \text{rank}(G_{d_i}(W, U)) + \text{rank}(G_{d_i}([m_{i+1}] \setminus W, U)). \quad (3) \]

Since \( G_{d_i} \) is a full rank square matrix all of its columns are independent and hence
\[ \text{rank}(G_{d_i}([m_{i+1}], U)) = \sum_{j \in U} \ell_i(j). \quad (4) \]

By the monotonicity of the rank function
\[ \text{rank}(G_{d_i}([m_{i+1}] \setminus W, U)) \leq \text{rank}(G_{d_i}([m_{i+1}] \setminus W, [m_j])). \quad (5) \]

Since all of the rows of \( G_{d_i} \) are independent we have
\[ \text{rank}(G_{d_i}([m_{i+1}] \setminus W, [m_j])) = \sum_{k \in [m_{i+1}] \setminus W} \ell_{i+1}(k) = R_{d_i} - \sum_{k \in W} \ell_{i+1}(k). \quad (6) \]

The relations (3)-(6) imply the necessity of the condition
\[ \text{rank}(G_i(W, U)) \geq \sum_{j \in U} \ell_i(j) + \sum_{k \in W} \ell_{i+1}(k) - R_{d_i}. \]

### 2.2 Proof of Sufficiency

Assume throughout this subsection that the conditions of Theorem 4 are satisfied. First suppose that \( m_{i+1} = 1 \) and \( W = \{1\} \). Then for every set \( U \subseteq [m_j] \) the inequality (2) reduces to:
\[ \text{rank}(G_i(\{1\}, U)) \geq \sum_{j \in U} \ell_i(j) + \ell_{i+1}(1) - R_{d_i}. \]

Since the definition of vector \( d_i \) implies \( \ell_{i+1}(1) = R_{d_i} \), it follows that \( \text{rank}(G_i(\{1\}, U)) \geq \sum_{j \in U} \ell_i(j) \).

By Theorem 5, there exists a rank–\( R_{d_i} \) submatrix \( \tilde{G}_i \) of matrix \( G_i \) which consists of a subset of \( \ell_i(j) \) columns, for \( j \in [m_j] \), from each column block \( G_i(\{1\}, \{j\}) \). Since \( \text{rank}(\tilde{G}_i) = R_{d_i} \), \( \tilde{G}_i \) has a submatrix \( G_{d_i} \) consisting of \( R_{d_i} \) independent rows. \( G_{d_i} \) is a solution for flow \( d_i \).

We can similarly argue the existence of a solution \( G_{d_i} \) for flow \( d_i \) when \( m_i = 1 \). Next suppose \( m_i \geq 2 \) and \( m_{i+1} \geq 2 \). We will use induction and a divide-and-conquer procedure to prove the sufficiency result. For our inductive hypothesis we assume that Theorem 4 is true for any block matrix \( G_i \) consisting of \( n_{i+1} \times n_i \) blocks where \( n_i \leq m_i \), \( n_{i+1} \leq m_{i+1} \), and \( m_{i+1} \times m_i \neq n_{i+1} \times n_i \).

Let \( U \subseteq [m_i] \) and \( W \subseteq [m_{i+1}] \). We say that \( G_i(W, U) \) is a tight submatrix of \( G_i \) if the inequality (2) holds with equality for \( U \) and \( W \).

**Lemma 9.** Either \( G_i \) has a tight submatrix or \( G_i \) has a submatrix \( T \) for which for all \( \hat{U} \subseteq [m_i] \) and \( \hat{W} \subseteq [m_{i+1}] \),
\[ \text{rank}(T(\hat{W}, \hat{U})) \geq \sum_{j \in \hat{U}} \ell_i(j) + \sum_{k \in \hat{W}} \ell_{i+1}(k) - R_{d_i}. \quad (7) \]

and \( T \) has a tight submatrix; i.e., (7) hold with equality for some \( \hat{U} \subseteq [m_i] \) and \( \hat{W} \subseteq [m_{i+1}] \). In the latter case we replace \( G_i \) with \( T \) for the rest of the proof of Theorem 4.

By our previous argument, \( G_i \) has one or more tight submatrices. We call a tight submatrix \( G_i(W, U) \) a proper submatrix if it is not one of the following cases:
1. $|U| = m_i$ and $|W| = 1$, or
2. $|U| = 1$ and $|W| = m_{i+1}$.

For the rest of the proof of Theorem 4 we need a proper submatrix. We have the following result:

**Lemma 10.** Either (a) $G_i$ has a proper submatrix or (b) it has a submatrix $T$ which satisfies (7) for all $U \subseteq [m_i]$ and $W \subseteq [m_{i+1}]$ and $T$ has a proper submatrix $T(W,U)$. If case (b) applies then we replace $G_i$ with the corresponding submatrix $T$ for the rest of the proof of Theorem 4.

Let $P = G_i(W,U)$ be a proper submatrix of $G_i$. Next we reorder and relabel the row blocks and the column blocks of $G_i$ such that $P = G_i([m_{i+1}], [m_i])$. It is straightforward to reverse the ordering and relabeling operation at the end of our argument. We label the four parts of matrix $G_i$ as the following, where $A, B$, and/or $L$ may possibly be empty matrices:

$$G_i = \begin{bmatrix} P & A \\ B & L \end{bmatrix}.$$ 

We denote by $G_A$ the matrix $[ P \ A ]$. We further consider a partition of $G_A$ into blocks as the following:

$$G_A = \begin{bmatrix} G_i(1,[m_i]) & G_i(1,n_i+1) & \cdots & G_i(1,m_i) \\ & \vdots & \ddots & \vdots \\ G_i(n_{i+1},[m_i]) & G_i(n_{i+1},n_i+1) & \cdots & G_i(n_{i+1},m_i) \end{bmatrix}.$$ 

Next consider the vector

$$d_A = (\text{rank}(P), \ell_i(n_i+1), \ell_i(n_i+2), \ldots, \ell_i(m_i); \ell_{i+1}(1), \ldots, \ell_{i+1}(n_{i+1})).$$

We verify that $d_A$ is a valid flow vector. By the tightness of matrix $P$ we have:

$$\text{rank}(P) = \sum_{j=1}^{n_{i+1}} \ell_{i+1}(j) + \sum_{k=1}^{n_i} \ell_i(k) - R_{d_i} = \sum_{j=1}^{n_{i+1}} \ell_{i+1}(j) - \sum_{k=n_{i+1}}^{m_i} \ell_i(k).$$

Therefore we have

$$\text{rank}(P) + \sum_{j=n_{i+1}}^{m_i} \ell_i(j) = \sum_{k=1}^{n_{i+1}} \ell_{i+1}(k) = R_A. \tag{8}$$

**Lemma 11.** Matrix $G_A$ supports flow $d_A$.

Let $G_{d_A} = \begin{bmatrix} P_t & A_t \end{bmatrix}$ be the submatrix of $G_A$ corresponding to a solution for flow $d_A$; here $P_t$ is a submatrix of $P$ and $A_t$ is a submatrix of $A$. We let $G'_{d_A} = \begin{bmatrix} P_c \ A_t \end{bmatrix}$, where $P_c$ is the submatrix of $P$ consisting of the rows of $P$ that are used for the construction of $P_t$; $P_t$ is then a submatrix of $P_c$ consisting of a subset of its columns. Notice that matrices $P_t$ and $P_c$ have a natural partition into $n_{i+1} \times n_i$ block matrices which is induced by the block structure of matrix $P$. Next let $G_B = \begin{bmatrix} P_c \\ B \end{bmatrix}$ and partition $G_B$ into blocks as follows:

$$G_B = \begin{bmatrix} P_c([m_{i+1}],1) & \cdots & P_c([m_{i+1}],n_i) \\ G_i(n_{i+1}+1,1) & \cdots & G_i(n_{i+1}+1,n_i) \\ \vdots & \ddots & \vdots \\ G_i(m_{i+1},1) & \cdots & G_i(m_{i+1},n_i) \end{bmatrix}.$$
Next we define the vector
\[ \mathbf{d}_B = (\ell_i(1), \ldots, \ell_i(n_i); \text{rank}(P), \ell_{i+1}(n_{i+1} + 1), \ldots, \ell_{i+1}(m_{i+1})) \].

