Generalized Network Sharing Outer Bound and the Two-Unicast Problem

by Sudeep U. Kamath

Research Project

Submitted to the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, in partial satisfaction of the requirements for the degree of **Master of Science, Plan II**.

Approval for the Report and Comprehensive Examination:

Committee:

Prof. David N.C. Tse

Research Advisor

Date

* * * * * *

Prof. Venkat Anantharam

Research Advisor

Date

Contents

List of Figures

Acknowledgments

Research support from the ARO MURI grant W911NF-08-1-0233, "Tools for the Analysis and Design of Complex Multi-Scale Networks", from the NSF grant CNS-0910702, from the NSF Science & Technology Center grant CCF-0939370, "Science of Information", from Marvell Semiconductor Inc., and from the U.C. Discovery program is gratefully acknowledged.

Chapter 1

Introduction

Characterizing the capacity region of a general network is the holy grail of the subject of network information theory. A special class of networks are *directed wireline* networks where links between nodes are *unidirectional*, *orthogonal* and *noise-free*. In such networks, the complex aspects of real-world communication channels, namely broadcast, superposition, interference, noise are absent. Understanding the capacity regions of directed wireline networks would be a useful source of insight into capacity regions of general networks.

The problem of communicating from a single source to a single destination is the simplest such problem. The solution to this is provided by the Max Flow Min Cut Theorem [1]. Routing flows is sufficient to achieve the capacity of such a network. [2] introduced the problem of communicating a common message from one source to many destinations and showed that *network coding* is required, that is, that routing is insufficient to achieve the capacity, which was characterized also by the minimum cut.

On the other hand, recent results due to Dougherty, Freiling, Zeger suggest that characterizing the capacity region of a general multi-source multi-sink directed wireline network is a hard problem. They show that scalar-linear solvability of a general network is equivalent to the solvability of a general polynomial collection [3] and that linear network coding is insufficient to achieve capacity [4]. There is more bad news. Given a joint distribution on $n \geq 2$ discrete random variables $(X_1, X_2, \ldots, X_n)$, with each X_i taking values in a finite set, define its *entropy vector* as the $2^n - 1$ dimensional vector with an entry for the entropy of $\{X_i : i \in S\}$ for each non-empty subset S of $\{1, 2, \ldots, n\}$. The set of *entropic functions* Γ_n^* is defined as the subset of $\mathbb{R}^{2^n-1}$ consisting of all the vectors that are entropy vectors of some joint distribution on n random variables. [5] showed the existence of a non-Shannon information inequality, that is an inequality obeyed by all points in Γ_n^*, that did not follow from the non-negativity of conditional mutual information. Characterizing $\bar{\Gamma}_n^*$, the closure of Γ_n^* is known to be a very hard problem for $n \geq 4$. Chan and Grant [6] show that the problem of determining the achievable rate pairs (R_0, R_1) in a network with two messages and collocated sources but a large number of destinations, each destination requiring either the common message or both messages, is equivalent to the problem of characterizing the set $\bar{\Gamma}_n^*$, for each n. Their construction uses $\Omega(2^n)$ destinations for demonstrating necessity of knowledge of $\bar{\Gamma}_n^*$.

The networks presented as "counterexamples" in all the above works have three or more sources or three or more destinations. A natural question to ask is whether having fewer sources and destinations will lead to a more amenable problem. The smallest such problem is that of unicast between two sources and two destinations - each source with an independent message for its own destination. We are led to the study of the two-unicast problem, as a possible fruitful direction. The only network coding results in the literature dealing exclusively with the two-unicast network are [7] and [8], which provide the necessary and sufficient conditions for achieiving $(1, 1)$ in a two-unicast network with all links having integer capacities. This result unfortunately, relies heavily on the assumption of integer link capacities, and hence cannot give us necessary and sufficient conditions for achieving other points such as $(2, 2)$ or $(3, 3)$ by scaling of link capacities. Characterizing the complete capacity region of a given two-unicast network is an interesting direction, which is the motivation for this work.

In this thesis, we investigate the two-unicast problem. We present a simple improvement over the Network Sharing outer bound [9] for the multiple unicast problem, which we call the Generalized Network Sharing (GNS) outer bound. We observe two interesting properties of this bound related to the two-unicast problem:

- The GNS outer bound is the tightest bound that can be realized using only so-called edge-cut bounds.
- The GNS outer bound is tight for those two-unicast networks for which all edges except those from a so-called minimal GNS set have sufficiently large capacities.