Since
\[ \text{rank}(P) = \sum_{j=1}^{n_i+1} \ell_{i+1}(j) + \sum_{k=1}^{n_i} \ell_i(k) - R_{d_i} = \sum_{k=1}^{n_i} \ell_i(k) - \sum_{j=n_{i+1}+1}^{m_{i+1}} \ell_{i+1}(j), \]

it follows that
\[ \sum_{k=1}^{n_i} \ell_i(k) = \text{rank}(P) + \sum_{j=n_{i+1}+1}^{m_{i+1}} \ell_{i+1}(j) = R_B \]
and \( \mathbf{d}_B \) is a valid flow vector. Furthermore we have:

**Lemma 12.** Matrix \( G_B \) supports flow \( \mathbf{d}_B \).

Let \( G_{d_B} = \begin{bmatrix} P_{ct} & A_t \\ B_t & L_t \end{bmatrix} \) be the submatrix of \( G_B \) corresponding to a solution for flow \( \mathbf{d}_B \); here \( P_{ct} \) is a submatrix of \( P_c \) and \( B_t \) is a submatrix of \( B \). We let \( G'_{d_B} = \begin{bmatrix} P_{cr} & A_t \\ B_t & L_t \end{bmatrix} \), where \( P_{cr} \) is the submatrix of \( P_c \) consisting of the columns of \( P_c \) that are used for the construction of \( P_{ct} \); \( P_{ct} \) is then a submatrix of \( P_{cr} \) consisting of a subset of its rows. Finally, let \( L_t \) be the submatrix of \( L \) obtained by intersecting the set of columns with indices matching those of the columns of \( A_t \) with the set of rows with indices matching those of \( B_t \). Observe that
\[ G_{d_i} = \begin{bmatrix} P_{cr} & A_t \\ B_t & L_t \end{bmatrix} \]
is a submatrix of \( G_i \). Our final step is the following lemma:

**Lemma 13.** Matrix \( G_{d_i} \) as defined above is a solution for flow \( \mathbf{d}_i \) for matrix \( G_i \).

To summarize the preceding argument, we have established the existence of a solution \( G_{d_i} \) when \( m_i = 1 \), when \( m_{i+1} = 1 \), and when \( m_i \geq 2 \) and \( m_{i+1} \geq 2 \). Our proof of Theorem 4 is complete.

### 3 Proof of Theorem 6

We prove a more general statement. Consider a network \( \mathcal{N} \) with an arbitrary number of nodes in the first and last layers. For a rate-\( R \) flow vector \( \mathbf{d} = (\ell_1(1), \ldots, \ell_1(m_1); \ell_M(1), \ldots, \ell_M(m_M)) \) we show that:

**Theorem 14.** Network \( \mathcal{N} \) supports rate-\( R \) flow \( \mathbf{d} \) if and only if for every cut \( \Omega \),
\[ \mathcal{C}(\Omega) \geq \sum_{\mathcal{O}_1(j) \in \Omega} \ell_1(j) + \sum_{\mathcal{O}_M(k) \in \Omega} \ell_M(k) - R. \tag{9} \]

Notice that for \( M = 2 \), Theorem 14 reduces to Theorem 4. Also if \( \mathcal{N} \) has a single node each in the first and last layers, then \( \ell_1(1) = \ell_M(1) = R \), and hence (9) and Theorem 1 imply that for every cut \( \Omega \) with \( \mathcal{S} \in \Omega \) and \( \mathcal{D} \in \bar{\Omega} \), \( \mathcal{C}(\Omega) \geq \mathcal{C} \geq R \). Thus, Theorem 14 implies Theorem 6.
3.1 Proof of Theorem 14

We use induction on $M$. For $M = 2$, Theorem 4 gives the result. For $M > 2$, choose $K \in \{2, \cdots, M - 1\}$. Define networks $\mathcal{N}_A$ and $\mathcal{N}_B$ to respectively be the subnetworks of $\mathcal{N}$ with node set $\mathcal{O}_1 \cup \cdots \cup \mathcal{O}_K$ and $\mathcal{O}_K \cup \cdots \cup \mathcal{O}_M$. The next step of our argument is to show that the inductive hypothesis and (9) imply the existence of non-negative integers $\ell_K(1), \cdots, \ell_K(m_K)$ such that $\sum_{i=1}^{m_K} \ell_K(i) = R$ and networks $\mathcal{N}_A$ and $\mathcal{N}_B$ support the rate-$R$ flows

\[ d_A = (\ell(1), \cdots, \ell_1(m_1); \ell_K(1), \cdots, \ell_K(m_K)) \]
\[ d_B = (\ell_K(1), \cdots, \ell_K(m_K); \ell_M(1), \cdots, \ell_M(m_M)). \]

This step would establish that $\mathcal{N}$ supports flow $d$ since submatrix $R_{d_1}$ can be obtained from the solution to $\mathcal{N}_A$ for $i \in \{1, \cdots, K - 1\}$ and from the solution to $\mathcal{N}_B$ for $i \in \{K, \cdots, M - 1\}$.

By the inductive hypothesis if the desired $\ell_K(1), \cdots, \ell_K(m_K)$ exist then they are non-negative integers which form a feasible solution to the following system of linear constraints:

\[
\begin{align*}
C(\Omega_A) & \geq \sum_{i \in \Omega_A} \ell_1(i) + \sum_{i \in \Omega_K} \ell_K(i) - R, & \text{for every } \Omega_A \text{ in } \mathcal{N}_A, \\
C(\Omega_B) & \geq \sum_{i \in \Omega_B} \ell_K(i) + \sum_{i \in \Omega_M} \ell_M(i) - R, & \text{for every } \Omega_B \text{ in } \mathcal{N}_B, \\
\sum_{i=1}^{m_K} \ell_K(i) & = R.
\end{align*}
\]

For any set $T \subseteq \mathcal{O}_K$ define:

\[
\begin{align*}
f_A(T) & = \min \left\{ C(\Omega_A) - \sum_{i \in \Omega_A} \ell_1(i) + R : \Omega_A \cap \Omega_K = T \right\} \\
f_B(T) & = \min \left\{ C(\Omega_B) - \sum_{i \in \Omega_B} \ell_M(i) + R : \Omega_K \cap \Omega_B = T \right\}.
\end{align*}
\]

Then the system (10) of linear constraints is equivalent to

\[
\begin{align*}
\sum_{j \in T} \ell_K(j) & \leq f_A(T), & \text{for every } T \subseteq \mathcal{O}_K \\
\sum_{j \in T} \ell_K(j) & \leq f_B(T), & \text{for every } T \subseteq \mathcal{O}_K \\
\sum_{i=1}^{m_K} \ell_K(i) & = R.
\end{align*}
\]

**Lemma 15.** The functions $f_A(T)$ and $f_B(T)$ are I) submodular, II) nondecreasing, and satisfy III) $f_A(\emptyset) = 0$ and $f_B(\emptyset) = 0$. Notice that function $f$ is submodular if for every $T_1$ and $T_2$:

\[ f(T_1) + f(T_2) \geq f(T_1 \cap T_2) + f(T_1 \cup T_2) \]

and is nondecreasing if for every $T_1 \subseteq T_2$, $f(T_1) \leq f(T_2)$.

Referring to terminology in polyhedral optimizations (see [24, §5.15]) a polytope $P$ is integer if and only if each vertex of $P$ has integral coordinates. If a polyhedron $P = \{ x : Ax \leq b \}$ in $n$ dimensions is integer, then any linear programming problem $\max \{ c^T x : Ax \leq b \}$ with a finite solution must have a solution with integral coordinates.

Let $f$ be a submodular function on some set $V$ with $v$ elements. The *polymatroid* associated with $f$ is:

\[ P_f = \{ x \in \mathbb{R}^v : x \geq 0, \ x(U) \leq f(U) \text{ for every } U \subseteq V \}, \]

where we define $x = [x(1) \cdots x(v)]^T$ and $x(U) = \sum_{u \in U} x(u)$. 

10
Theorem 16. ([9]) Let $f_1$ and $f_2$ be nondecreasing submodular set functions with integer values on $V$ with $f_1(\emptyset) = f_2(\emptyset) = 0$. Then $P_{f_1} \cap P_{f_2}$ is integer and for each $U \subseteq V$,

$$
\max \{x(U) : x \in P_{f_1} \cap P_{f_2}\} = \min_{T \subseteq U} (f_1(T) + f_2(U \setminus T)).
$$

(14)

For the submodular set functions $f_A$ and $f_B$, define the polymatroids:

$$
P_{f_A} = \{x \in \mathbb{R}^{m_K} : x \geq 0, \quad x(T) \leq f_A(T) \quad \text{for every } T \subseteq O_K\},
$$

$$
P_{f_B} = \{x \in \mathbb{R}^{m_K} : x \geq 0, \quad x(T) \leq f_B(T) \quad \text{for every } T \subseteq O_K\}.
$$

For (13) to have a non-negative and integral solution, $\max \{x(O_K) : x \in P_{f_A} \cap P_{f_B}\} \geq R$ is clearly necessary. To show sufficiency suppose $y \in P_{f_A} \cap P_{f_B}$ achieves $\max \{x(O_K) : x \in P_{f_A} \cap P_{f_B}\} \geq R$. Then for every choice of $0 \leq \ell_K(j) \leq y(j)$, $[\ell_K(1) \quad \cdots \quad \ell_K(m_K)]^T \in P_{f_A} \cap P_{f_B}$ and so we choose $\ell_K(j)$ such that $\sum_{i=1}^{m_K} \ell_K(i) = R$. Lemma 15 and Theorem 16 imply that:

$$
\max \{x(O_K) : x \in P_{f_A} \cap P_{f_B}\} = \min_{T \subseteq O_K} (f_A(T) + f_B(O_K \setminus T))
$$

(15)

and the optimum can be achieved by a non-negative integer solution. Theorem 14 follows from (15) and the following lemma:

Lemma 17. $\min_{T \subseteq O_K} (f_A(T) + f_B(O_K \setminus T)) \geq R$ if and only if for every cut $\Omega$ in $\mathcal{N}$,

$$
C(\Omega) \geq \sum_{O(\ell) \in \Omega} \ell_1(\ell) + \sum_{O(\ell) \in \Omega} \ell_M(\ell) - R.
$$

(16)

References

[1] R. Ahlswede, N. Cai, S-Y. R. Li, and R. W. Yeung, “Network information flow,” *IEEE Trans. Inform. Theory*, Vol. 46, pp. 1204-1216, July 2000.

[2] A. Anmaudruz, C. Fраголи, “Combinatorial algorithms for wireless information flow”, *Proceedings of ACM-SIAM Symposium on Discrete Algorithms (SODA09)*, January 2009.

[3] M. R. Aref, *Information Flow in Relay Networks*, Ph.D. dissertation, Stanford Univ., Stanford, CA, 1980.

[4] K. A. Atkinson, *An Introduction to Numerical Analysis*, 2nd edition, John Wiley & Sons, New York, 1989.

[5] S. Avestimehr, S. Diggavi and D. Tse, “A deterministic approach to wireless relay networks”, *Proceedings of Allerton Conference on Communication, Control, and Computing*, Illinois, September 2007.

[6] S. Avestimehr, S. Diggavi and D. Tse, “Wireless network information flow”, *Proceedings of Allerton Conference on Communication, Control, and Computing*, Illinois, September 2007.

[7] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Second Edition, John Wiley & Sons, Inc., Hoboken, New Jersey, 2006.

[8] R. Diestel, *Graph Theory*, Third Edition, Springer, Berlin, 2005.
[9] J. Edmonds, “Submodular functions, matroids and certain polyhedra” In *Combinatorial Structures and Their Applications; Proceedings of the Calgary International Conference on Combinatorial Structures and Their Applications* 1969 (R. Guy, H. Hanani, N. Sauer, J. Schonheim, eds.), Gordon and Breach, New York, pp. 69–87, 1970.

[10] J. Edmonds, E. Giles, “A min-max relation for submodular functions on graphs”, *Annals of Discrete Mathematics*, 1, pp. 185–204, 1977.

[11] R. Etkin, D. Tse, and H. Wang, “Gaussian interference channel capacity to within one bit”, Submitted.

[12] L. R. Ford, and D. R. Fulkerson, “Maximal flow through a network,” *Canadian Journal of Mathematics*, vol. 8, pp. 399–404, 1956.

[13] M. Grötschel, L. Lovász, A. Schrijver, “The ellipsoid method and its consequences in combinatorial optimization”, *Combinatorica* 1, pp. 169-197, 1981.

[14] S. Iwata, L. Fleischer, and S. Fujishige, “A combinatorial, strongly polynomial-time algorithm for minimizing submodular functions”, In *Proceedings of the 32th Annual ACM Symposium on Theory of Computing (STOC)*, 2000.

[15] B. Korte and L. Lovász, “Greedoids—a structural framework for the greedy algorithm,” in: W. Pulleybank (Ed.), *Progress in Combinatorial Optimization*, Academic Press, New York, pp. 221-243, 1984.

[16] B. Korte and L. Lovász, “Structural properties of greedoids,” *Combinatorica*, Volume 3, Numbers 3-4, pp. 359-374, September, 1983.

[17] E. L. Lawler, C. U. Martel, “Computing maximal “polymatroidal” network flows,” *Mathematics of Operations Research* 7, pp. 334-347, 1982.

[18] L. Minsky, *Transversal Theory*, Mathematics in Science and Engineering, Vol. 75, Academic Press, London, 1971.

[19] S. Mohajer, S. Diggavi, C. Fragouli, and D. Tse, “Transmission techniques for relay–interference networks”, *Proceedings of Allerton Conference on Communication, Control, and Computing*, Illinois, September 2008.

[20] H. Perfect, “A generalization of Rado’s theorem on independent transversals,” *Proc. Cambridge Phil. Soc.* 66, 513–515, 1969.

[21] L. Qi, “Odd submodular functions, Dilworth functions and discrete convex functions”, *Mathematics of Operations Research*, Vol. 13, No. 3, August 1988.

[22] N. Ratnakar, G. Kramer, “The multicast capacity of deterministic relay networks with no interference”, *IEEE Transactions on Information Theory* 52(6), pp. 2425-2432, 2006.

[23] A. Schrijver, “A combinatorial algorithm minimizing submodular functions in strongly polynomial time”, *Journal of Combinatorial Theory, Series B* 80, pp. 346-355, 2000.

[24] A. Schrijver, *Combinatorial Optimization*, Springer, Berlin, 2003.

[25] D. J. A. Welsh, *Matroid Theory*, Academic Press, London, 1976.
Algorithm 1 Submatrix $G_{d_i}$

Initialize: $T_i^{(0)} = T_c^{(0)} = G_i$, $t = 0$.

1. Look for a submatrix $T_i^{(t+1)}$ of $G_i$ that can be obtained by removing a row from $T_i^{(t)}$ and has the property that
\[
\text{rank}(T_i^{(t+1)}(W, U)) \geq \sum_{j \in U} \ell_i(j) + \sum_{k \in W} \ell_{i+1}(k) - R
\]
for all $U \subseteq [m_i]$ and $W \subseteq [m_{i+1}]$. If such a $T_i^{(t+1)}$ exists, then $t \leftarrow t + 1$ and repeat 1. Otherwise $T_c^{(0)} \leftarrow T_i^{(t)}$, $t \leftarrow 0$, and goto 2.

2. Look for a submatrix $T_c^{(t+1)}$ of $G_i$ that can be obtained by removing a column from $T_c^{(t)}$ and has the property that
\[
\text{rank}(T_c^{(t+1)}(W, U)) \geq \sum_{j \in U} \ell_i(j) + \sum_{k \in W} \ell_{i+1}(k) - R
\]
for all $U \subseteq [m_i]$ and $W \subseteq [m_{i+1}]$. If such a $T_c^{(t+1)}$ exists, then $t \leftarrow t + 1$ and repeat 2. Otherwise output $G_{d_i} = T_c^{(t)}$.

Appendix

Proof of Theorem 7:

Suppose that matrix $G_i$ and flow $d_i$ satisfy the conditions of Theorem 4. We first prove that Algorithm 1 will find a solution $G_{d_i}$ for flow $d_i$, and we later discuss the complexity of the algorithm.

We begin by showing that after exiting Step 1, matrix $T_c^{(0)}$ has exactly $\sum_{j=1}^{m_{i+1}} \ell_{i+1}(j) = R$ rows. By definition, $\text{rank}(T_c^{(0)}([m_{i+1}], [m_i])) \geq \sum_j \ell_i(j) + \sum_k \ell_{i+1}(k) - R = R$. Suppose that $T_c^{(0)}$ has more than $R$ rows. By Theorem 4, $T_c^{(0)}$ has a submatrix $T_d$, which is a solution for flow $d_i$, and hence $T_d$ has $R$ rows. Therefore all rows that belong to $T_c^{(0)}$ but not to $T_d$ can be removed and the resulting matrix $T$ satisfies $\text{rank}(T(W, U)) \geq \sum_{j \in U} \ell_i(j) + \sum_{k \in W} \ell_{i+1}(k) - R$ for all $U \subseteq [m_i]$ and $W \subseteq [m_{i+1}]$. This contradicts the assumption that $T_c^{(0)}$ has no more rows that can be removed.

We can similarly argue that the matrix $G_{d_i}$ output at the end of Step 2 has $R$ columns. Since $G_{d_i}$ is an $R \times R$ matrix for which $\text{rank}(G_{d_i}(W, U)) \geq \sum_{j \in U} \ell_i(j) + \sum_{k \in W} \ell_{i+1}(k) - R$ for all $U \subseteq [m_i]$ and $W \subseteq [m_{i+1}]$, $G_{d_i}$ is a solution for flow $d_i$.