These properties seem to suggest that the GNS outer bound may be tight for all two-unicast networks. Unfortunately we find that this is not the case and conclude the paper with a two-unicast "counterexample" network. The counterexample is illuminating, in that it reveals several other non-trivial attributes of the general two-unicast problem.

Chapter 2

The Generalized Network Sharing Outer Bound

2.1 Network Model

In this section, we set up notation and definitions that will be used in the rest of the thesis.

Definition: A network $\mathcal{N}$ consists of a directed acyclic graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ along with a link-capacity vector $\underline{C} = (C_e)_{e \in \mathcal{E}(\mathcal{G})}$ with $C_e \in \mathbb{R}_{\geq 0} \cup \{\infty\}$ $\forall e \in \mathcal{E}(\mathcal{G})$.

Definition: An n-unicast network ($n \geq 1$) has n distinguished vertices $s_1, s_2, \ldots, s_n$ called sources and n distinguished vertices $t_1, t_2, \ldots, t_n$ called destinations, where each source s_i has independent information to be communicated to destination t_i.

We assume, without loss of generality, that

- sources have no incoming edges and destinations have no outgoing edges, and
- every vertex $v \in \mathcal{V}(\mathcal{G})$ is reachable from at least one of the sources and can reach at least one of the destinations.

Definition: For edge $e = (v, v') \in \mathcal{E}(\mathcal{G})$, $\mathsf{tail}(e) := v$ and $\mathsf{head}(e) := v'$, the edge being directed from the tail to the head.

Definition: For $v \in \mathcal{V}(\mathcal{G})$, $\mathsf{In}(v) := \{e \in \mathcal{E} : \mathsf{head}(e) = v\}$ and $\mathsf{Out}(v) := \{e \in \mathcal{E} : \mathsf{tail}(e) = v\}$ denote the edges entering into and leaving v respectively.

Definition: For $S \subseteq \mathcal{E}(\mathcal{G})$, $C(S) := \sum_{e \in S} C_e$.

Definition: For disjoint non-empty $A, B \subseteq \mathcal{V}(\mathcal{G})$, we say $S \subseteq \mathcal{E}(\mathcal{G})$ is an $A - B$ cut if there is no directed path from any vertex in A to any vertex in B in the graph $\mathcal{G} \setminus S$.

Definition: The *mincut* from A to B is denoted $c(A; B) := \min \{C(S) : S \text{ is an } A - B \text{ cut}\}$.

Definition: We say that the rate tuple $(R_1, R_2, \ldots, R_n)$ is *achievable* for the n-unicast network $\mathcal{N} = (\mathcal{G}, \underline{\mathrm{C}})$, if there exists a positive integer N (called block length), a finite alphabet $\mathcal{A}$ with $|\mathcal{A}| \geq 2$ and encoding functions:

- for $e \in \mathsf{Out}(s_i),\ f_e : \mathcal{A}^{\lceil NR_i \rceil} \mapsto \mathcal{A}^{\lfloor NC_e \rfloor},\ 1 \leq i \leq n,$
- for $e \in \mathsf{Out}(v),\ v \neq s_i, 1 \leq i \leq n,\ f_e : \Pi_{e' \in \mathsf{In}(v)} \mathcal{A}^{\lfloor NC_{e'} \rfloor} \mapsto \mathcal{A}^{\lfloor NC_e \rfloor},$

and decoding functions $f_{t_i} : \Pi_{e' \in \mathsf{In}(t_i)} \mathcal{A}^{\lfloor NC_{e'} \rfloor} \mapsto \mathcal{A}^{\lceil NR_i \rceil},\ 1 \leq i \leq n$, with the property that

$$g_{t_i}(m_1, m_2, \ldots, m_n) = m_i,\ \ \forall (m_1, m_2, \ldots, m_n) \in \mathcal{A}^{\lceil NR_1 \rceil} \times \mathcal{A}^{\lceil NR_2 \rceil} \times \ldots \mathcal{A}^{\lceil NR_n \rceil}$$

where $g_{t_i} : \mathcal{A}^{\lceil NR_1 \rceil} \times \mathcal{A}^{\lceil NR_2 \rceil} \times \ldots \mathcal{A}^{\lceil NR_n \rceil} \mapsto \mathcal{A}^{\lceil NR_i \rceil}$ are functions induced inductively by $\{f_e : e \in \mathcal{E}(\mathcal{G})\}$ and $f_{t_i}, 1 \leq i \leq n$.