To find the complexity of Algorithm 1, we argue that at Step 1 or Step 2 a removable row or column can be found respectively by testing at most $R + 1$ rows or columns. For example, in Step 1, any row which is not part of $T_c^{(0)}$ can be removed without violating any of the stated rank conditions. The pigeonhole principle implies that at most $R + 1$ rows need to be checked to find a row which is not part of $T_c^{(0)}$. Define $\alpha(\Omega) = \text{rank}(G_i(\Omega)) - \sum_{j \in \Omega \cap O_i} \ell_i(j) - \sum_{k \in \Omega \cap O_{i+1}} \ell_{i+1}(k)$ for every cut $\Omega \subseteq O_i \cup O_{i+1}$. Then the conditions of Theorem 4 are equivalent to the requirement that $\alpha(\Omega) \geq 0$ for every cut $\Omega \subseteq O_i \cup O_{i+1}$. Lemma 18 in the appendix proves the submodularity of $\alpha(\Omega)$. By Lemma 20, which appears later in the appendix, the function $- \sum_{j \in \Omega \cap O_i} \ell_i(j) + \sum_{k \in \Omega \cap O_{i+1}} \ell_{i+1}(k)$ is also a submodular function of $\Omega$. Thus $\alpha(\Omega)$, which is the sum of two submodular functions, is also submodular. We next verify whether or not the minimum of $\alpha(\Omega)$
is non-negative. It is known (see, e.g., [23] and [14]) that the minimum value of a submodular function can be found in strongly polynomial time. Here we use the algorithm by Schrijver [23] that finds the minimum of a submodular function \( f \) on the power set of set \( E_f \), in time \( O(|E_f|^6 t_f) \), where \( t_f \) is the time for evaluating function \( f \) for some subset of \( E_f \). In our problem \( f = \alpha \) which is defined on \( E = \bigcup_{i=1}^{t} O_i \). If we suppose that \( |O_i| \leq m \), then \( |E| \leq 2m \). Every evaluation of \( \alpha \) requires calculating the rank of submatrix \( G_i(\Omega) \) with size at most the size of \( G_i \). Suppose that \( G_i \) has dimension at most \( h_0 \times h_0 \). The rank of \( G_i(\Omega) \) can then be evaluated, for instance, by Gaussian elimination in \( O(h_0^3) \) time [4]. The complexity of minimizing \( \alpha(\Omega) \) is therefore

\[
O(|E|^{6} t_{\alpha}) = O\left(m^6 h_0^3 \right).
\]

The number of iterations of Step 1 of Algorithm 1 is at most the number of rows, \( h_0 \). Likewise there are at most \( h_0 \) iterations of Step 2. Since each iteration requires at most \( R + 1 \) minimization of the submodular function \( \alpha \), the total complexity of the algorithm is at most

\[
(2h_0) \cdot (R + 1) \cdot O\left(m^6 h_0^3 \right) = O\left(Rm^6 h_0^4 \right).
\]

As we run Algorithm 1 for \( G_i, i \in [M - 1] \), the complexity of this part will be \( O\left(MRm^6 h_0^4 \right) \).

**Proof of Theorem 8:**

First we prove that the capacity \( C \) of the network \( \mathcal{N} \) can be computed in polynomial time. For any cut \( \Omega \subseteq \mathcal{V}, C(\Omega) = \sum_{i=1}^{M-1} \text{rank}(G_i(\Omega)) \). By Lemma 18, rank \( (G_i(\Omega)) \) is a submodular function of \( \Omega \). Thus \( C(\Omega) \) is a submodular function of \( \Omega \). Next suppose that \( \Omega_1 \) and \( \Omega_2 \) are two cuts that separate \( S \) from \( D \). Then \( \Omega_1 \cap \Omega_2 \) and \( \Omega_1 \cup \Omega_2 \) separate \( S \) from \( D \). Therefore \( C(\Omega) \) is a submodular function over all cuts \( \Omega \subseteq \mathcal{V} \) that separate \( S \) from \( D \). To evaluate \( C = \min C(\Omega) \) over all cuts that separate \( S \) from \( D \) we use a submodular minimization algorithm [23] with running time of \( O(|E|^6 t_C) \). Here \( E_C = \mathcal{V} \setminus \{S, D\} \) is the ground set of \( C(\Omega) \), and \( |E_C| \leq m(M - 2) \), where \( m \) is the maximum number of nodes in each layer. \( t_C \) denotes the time for the evaluation of \( C(\Omega) = \sum_{i=1}^{M-1} \text{rank}(G_i(\Omega)) \) for a given \( \Omega \), and it involves \( M - 1 \) rank evaluations. Recall that each matrix \( G_i \) has size at most \( h_0 \times h_0 \). Then \( G_i(\Omega) \) has size at most \( h_0 \times h_0 \). Therefore using Gaussian elimination for rank evaluation, we have \( t_C = O(Mh_0^3) \). Therefore, the total complexity of computing the capacity \( C \) is \( O(m^6 M^7 h_0^3) \).

Next we discuss the complexity of the evaluation of the vectors \( d_i \). As we discussed in the proof of Theorem 14, in order to determine \( \ell_K(j) \) for a fixed \( K \), we need to solve the optimization problem (15) for the vector \( \mathbf{x} \). This is an integer programming problem over the intersection of two polymatroids. Let \( E_x = \{x(1), \cdots, x(m_K)\} \). Then \( |E_x| \leq m \). Let \( t_x \) be the time needed for one evaluation of functions \( f_A(T) \) and \( f_B(T) \) as defined in Section 3. By applying the result of [24, Theorem 47.1], it follows that \( \mathbf{x} \) can be found in time \( O(|E_x|^{6} t_x) \) if we use the algorithm of [23] for minimizing an integer programming problem over a polymatroid. Recall that

\[
f_A(T) = \min \left\{ C(\Omega_A) - \sum_{O_i(j) \in \Omega_A} \ell_1(j) + R : O_K \cap \Omega_A = T \right\}.
\]

Define for cut \( \Omega_A \) in \( \mathcal{N}_A \) with \( O_K \cap \Omega_A = T \), \( \phi_T(\Omega_A) = C(\Omega_A) - \sum_{O_i(j) \in \Omega_A} \ell_1(j) + R \). By Lemmas 19 and 20 the functions \( C(\Omega) \) and \( -\sum_{O_i(j) \in \Omega} \ell_1(j) + R \) are submodular over the set of all cuts in \( \mathcal{N}_A \). Next, if for two cuts \( \Omega_1 \) and \( \Omega_2 \) we have \( O_K \cap \Omega_1 = T \) and \( O_K \cap \Omega_2 = T \) then De Morgan’s laws imply that \( O_K \cap (\Omega_1 \cap \Omega_2) = T \) and \( O_K \cap (\Omega_1 \cup \Omega_2) = T \). Thus \( \phi_T(\Omega_A) \) is a submodular function over all cuts \( \Omega_A \) with \( O_K \cap \Omega_A = T \). Hence the evaluation of \( f_A(T) \) involves the minimization of a submodular function. The complexity of such a minimization over a set \( E_\phi \) is \( O(|E_\phi|^{6} t_\phi) \). Since
the nodes in \( O_K \cap \Omega_A = T \) are already fixed, the set \( E_{\phi_T} \) is the set of all nodes in \( O_1 \cup \cdots \cup O_{K-1} \). Therefore \(|E_{\phi_T}| \leq m(K-1)\). \( t_{\phi_T} \) is the time for an evaluation of \( C(\Omega_A) - \sum_{C_1(j) \in \Omega_A} \ell_1(j) + R \) for a given cut \( \Omega_A \). Since \( C(\Omega_A) = \sum_{i=1}^{K-1} \text{rank}(G_i(\Omega_A)) \), there are \( K-1 \) rank evaluations. Each rank function can be evaluated in time at most \( O(h_0^3) \) using Gaussian elimination. Thus \( t_{\phi_T} = O(Kh_0^3) \) and \( f_A(T) \) can be evaluated in time at most

\[
O(|E_{\phi_T}|^6t_{\phi_T}) = O(Kh_0^3 \cdot (m(K-1))^6) = O(K^7 m^6 h_0^3).
\]

Similarly, \( f_B(T) \) can be evaluated in \( O(((M-K)^7 + K^7)m^6 h_0^3) \). The total time for evaluating \( f_A \) and \( f_B \) is therefore \( O(((M-K)^7 + K^7)m^{12} h_0^3) \). Therefore evaluating vector \( x \) needs time

\[
O(m^6 \cdot ((M-K)^7 + K^7)m^6 h_0^3) = O(((M-K)^7 + K^7)m^{12} h_0^3)
\]

The function above is maximized when \( K = 0 \) and the time complexity is \( O(M^7 m^{12} h_0^3) \). Since the vector \( x \) needs to be computed for every layer \( i \in \{2, \cdots, M-1\} \), we find that the total complexity of the second stage is \( O(M^8 m^{12} h_0^3) \). Thus the total complexity of constructing the transmission scheme is

\[
O(M^8 m^{12} h_0^3) + O(MRm^6 h_0^4).
\]

Proof of Lemma 9:

Let \( T(0) = G_i \) and for \( t \geq 1 \) define \( T(t) \) to be a submatrix of \( G_i \) obtained by removing an arbitrary row from \( T(t-1) \). Observe that for every \( t \geq 1 \), \( \hat{U} \subseteq [m_i] \), and \( \hat{W} \subseteq [m_{i+1}] \),

\[
\text{rank}(T(t)(\hat{W}, \hat{U})) \geq \text{rank}(T(t-1)(\hat{W}, \hat{U})) - 1.
\]

Suppose that (2) is satisfied with strict inequality for all \( U \subseteq [m_i] \) and \( W \subseteq [m_{i+1}] \). Then by (17), for all \( \hat{U} \subseteq [m_i] \) and \( \hat{W} \subseteq [m_{i+1}] \),

\[
\text{rank}(T(t)(\hat{W}, \hat{U})) \geq \sum_{j \in \hat{U}} \ell_i(j) + \sum_{k \in \hat{W}} \ell_{i+1}(k) - R_{d_i}.
\]

If \( T^{(1)} \) has a tight submatrix we are done. Otherwise, let \( T^{(\lambda)} \) denote the empty submatrix of \( G_i \). Observe that

\[
\text{rank}(T^{(\lambda)}([m_{i+1}], [m_i])) = 0 < \sum_{j=1}^{m_i} \ell_i(j) + \sum_{k=1}^{m_{i+1}} \ell_{i+1}(k) - R_{d_i} = R_{d_i},
\]

and so the rows of \( G_i \) cannot be removed indefinitely without violating at least one rank condition. Therefore by (17) there must be some \( t < \lambda \) such that \( T = T^{(t)} \) satisfies (7) for all \( \hat{U} \) and \( \hat{W} \) and \( T \) has a tight submatrix.

Proof of Lemma 10:

Consider a tight submatrix \( G_i(W, U) \) of \( G_i \) that is not proper. Suppose, without loss of generality, that \( G_i(W, U) \) is of the form \( P = G_i(\{1\}, [m_i]) \). Remove rows from \( G_i \) arbitrarily among the row blocks with indices in \( \{2, 3, \cdots, m_{i+1}\} \), until no further rows can be removed from the resulting submatrix \( T \) of \( G_i \) without violating \( \text{rank}(T(\hat{W}, \hat{U})) \geq \sum_{j \in \hat{U}} \ell_i(j) + \sum_{k \in \hat{W}} \ell_{i+1}(k) - R_{d_i} \) for some \( \hat{U} \subseteq [m_i] \) and \( \hat{W} \subseteq [m_{i+1}] \). Notice that this process terminates before we remove all rows from blocks with indices in \( \{2, 3, \cdots, m_{i+1}\} \) for if \( T \) is a matrix with rows only from row block 1, then
\[
\text{rank}(T(\{2, \cdots, m_{i+1}\}, [m_i])) = 0 < \sum_{j=1}^{m_i} \ell_i(j) + \sum_{k=2}^{m_{i+1}} \ell_{i+1}(k) - R_{d_i} = \sum_{k=2}^{m_{i+1}} \ell_{i+1}(k). \]
Next replace \( G_i \) with \( T \) for the rest of our argument. Since no row can be removed from \( G_i \) among the row blocks with indices in \( \{2, 3, \cdots, m_{i+1}\} \) without violating a rank condition, there exists a tight submatrix \( Q = G_i(W', U') \) of \( G_i \) which has a non-empty intersection with the row blocks of \( G_i \) with indices in \( \{2, 3, \cdots, m_{i+1}\} \). Now consider some possibilities for \( Q \). If \( Q \) is a proper submatrix, then we are done. If \( Q \) is not a proper submatrix consider two cases:

- \( |W'| = 1 \) and \( |U'| = m_i \). Without loss of generality suppose that \( Q = G_i(\{2\}, [m_i]) \). Since \( P \) and \( Q \) are tight, (2) implies \( \text{rank}(P) = \ell_{i+1}(1) \) and \( \text{rank}(Q) = \ell_{i+1}(2) \). Next consider the submatrix of \( G_i \) given by

\[
E = \begin{bmatrix} P \\ Q \end{bmatrix} = G_i(\{1, 2\}, [m_i]).
\]

By (2), we have \( \text{rank}(E) \geq \ell_{i+1}(1) + \ell_{i+1}(2) \). However, by considering the rows of \( E, P, \) and \( Q \), we see that, \( \text{rank}(E) \leq \text{rank}(P) + \text{rank}(Q) \), and therefore \( \text{rank}(E) \leq \ell_{i+1}(1) + \ell_{i+1}(2) \).

Thus, \( \text{rank}(E) = \ell_{i+1}(1) + \ell_{i+1}(2) \). \( E \) is therefore both a tight and a proper submatrix of \( G_i \).

- \( |W'| = m_{i+1} \) and \( |U'| = 1 \). Without loss of generality suppose that \( Q = G_i([m_{i+1}], \{1\}) \). We will next consider a collection of subcases. Suppose first that other than \( P \) and \( Q \) there exists another tight submatrix \( K \). If \( K \) is proper then there is nothing further to prove. If \( K \) is not proper then without loss of generality we can assume that \( K \) is either of the form \( K = G_i(\{2\}, [m_i]) \) or \( K = G_i([m_{i+1}], \{2\}) \). If \( K = G_i(\{2\}, [m_i]) \), then the matrix

\[
\begin{bmatrix} P \\ K \end{bmatrix}
\]

is proper by our argument in the previous case. Likewise, if \( K = G_i([m_{i+1}], \{2\}) \) then the matrix \( [Q \ K] \) is proper. Suppose next that \( P \) and \( Q \) are the only tight submatrices of \( G_i \). We already have assumed that no other row from \( G_i(\{2, 3, \cdots, m_{i+1}\}, [m_i]) = T(\{2, 3, \cdots, m_{i+1}\}, [m_i]) \) can be removed without violating some rank condition (2). Since \( P \) and \( Q \) are the only tight submatrices of \( G_i \), removing one row from \( G_i(\{2, 3, \cdots, m_{i+1}\}, [m_i]) \) will cause the violation of a rank condition only for the tight submatrix \( Q = G_i([m_{i+1}], \{1\}) \). If a row in \( G_i(\{2, 3, \cdots, m_{i+1}\}, \{1\}) \) was a linear combination of some other rows in \( Q \) it could be removed without violating any rank condition for submatrix \( Q \). Since \( Q \) is a tight submatrix and we constructed \( T \) so that no further rows could be removed from it without violating a rank condition, this is impossible. Hence every row in \( G_i(\{2, 3, \cdots, m_{i+1}\}, \{1\}) \) is independent from all other rows in \( Q \). Thus

\[
\text{rank}(Q) = \text{rank}(G_i(1, 1)) + r_{i+1}(2) + \cdots + r_{i+1}(m_{i+1}), \tag{18}
\]

where \( r_{i+1}(j) \) is the number of rows of the \( j \)th row block of \( G_i \). By inequality (2) for \( U = [m_i] \) and \( W = \{j\} \), we have \( \text{rank}(G_i(\{j\}, [m_i])) \geq \ell_{i+1}(j) \) for every \( j \in \{2, \cdots, m_{i+1}\} \). Furthermore, the rank of a matrix is at most the number of rows of the matrix, and hence \( r_{i+1}(j) \geq \text{rank}(G_i(\{j\}, [m_i])) \). Consequently \( r_{i+1}(j) \geq \ell_{i+1}(j) \). This together with (18) and the fact that \( \text{rank}(Q) = \ell_i(1) \) imply that

\[
\text{rank}(G_i(1, 1)) \leq \ell_i(1) - \ell_{i+1}(2) - \cdots - \ell_{i+1}(m_{i+1}) = \ell_i(1) + \ell_{i+1}(1) - R_{d_i}. \tag{19}
\]

However, by evaluating inequality (2) for \( U = \{1\} \) and \( W = \{1\} \) we obtain

\[
\text{rank}(G_i(1, 1)) \geq \ell_i(1) + \ell_{i+1}(1) - R_{d_i}. \tag{20}
\]

(19) and (20) imply that

\[
\text{rank}(G_i(1, 1)) = \ell_i(1) + \ell_{i+1}(1) - R_{d_i}.
\]
Hence $G_i(1, 1)$ is tight, and this contradicts the assumption that $P$ and $Q$ are the only tight submatrices of $G_i$.

**Proof of Lemma 11:**

Let $p$ be the index of the first column block of $G_A$ and let $n_i + 1, \ldots, m_i$ respectively be the indices of the other column blocks. Furthermore let $1, \ldots, n_i + 1$ be the indices of the row blocks of $G_A$. Select any two subsets $W \subseteq [n_{i+1}]$ and $U \subseteq \{p, n_i + 1, \ldots, m_i\}$. We consider two cases:

1. If $p \in U$: consider the submatrix $G_A(W, U)$. This submatrix is the same as $G_i(W, U')$ where $U' = (U \setminus \{p\}) \cup [n_i]$. Since $G_i(W, U')$ satisfies the condition of inequality (2) for vector $d_i$, we have:

$$\text{rank}(G_A(W, U)) \geq \sum_{j \in W} \ell_{i+1}(j) + \sum_{k \in U'} \ell_i(k) - R_{d_i}.$$  

(21)

If we expand the right hand side of (21), we have:

$$\text{rank}(G_A(W, U)) \geq \sum_{j \in W} \ell_{i+1}(j) + \sum_{k \in U \setminus \{p\}} \ell_i(k) + \sum_{t=1}^{n_i} \ell_i(t) - R_{d_i}$$

$$= \sum_{j \in W} \ell_{i+1}(j) + \sum_{k \in U \setminus \{p\}} \ell_i(k) - \sum_{t=n_i+1}^{m_i} \ell_i(t)$$

$$= \sum_{j \in W} \ell_{i+1}(j) + \sum_{k \in U \setminus \{p\}} \ell_i(k) - (R_A - \text{rank}(P))$$  

(22)

where (22) follows by (8). The last expression is the condition of inequality (2) for $G_A(W, U)$ when $G_A$ supports flow $d_A$.