Definition: The capacity region for an n-unicast network $\mathcal{N}$, denoted $\mathcal{C}(\mathcal{N}) = \mathcal{C}(\mathcal{G}, \underline{\mathrm{C}})$, is the closure of the set of achievable rate tuples.

Definition: The vector linear coding capacity region $\mathcal{C}_{\mathsf{vector}}$ is the closure of the set of achievable rate tuples over choice of $\mathcal{A}$ as any finite field and all functions being linear operations on finite-dimensional vector spaces over the finite field for some block length N.

Definition: The scalar linear coding capacity region $\mathcal{C}_{\mathsf{scalar}}$ is the *convex* closure of the set of achievable rate tuples over choice of $\mathcal{A}$ as any finite field and all functions being linear operations on finite-dimensional vector spaces over the finite field with block length $N = 1$.

Remark: The set of achievable rate tuples by linear coding with block length $N = 1$ may not by itself be convex.

The main results in this thesis deal exclusively with two-unicast networks.

2.2 The Generalized Network Sharing Outer Bound

In this section, we first describe the Network Sharing outer bound. Following this, we describe our improvement which we call the Generalized Network Sharing (GNS) outer bound.

Theorem 1 *(Network Sharing outer bound [9]) Fix* $(i, j) = (1, 2)$ *or* $(2, 1)$*. For a two-unicast network* $\mathcal{N} = (\mathcal{G}, \underline{C})$*, if* $T \subseteq \mathcal{E}(\mathcal{G})$ *is an* $s_1, s_2 - t_1, t_2$ *cut and if* $S \subseteq T$ *is such that for each edge* $e \in T \setminus S$*, we have that* $\mathsf{tail}(e)$ *is reachable from* s_i *but not from* s_j *in* $\mathcal{G}$ *and* $\mathsf{head}(e)$ *can reach* t_j *but not* t_i *in* $\mathcal{G}$*, then we have* $R_1 + R_2 \leq C(S)\ \forall (R_1, R_2) \in \mathcal{C}(\mathcal{N})$*.*

It turns out that the outer bound technique used in [9] can in fact, be used to obtain a tighter outer bound. To describe this improvement, we first have the following definition.

Definition: A set $S \subseteq \mathcal{E}(\mathcal{G})$ is defined to be a *GNS set* if

- $\mathcal{G} \setminus S$ has no paths from s_1 to t_1, s_2 to t_2 and s_2 to t_1 OR
- $\mathcal{G} \setminus S$ has no paths from s_1 to t_1, s_2 to t_2 and s_1 to t_2.

Theorem 2 *(Generalized Network Sharing outer bound) For a two-unicast network* $\mathcal{N} = (\mathcal{G}, \underline{C})$ *and a GNS set* $S \subseteq \mathcal{E}(\mathcal{G})$*, we have* $R_1 + R_2 \leq C(S)\ \forall\ (R_1, R_2) \in \mathcal{C}(\mathcal{N})$*.*

Note that Theorem 2 implies Theorem 1.

Proof: Consider a scheme $\mathcal{S}$ of block length N achieving rate pair (R_1, R_2) over alphabet $\mathcal{A}$. Let W_1, W_2 be independent and distributed uniformly over the sets $\mathcal{A}^{\lceil NR_1 \rceil}$ and $\mathcal{A}^{\lceil NR_2 \rceil}$ respectively. For each edge e, define X_e as the concatenated evaluation of the functions specified by the scheme $\mathcal{S}$ for edge e, where X_e takes values in $\mathcal{A}^{\lfloor NC_e \rfloor}$.

Let $X_S := (X_e)_{e \in S}$. Then, $H(W_1, W_2 | X_S) = H(W_1 | X_S) + H(W_2 | W_1, X_S)$. But, $H(W_1 | X_S) = 0$ because $\mathcal{G} \setminus S$ has no paths from s_1 or s_2 to t_1. And $H(W_2 | W_1, X_S) = 0$ because

$\mathcal{G} \setminus S$ has no paths from s_2 to t_2. Thus, $H(W_1, W_2|X_S) = 0$. So,

$$\begin{aligned} N \cdot \log|\mathcal{A}| \cdot (R_1 + R_2) &\leq H(W_1) + H(W_2) \\ &= H(W_1, W_2) \\ &\leq H(X_S) \\ &\leq \sum_{e \in S} H(X_e) \\ &\leq \sum_{e \in S} N \cdot \log|\mathcal{A}| \cdot C_e \\ &= N \cdot \log|\mathcal{A}| \cdot C(S). \end{aligned}$$

As the inequality holds for every achievable rate pair, it also holds for every point in the closure of the set of achievable rate pairs. ∎

This theorem leads to the following natural definitions for a two-unicast network $\mathcal{N}$.