2. If $p \notin U$: define $V = U \cup \{p\}$ so that we can apply (22):

$$\text{rank}(G_A(W, V)) \geq \sum_{j \in W} \ell_{i+1}(j) + \sum_{k \in U} \ell_i(k) + \text{rank}(P) - R_A.$$  

(23)

Observe that

$$\text{rank}(G_A(W, U)) \geq \text{rank}(G_A(W, V)) - \text{rank}(G_A(W, \{p\}))$$

$$\geq \text{rank}(G_A(W, V)) - \text{rank}(P).$$  

(24)

By (23) and (24),

$$\text{rank}(G_A(W, U)) \geq \sum_{j \in W} \ell_{i+1}(j) + \sum_{k \in U} \ell_i(k) - R_A,$$

which is the rank condition (2) for $G_A(W, U)$ when $G_A$ supports flow $d_A$.

The block matrix $G_A$ has block dimension $n_{i+1} \times (m_i - n_i + 1)$. Therefore, $G_A$ has strictly fewer blocks than $G_i$ unless $n_{i+1} = m_{i+1}$ and $m_i - n_i + 1 = m_i$ or $n_i = 1$. This is impossible since $P$ is a proper submatrix of $G_i$. It follows from our induction hypothesis matrix $G_A$ supports flow $d_A$. 

17
Proof of Lemma 12:
Let the indices for the row and column blocks of \( G_B \) respectively be \( p, n_{i+1}, \ldots, m_i \) and \( 1, \ldots, n_i \). Define \( G'_B = \begin{bmatrix} P \\ B \end{bmatrix} \). Observe that matrix \( G'_B \) supports flow \( d_B \) since the only change needed to the proof of Lemma 11 is to take the transposition of all matrices. Next, since \( G_{d_A} \) is a solution the flow \( d_A \), then by definition of flow the matrix \( P_t \) has full column rank and has \( \text{rank}(P) \) columns. Therefore
\[
\text{rank}(P_t) = \text{rank}(P).
\]
Since \( P_c \) is a submatrix of \( P \) and \( P_t \) is a submatrix of \( P_c \), we have
\[
\text{rank}(P) \geq \text{rank}(P_c) \geq \text{rank}(P_t),
\]
and therefore
\[
\text{rank}(P_c) = \text{rank}(P). \tag{25}
\]
This implies that the rows that are in \( P \) but not in \( P_c \) are linear combination of the rows of \( P_c \). It follows that for every \( U \subseteq [n_i] \) the rows that are in \( P([n_{i+1}],U) \) but not in \( P_c([n_{i+1}],U) \) are linear combination of the rows of \( P_c([n_{i+1}],U) \). Hence
\[
\text{rank}(P_c([n_{i+1}],U)) = \text{rank}(P([n_{i+1}],U)).
\]
This implies that for every \( W \subseteq \{p, n_{i+1} + 1, \ldots, m_i + 1\} \), if \( p \in W \) then \( G_B(W,U) = \begin{bmatrix} P_c([n_{i+1}],U) \\ G_i(W \setminus p, U) \end{bmatrix} \) and
\[
\text{rank}(G_B(W,U)) = \text{rank} \left( \begin{bmatrix} P_c([n_{i+1}],U) \\ G_i(W \setminus p, U) \end{bmatrix} \right) = \text{rank} \left( \begin{bmatrix} P([n_{i+1}],U) \\ G_i(W \setminus p, U) \end{bmatrix} \right) = \text{rank}(G'_B(W,U)).
\]
If \( p \notin W \), \( G_B(W,U) \) does not depend on \( P_c \) and \( \text{rank}(G_B(W,U)) = \text{rank}(G'_B(W,U)) \). Therefore in general for every \( W \) and \( U \), replacing \( P_c \) with \( P \) in \( G_B \) will not change any rank function and we can still use the result of Lemma 11.

Proof of Lemma 13:
We have to verify two properties of \( G_{d_i} \). The first is that every row block \( j \in [m_{i+1}] \) of \( G_{d_i} \) has \( \ell_{i+1}(j) \) rows and every column block \( k \in [m_i] \) has \( \ell_i(k) \) columns. The second property is that \( \text{rank}(G_{d_i}) = R_{d_i} \). By our construction, the number of columns in column blocks \( j \in \{n_i + 1, \ldots, m_i\} \) and the number of rows in row blocks \( k \in [n_{i+1}] \) are respectively determined by \( G_{d_A} \). Since
\[
d_A = (\text{rank}(P), \ell_i(n_i + 1), \ell_i(n_i + 2), \ldots, \ell_i(m_i); \ell_i+1(1), \ldots, \ell_i+1(n_i+1)),
\]
there are the right number of rows and columns in these cases. Furthermore, the number of columns in column blocks \( j \in [n_i] \) and the number of rows in row blocks \( i \in \{n_{i+1} + 1, \ldots, m_i + 1\} \) are respectively determined by \( G_{d_B} \). Since
\[
d_B = (\ell_i(1), \ldots, \ell_i(n_i); \text{rank}(P), \ell_i+1(n_i + 1), \ldots, \ell_i+1(m_i+1))
\]
there are the right number of columns and rows in these cases as well. Hence every row block \( j \in [m_{i+1}] \) of \( G_{d_i} \) has \( \ell_{i+1}(j) \) rows and every column block \( k \in [m_i] \) has \( \ell_i(k) \) columns.
For the second property, observe that since \( G_{d_B} \) is a solution of flow \( d_B \) for \( G_B \), then by the definition of the solution of a flow the matrix \( P_{ct} \) has full row rank and has \( \text{rank}(P) \) rows. Therefore

\[
\text{rank}(P_{ct}) = \text{rank}(P).
\]

Since \( P_{ct} \) is a submatrix of \( P_{cr} \) consisting of a subset of its rows, \( \text{rank}(P_{cr}) \geq \text{rank}(P_{ct}) \). Because \( P_{cr} \) is a submatrix of \( P \), \( \text{rank}(P_{cr}) \leq \text{rank}(P) \). Thus

\[
\text{rank}(P_{cr}) = \text{rank}(P),
\]

and so the rows that are in \( P_{cr} \) are linear combinations of the rows in \( P_{ct} \). Next since the rows of \( B_t \) are also rows of the full-rank matrix \( G_{d_B} \), it follows that all rows of \( B_t \) are linearly independent and are independent from all other rows of \( G_{d_B} \) and consequently from all other rows in \( G'_{d_B} \). Therefore, all rows in \( [B_t \ L_t] \) are linearly independent and are independent from all other rows in \( G_{d_i} \). Therefore the two relationships follow:

\[
\text{rank}(G_{d_i}) = \text{rank}([B_t \ L_t]) + \text{rank}([P_{cr} \ A_t]),
\]

\[
\text{rank}([B_t \ L_t]) = \sum_{j=n_t+1+1}^{n_{t+1}} \ell_i(j).
\]

Notice that by (26) and (27) \( \text{rank}(P_{cr}) = \text{rank}(P_t) \). Recall that \( P_t \) is a submatrix of \( P_c \) consisting of a subset of its columns, the columns of \( P_c \) are linear combinations of the columns of \( P_{cr} \). Therefore

\[
\text{rank}([P_{cr} \ A_t]) = \text{rank}([P_c \ A_t]).
\]

By (25) and (26), \( \text{rank}(P_c) = \text{rank}(P_t) \). Recall that \( P_t \) is a submatrix of \( P_c \) consisting of a subset of its columns. Therefore the columns of \( P_c \) are linear combinations of the columns of \( P_t \). Hence,

\[
\text{rank}([P_c \ A_t]) = \text{rank}([P_t \ A_t]).
\]

Since \( G_{d_A} = [P_t \ A_t] \) is a solution for flow \( d_A \), it follows that \( \text{rank}(G_{d_A}) = \sum_{j=1}^{n_{t+1}} \ell_i(j) \). Thus, (30) and (31) imply

\[
\text{rank}([P_{cr} \ A_t]) = \sum_{j=1}^{n_{t+1}} \ell_i(j).
\]

From (28), (29), and (30) we conclude that

\[
\text{rank}(G_{d_i}) = \sum_{j=1}^{n_{t+1}} \ell_i(j) = R_{d_i}.
\]

**Proof of Lemma 15:**

**Part I:**

Here we only prove the submodularity of function \( f_A(T) \) as the proof for \( f_B(T) \) is similar. Consider layers \( O_i \) and \( O_{i+1} \) and transfer function \( G_i \). Recall that \( G_i(\Omega) \) is the transfer function from the nodes in \( \Omega \) to the nodes in \( \bar{\Omega} = (O_i \cup O_{i+1}) \setminus \Omega \). We first prove that:

**Lemma 18.** \( \text{rank}(G_i(\Omega)) \) is a submodular function over cuts \( \Omega \subseteq O_i \cup O_{i+1} \); i.e., for every two such cuts \( \Omega_1 \) and \( \Omega_2 \):

\[
\text{rank}(G_i(\Omega_1)) + \text{rank}(G_i(\Omega_2)) \geq \text{rank}(G_i(\Omega_1 \cap \Omega_2)) + \text{rank}(G_i(\Omega_1 \cup \Omega_2)).
\]
Figure 2: (a) The matrices $G_i(\Omega_1)$ and $G_i(\Omega_2)$, (b) the matrices $G_i(\Omega_1 \cap \Omega_2)$ and $G_i(\Omega_1 \cup \Omega_2)$.