Definition: The *GNS sum-rate bound* $c_{\mathsf{gns}}(s_1, s_2; t_1, t_2)$ is defined as $c_{\mathsf{gns}}(s_1, s_2; t_1, t_2) := \min\{C(S) : S \subseteq \mathcal{E}(\mathcal{G}) \text{ is a GNS set}\}$.

Definition: The *GNS outer bound* $\mathcal{C}_{\mathsf{GNS}}$ is defined as the region $\{(R_1, R_2) : R_1 \leq c(s_1; t_1), R_2 \leq c(s_2; t_2), R_1 + R_2 \leq c_{\mathsf{gns}}(s_1, s_2; t_1, t_2)\}$.

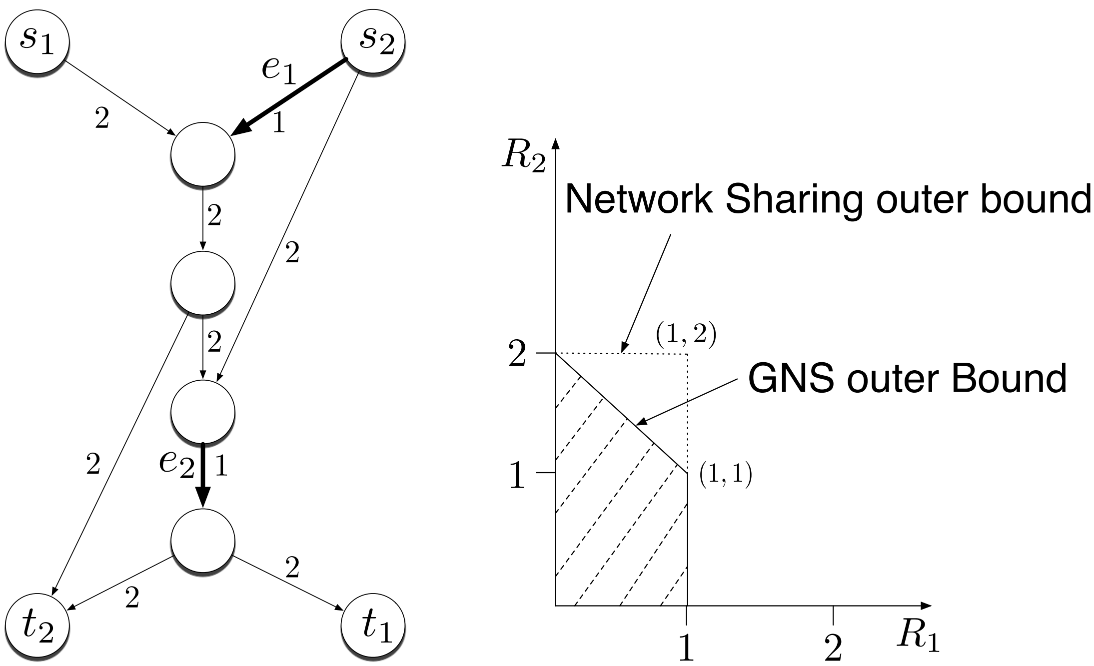
For a given two-unicast network $\mathcal{N}$, the GNS sum-rate bound is a number while the GNS outer bound is a region. The following generalization of Theorem 2 may be proved similarly.

Theorem 3 *Consider an n-unicast network $\mathcal{N} = (\mathcal{G}, \underline{C})$. For non-empty $I \subseteq \{1, 2, \ldots, n\}$ and $S \subseteq \mathcal{E}(\mathcal{G})$, suppose there exists a bijection $\pi : I \mapsto \{1, 2, \ldots, |I|\}$ such that $\forall\, i, j \in I$, $\mathcal{G} \setminus S$ has no paths from source s_i to destination t_j whenever $\pi(i) \geq \pi(j)$. Then,*

$$\sum_{i \in I} R_i \leq C(S) \quad \forall (R_1, R_2, \ldots, R_n) \in \mathcal{C}(\mathcal{N}).$$

The GNS outer bound is a special case of the edge-cut bounds in [10]. However, it is simpler and much more explicit. Moreover, we will show in Theorem 5 that the GNS outer bound is the tightest possible outer bound resulting from edge-cut bounds for two-unicast networks and is thus, equivalent to the bound in [10] for two-unicast networks. The GNS outer bound is also a special case of the even stronger LP bound [11], which is the tighest outer bound obtainable using

Shannon information inequalities. In Section 3.3, we will show that the LP bound is in general tighter than the GNS outer bound for two-unicast networks. However, the GNS outer bound can be strictly better than the Network Sharing outer bound [9] as shown in Fig. 2.1.