We point out that the preceding result was first proved in [6] in order to study the time-expanded representation of a network which is not layered. [6] established Lemma 18 through an information theoretic argument involving the submodularity of the entropy function. We offer a new and combinatorial proof of Lemma 18.

Proof. Consider matrix $G_i$ in Figure 2-(a). Suppose that we have reordered the row blocks of $G_i$ and the column blocks of $G_i$ such that the blocks corresponding to the transfer matrices of $G_i(\Omega_1)$ and $G_i(\Omega_2)$ appear as in Figure 2-(a). We have depicted and labeled the different parts of the transfer matrices $G_i(\Omega_1 \cap \Omega_2)$ and $G_i(\Omega_1 \cup \Omega_2)$ in Figure 2-(b). Therefore we have:

$$G_i(\Omega_1 \cap \Omega_2) = \begin{bmatrix} A_2 \\ C \\ B_2 \end{bmatrix}, \quad G_i(\Omega_1 \cup \Omega_2) = \begin{bmatrix} A_1 & C & B_1 \end{bmatrix}.$$

We first prove that:

$$\text{rank} (G_i(\Omega_1)) + \text{rank} (C) \geq \text{rank} \left( \begin{bmatrix} A_1 \\ C \end{bmatrix} \right) + \text{rank} \left( \begin{bmatrix} A_2 \\ C \end{bmatrix} \right). \quad (33)$$

Let $p$ be the maximum number of rows $a_1, \cdots, a_p$ in $A_2$ which are independent in $\begin{bmatrix} a_1 \\ \vdots \\ a_p \\ C \end{bmatrix}$ and $q$ be the maximum number of rows $b_1, \cdots, b_q$ in $G_i(\Omega_1)$ but not in $\begin{bmatrix} A_1 & C \end{bmatrix}$ which are independent in $\begin{bmatrix} b_1 \\ \vdots \\ b_q \\ A_1 & C \end{bmatrix}$. We have:

$$\text{rank} \left( \begin{bmatrix} A_2 \\ C \end{bmatrix} \right) = \text{rank} (C) + p, \quad (34)$$

$$\text{rank} (G_i(\Omega_1)) = \text{rank} \left( \begin{bmatrix} A_1 & C \end{bmatrix} \right) + q. \quad (35)$$
Let \( a_1', \ldots, a_p' \) denote the rows in \( G_i(\Omega_1) \) which respectively end in the rows \( a_1, \ldots, a_p \) of \( A_2 \). Then these rows are clearly independent in 
\[
\begin{bmatrix}
a_1' \\
\vdots \\
a_p'
\end{bmatrix}
\begin{bmatrix}
A_1 \\
C
\end{bmatrix},
\]
and therefore \( q \geq p \). By (34) and (35),
\[
\text{rank} \left( G_i(\Omega_1) \right) - \text{rank} \left( \begin{bmatrix} \ A_1 \\ C \end{bmatrix} \right) \geq \text{rank} \left( \begin{bmatrix} \ A_2 \\ C \end{bmatrix} \right) - \text{rank} \left( \begin{bmatrix} \ C \\ B_1 \end{bmatrix} \right),
\]
which implies (33). A similar argument for \( G_i(\Omega_2) \) implies that:
\[
\text{rank} \left( G_i(\Omega_2) \right) + \text{rank}(C) \geq \text{rank} \left( \begin{bmatrix} \ C \\ B_1 \end{bmatrix} \right) + \text{rank} \left( \begin{bmatrix} \ C \\ B_2 \end{bmatrix} \right).
\]
(36)

By adding together inequalities (33) and (36) we find that
\[
\text{rank} \left( G_i(\Omega_1) \right) + \text{rank} \left( G_i(\Omega_2) \right) \geq \left( \text{rank} \left( \begin{bmatrix} \ A_1 \\ C \end{bmatrix} \right) + \text{rank} \left( \begin{bmatrix} \ C \\ B_1 \end{bmatrix} \right) - \text{rank}(C) \right)
+ \left( \text{rank} \left( \begin{bmatrix} \ A_2 \\ C \end{bmatrix} \right) + \text{rank} \left( \begin{bmatrix} \ C \\ B_2 \end{bmatrix} \right) - \text{rank}(C) \right).
\]
(37)

If we use the submodularity of the rank function of a matrix [25] we deduce that if \( W_1 \) and \( W_2 \) are the indices of rows (columns) of some matrix and \( \text{rank}(W) \) is the number of independent rows (columns) among those with indices in \( W \), then:
\[
\text{rank}(W_1) + \text{rank}(W_2) \geq \text{rank}(W_1 \cap W_2) + \text{rank}(W_1 \cup W_2).
\]
(38)

Applying (38) to the columns of matrix \( G_i(\Omega_1 \cup \Omega_2) \) and to the rows of matrix \( G_i(\Omega_1 \cap \Omega_2) \) we find that:
\[
\text{rank} \left( \begin{bmatrix} \ A_1 \\ C \end{bmatrix} \right) + \text{rank} \left( \begin{bmatrix} \ C \\ B_1 \end{bmatrix} \right) \geq \text{rank}(C) + \text{rank} \left( G_i(\Omega_1 \cup \Omega_2) \right)
\]
(39)

\[
\text{rank} \left( \begin{bmatrix} \ A_2 \\ C \end{bmatrix} \right) + \text{rank} \left( \begin{bmatrix} \ C \\ B_2 \end{bmatrix} \right) \geq \text{rank}(C) + \text{rank} \left( G_i(\Omega_1 \cap \Omega_2) \right).
\]
(40)

Lemma 18 follows from (37), (39), and (40).

Next we extend Lemma 18 to a multilayer network \( \mathcal{N}_A \):

**Lemma 19.** In the network \( \mathcal{N}_A \), \( \mathcal{C}(\Omega) \) is a submodular function over cuts \( \Omega \subseteq \mathcal{O}_1 \cup \cdots \cup \mathcal{O}_K \).

**Proof.** We decompose \( \Omega \) into the subsets \( \Omega = \Omega_1 \cup \Omega_2 \cup \cdots \cup \Omega_{K-1} \), where \( \Omega_i = \Omega \cap (\mathcal{O}_i \cup \mathcal{O}_{i+1}) \) defines a cut of the subnetwork of \( \mathcal{N}_A \) with set of nodes \( \mathcal{O}_i \cup \mathcal{O}_{i+1} \). We have
\[
\mathcal{C}(\Omega) = \sum_{j=1}^{K-1} \text{rank} \left( G_j(\Omega_j) \right) = \sum_{j=1}^{K-1} \text{rank} \left( G_j(\Omega_j) \right).
\]

If two cuts \( \Omega \) and \( \Omega' \) are respectively decomposed into \( [\Omega_i] \) and \( [\Omega'_i] \), then \( \Omega \cap \Omega' \) and \( \Omega \cup \Omega' \) will be respectively decomposed into \( [\Omega_i \cap \Omega'_i] \) and \( [\Omega_i \cup \Omega'_i] \). By Lemma 18, \( \text{rank} \left( G_j(\Omega_j) \right) \) is submodular. Therefore
\[
\mathcal{C}(\Omega) + \mathcal{C}(\Omega') = \sum_{j=1}^{K-1} \left( \text{rank} \left( G_j(\Omega_j) \right) + \text{rank} \left( G_j(\Omega'_j) \right) \right)
\geq \sum_{j=1}^{K-1} \left( \text{rank} \left( G_j(\Omega_j \cap \Omega'_j) \right) + \text{rank} \left( G_j(\Omega_j \cup \Omega'_j) \right) \right) = \mathcal{C}(\Omega \cap \Omega'_j) + \mathcal{C}(\Omega_j \cup \Omega'_j).
\]

Since the sum of submodular functions is submodular, the final result follows. \qed
We next prove the following useful lemma:

**Lemma 20.** For any function \( f \) defined on the set \( E \) and any given set \( V \subseteq E \), the function \( g \) on the power set of \( E \) defined as \( g(U) = \sum_{i \in U \cap V} f(k) \) is submodular.

**Proof.** By definition,

\[
g(U_1) + g(U_2) = \sum_{k \in U_1 \cap V} f(k) + \sum_{k \in U_2 \cap V} f(k)
\]

\[
= \sum_{k \in (U_1 \cap V) \cap (U_2 \cap V)} f(k) + \sum_{k \in (U_1 \cap V) \cup (U_2 \cap V)} f(k)
\]

\[
= \sum_{k \in (U_1 \cap U_2) \cap V} f(k) + \sum_{k \in (U_1 \cup U_2) \cap V} f(k)
\]

\[
= g(U_1 \cap U_2) + g(U_1 \cup U_2).
\]

\[\square\]

We next prove that \( f_A \) is submodular.