(a) Grail with variable capacities. e_1, e_2 have unit capacity and all other edges have capacity 2 units. $\{e_1, e_2\}$ is a GNS set.

(b) Network Sharing outer bound and GNS outer bound for network in (a). Capacity region of this network matches the GNS outer bound.

Figure 2.1: The GNS outer bound can be strictly better than the Network Sharing outer bound

Chapter 3

Properties with respect to the Two-Unicast Problem

The previous chapter described an improvement over the Network Sharing outer bound, which we called the Generalized Network Sharing outer bound. The present chapter provides a closer look at this outer bound with regard to the two-unicast problem. In Section 3.1, we show that it is the tightest bound in the class of edge-cut bounds. In Section 3.2, we show that when all edges outside a *minimal* GNS set have sufficiently large capacities, the GNS outer bound is tight. Finally, in Section 3.3, we will provide a "counterexample" network to show that the GNS outer bound is not tight in general.

3.1 Tightest outer bound resulting from edge-cut bounds

Definition: For an uncapacitated two-unicast network $\mathcal{G}$, an inequality of the form $\alpha_1 R_1 + \alpha_2 R_2 \leq C(S)$, with $\alpha_1, \alpha_2 \in \{0, 1\}, S \subseteq \mathcal{E}(\mathcal{G})$ is called an *edge-cut bound* if the inequality holds for all $(R_1, R_2) \in \mathcal{C}(\mathcal{G}, \underline{\mathrm{C}})$, for each choice of $\underline{\mathrm{C}}$.

The cutset outer bound, the Network Sharing outer bound [9] and the GNS outer bound all consist of edge-cut bounds. Further, the Network Sharing outer bound is an improvement over the cutset bound and the GNS outer bound is an improvement over the Network Sharing outer bound. In Theorem 5, we show that it is impossible to improve on the GNS outer bound using edge-cut bounds for two-unicast networks. Towards proving this theorem, we will first state and prove a useful result.

Theorem 4 *(Two-Multicast Theorem) For a two-multicast network $\mathcal{N} = (\mathcal{G}, \underline{C})$ with sources s_1 and s_2 multicasting independent messages at rates R_1 and R_2 respectively to both the destinations t_1 and t_2, (R_1, R_2) is an achievable rate pair if and only if*

$$R_1 \leq \min\{c(s_1; t_1), c(s_1; t_2)\}$$

$$R_2 \leq \min\{c(s_2; t_1), c(s_2; t_2)\}$$

$$R_1 + R_2 \leq \min\{c(s_1, s_2; t_1), c(s_1, s_2; t_2)\}$$

Proof: The necessity of these conditions is obvious. For proving sufficiency, fix a rate pair (R_1, R_2) that satisfies these conditions and consider a new network $\tilde{\mathcal{N}}$ obtained by adding a super-source s with two outgoing edges to s_1 and s_2 with link capacities R_1 and R_2 respectively (see Fig. 3.1).