**Proof.** Suppose \( T_1, T_2 \subseteq \mathcal{O}_K \), and let \( A_1 \) and \( A_2 \) be the two cuts in \( \mathcal{N}_A \) such that

\[
f_A(T_1) = C(A_1) - \sum_{\mathcal{O}_1(j) \in A_1} \ell_1(j) + R, \quad \mathcal{O}_K \cap \overline{A}_1 = T_1
\]

\[
f_A(T_2) = C(A_2) - \sum_{\mathcal{O}_1(j) \in A_2} \ell_1(j) + R, \quad \mathcal{O}_K \cap \overline{A}_2 = T_2
\]

Consider \( A_1 \cap A_2 \) and \( A_1 \cup A_2 \). By De Morgan’s laws, \( \mathcal{O}_K \cap (\overline{A}_1 \cap \overline{A}_2) = \mathcal{O}_K \cap (\overline{A}_1 \cup \overline{A}_2) = T_1 \cap T_2 \) and \( \mathcal{O}_K \cap (\overline{A}_1 \cup \overline{A}_2) = \mathcal{O}_K \cap (\overline{A}_1 \cap \overline{A}_2) = T_1 \cap T_2 \). By (11) we have

\[
f_A(T_1 \cup T_2) \leq C(A_1 \cap A_2) - \sum_{\mathcal{O}_1(j) \in A_1 \cap A_2} \ell_1(j) + R, \quad (41)
\]

\[
f_A(T_1 \cap T_2) \leq C(A_1 \cup A_2) - \sum_{\mathcal{O}_1(j) \in A_1 \cup A_2} \ell_1(j) + R, \quad (42)
\]

By the submodularity of \( C(\Omega) \) we have

\[
C(A_1 \cap A_2) + C(A_1 \cup A_2) \leq C(A_1) + C(A_2).
\]

Furthermore

\[
\sum_{\mathcal{O}_1(j) \in A_1 \cap A_2} \ell_1(j) + \sum_{\mathcal{O}_1(j) \in A_1 \cup A_2} \ell_1(j) = \sum_{\mathcal{O}_1(j) \in A_1} \ell_1(j) + \sum_{\mathcal{O}_1(j) \in A_2} \ell_1(j).
\]

(44)

By (41)–(44), we have

\[
f_A(T_1 \cap T_2) + f_A(T_1 \cup T_2) \leq f_A(T_1) + f_A(T_2).
\]

\[\square\]
Part II:

We prove the result for $f_A$ and it is straightforward to modify it for $f_B$. It suffices to prove that for every $T \subseteq \mathcal{O}_K$ and every $\mathcal{O}_K(i) \notin T$, 

$$f_A(T) \leq f_A(T \cup \{\mathcal{O}_K(i)\}).$$

Suppose that for $T$ and $\mathcal{O}_K(i) \notin T$, cut $A$ in $\mathcal{N}_A$ achieves $f_A(T \cup \{\mathcal{O}_K(i)\})$. We have

$$f_A(T \cup \{\mathcal{O}_K(i)\}) = C(A) - \sum_{\mathcal{O}_1(j) \in A} \ell_1(j) + R, \quad \mathcal{O}_K \cap A = T \cup \{\mathcal{O}_K(i)\}. \quad (45)$$

Next consider the cut $A' = A \cup \{\mathcal{O}_K(i)\}$. Notice that $\bar{A}' \cap \mathcal{O}_K = T$. Therefore

$$C(A') - \sum_{\mathcal{O}_1(j) \in A'} \ell_1(j) + R \geq f_A(T). \quad (46)$$

Observe that for every $j \leq K - 2$, $G_j(A) = G_j(A')$. Furthermore $G_{K-1}(A)$ has the same row blocks as $G_{K-1}(A')$ and an additional row block corresponding to the transfer function from the nodes in $A \cap \mathcal{O}_{K-1}$ to $\{\mathcal{O}_K(i)\}$. Therefore $C(A) \geq C(A')$. Finally, $\sum_{\mathcal{O}_1(j) \in A} \ell_1(j) = \sum_{\mathcal{O}_1(j) \in A'} \ell_1(j)$, and hence

$$C(A) - \sum_{\mathcal{O}_1(j) \in A} \ell_1(j) + R = C(A') - \sum_{\mathcal{O}_1(j) \in A'} \ell_1(j) + R. \quad (47)$$

(45), (46), and (47) imply that $f_A(T) \leq f_A(T \cup \{\mathcal{O}_K(i)\})$.

Part III:

Since $\sum_{\mathcal{O}_1(j) \in \mathcal{O}_A} \ell_1(j) \leq R$ and $\sum_{\mathcal{O}_M(k) \in \bar{\mathcal{O}}_B} \ell_M(k) \leq R$, $f_A$ and $f_B$ are non-negative functions. Observe that by choosing cut $A = \mathcal{O}_1 \cup \cdots \cup \mathcal{O}_K$ for network $\mathcal{N}_A$ and cut $B = \emptyset$ for network $\mathcal{N}_B$, we find that $\bar{A} \cap \mathcal{O}_K = \emptyset$, $B \cap \mathcal{O}_K = \emptyset$, and

$$f_A(\emptyset) = C(A) - \sum_{\mathcal{O}_1(j) \in A} \ell_1(j) + R = f_B(\emptyset) = C(B) - \sum_{\mathcal{O}_M(j) \in \bar{B}} \ell_M(j) + R = 0.$$

Proof of Lemma 17:

Suppose that $T_0$ achieves $\min_{T \subseteq \mathcal{O}_K} (f_A(T) + f_B(\mathcal{O}_K \setminus T))$ and cuts $\Omega_A$ and $\Omega_B$ respectively achieve $f_A(T_0)$ and $f_B(\mathcal{O}_K \setminus T_0)$ in $\mathcal{N}_A$ and $\mathcal{N}_B$. Then

$$f_A(T_0) = C(\Omega_A) - \sum_{\mathcal{O}_1(j) \in \Omega_A} \ell_1(j) + R, \quad \mathcal{O}_K \cap \bar{\Omega}_A = T_0 \quad (48)$$

$$f_B(\mathcal{O}_K \setminus T_0) = C(\Omega_B) - \sum_{\mathcal{O}_M(j) \in \Omega_B} \ell_M(j) + R, \quad \mathcal{O}_K \cap \Omega_B = \mathcal{O}_K \setminus T_0. \quad (49)$$

Let $\Omega = \mathcal{O}_A \cup \mathcal{O}_B$ be a cut in network $\mathcal{N}$. Since $\mathcal{O}_K \cap \Omega_A = \mathcal{O}_K \cap \Omega_B = \mathcal{O}_K \setminus T_0$, it follows that $G_i(\Omega) = G_i(\mathcal{O}_A)$ for $i \in \{1, \cdots, K - 1\}$ and $G_i(\Omega) = G_i(\mathcal{O}_B)$ for $i \in \{K, \cdots, M - 1\}$. Thus we have

$$C(\Omega) = C(\Omega_A) + C(\Omega_B).$$

23
From (48) and (49) we have
\[ f_A(T_0) + f_B(O_K \setminus T_0) = C(\Omega) - \sum_{\sigma_1(j) \in \Omega_A} \ell_1(j) - \sum_{\sigma_M(j) \in \Omega_B} \ell_M(j) + 2R. \] (50)

Since \( \sum_{\sigma_1(j) \in \Omega_A} \ell_1(j) = \sum_{\sigma_1(j) \in \Omega} \ell_1(j) \) and \( \sum_{\sigma_1(j) \in \Omega_B} \ell_M(j) = \sum_{\sigma_M(j) \in \Omega} \ell_M(j) \), (50) and (16) together imply \( f_A(T_0) + f_B(O_K \setminus T_0) \geq R \), as desired.

To prove the converse, consider a cut \( \Omega \) in network \( \mathcal{N} \) and partition it into two cuts \( \Omega_A \) and \( \Omega_B \) in networks \( \mathcal{N}_A \) and \( \mathcal{N}_B \) respectively. Let \( T_0 = \Omega \cap O_K \). Then by definition
\[
\begin{align*}
f_A(T_0) &\leq C(\Omega_A) - \sum_{\sigma_1(j) \in \Omega_A} \ell_1(j) + R, \quad \Omega_K \cap \tilde{\Omega}_A = T_0 \\
f_B(O_K \setminus T_0) &\leq C(\Omega_B) - \sum_{\sigma_M(j) \in \Omega_B} \ell_M(j) + R, \quad \Omega_K \cap \Omega_B = O_K \setminus T_0.
\end{align*}
\]

Therefore
\[
R \leq \min_{T \subseteq \Omega_K} (f_A(T) + f_B(O_K \setminus T)) \leq f_A(T_0) + f_B(O_K \setminus T_0) \leq C(\Omega_A) + C(\Omega_B) - \sum_{\sigma_1(j) \in \Omega_A} \ell_1(j) - \sum_{\sigma_M(j) \in \Omega_B} \ell_M(j) + 2R.
\]

If we substitute \( C(\Omega_A) + C(\Omega_B) = C(\Omega) \) and \( \sum_{\sigma_1(j) \in \Omega_A} \ell_1(j) = \sum_{\sigma_1(j) \in \Omega} \ell_1(j) \) and \( \sum_{\sigma_M(j) \in \Omega_B} \ell_M(j) = \sum_{\sigma_M(j) \in \Omega} \ell_M(j) \) into the preceding expression, we obtain
\[
C(\Omega) \geq \sum_{\sigma_1(j) \in \Omega} \ell_1(j) + \sum_{\sigma_1(j) \in \Omega} \ell_M(j) - R,
\]
which is the final result.