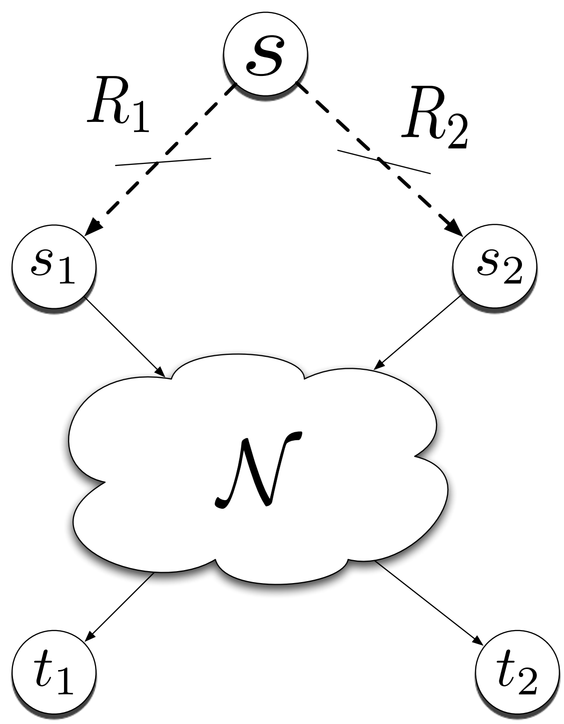


Figure 3.1: Network $\tilde{\mathcal{N}}$

We use the single source multicast result ([2], [12]) on $\tilde{\mathcal{N}}$ to infer the existence of a scheme for s multicasting at rate $R_1 + R_2$ to the destinations t_1 and t_2. This allows us to construct a two-multicast scheme in the original network $\mathcal{N}$ achieving the desired rate pair. ∎

Theorem 5 *Let $\mathcal{G}$ be an uncapacitated two-unicast network, and let $S \subseteq \mathcal{E}(\mathcal{G})$ such that $R_1+R_2 \leq C(S)$ is an edge-cut bound. If S is not a GNS set, then $c(s_1;t_1)+c(s_2;t_2) \leq C(S)$ for all choices of $\underline{C}$.*

Remark: $R_1 \leq c(s_1;t_1)$ cannot be improved upon because the rate pair $(c(s_1;t_1),0)$ is achievable. Similarly, $R_2 \leq c(s_2;t_2)$ cannot be improved upon. Thus, Theorem 5 says that the GNS outer bound

$$\mathcal{C}_{\mathsf{GNS}} = \{(R_1,R_2) : R_1 \leq c(s_1;t_1), R_2 \leq c(s_2;t_2), R_1+R_2 \leq c_{\mathsf{gns}}(s_1,s_2;t_1,t_2)\}$$

is the tightest outer bound that can be derived from edge-cut bounds for the two-unicast problem.

Proof: Suppose $R_1+R_2 \leq C(S)$ holds for all $(R_1,R_2) \in \mathcal{C}(\mathcal{G},\underline{C})$ for all choices of $\underline{C}$. Then, it must be that S is an s_i-t_i cut for $i=1,2$, so that $\mathcal{G} \setminus S$ has no paths from s_1 to t_1 or s_2 to t_2. If $\mathcal{G} \setminus S$ has no paths from s_1 to t_2 also, then S is a GNS set and the outer bound follows from Theorem 2. Likewise if $\mathcal{G} \setminus S$ has no paths from s_2 to t_1.

So, suppose that $\mathcal{G} \setminus S$ has no paths from s_1 to t_1 or s_2 to t_2 but it has paths from s_1 to t_2 and s_2 to t_1. Define $C_i(S) := \min\{C(T) : T \subseteq S,\ T \text{ is an } s_i - t_i \text{ cut}\}$ for $i=1,2$. Fix any choice of non-negative reals $\{c_e : e \in S\}$. Consider the following choice of link capacities:

$$C_e = \begin{cases} c_e & \forall e \in S \\ \infty & \forall e \notin S \end{cases}.$$

Note that for this choice of link capacities, $c(s_i;t_i) = C_i(S), i=1,2$.

By Theorem 4, (R_1,R_2) is achievable for two-multicast from s_1,s_2 to t_1,t_2 if and only if $R_1 \leq C_1(S)$ and $R_2 \leq C_2(S)$, since $c(s_1,s_2;t_1) \geq c(s_2;t_1) = \infty, c(s_1,s_2;t_2) \geq c(s_1;t_2) = \infty$. Thus, $(C_1(S),C_2(S))$ is achievable for two-multicast and hence, also for two-unicast. Since $R_1+R_2 \leq C(S)$ holds for all $(R_1,R_2) \in \mathcal{C}(\mathcal{G},\underline{C})$, we must have

$$C_1(S)+C_2(S) \leq C(S) \ \ \forall\{c_e : e \in S\}. \tag{3.1}$$

This is a purely graph theoretic property about the structure of the set of edges S relative to the uncapacitated network $\mathcal{G}$.

Now, for an arbitrary assignment of link capacities $\underline{C}$, we have by definition, $c(s_1;t_1) \leq C_1(S)$ and $c(s_2;t_2) \leq C_2(S)$. Using (3.1) gives us $c(s_1;t_1)+c(s_2;t_2) \leq C(S)$. ∎

Remark: Fig. 3.2 provides an example of an uncapacitated two-unicast network $\mathcal{G}$. Consider different sets $S_i \subseteq \mathcal{E}, i=1,2,3$ such that $\mathcal{G} \setminus S_i$ has no paths from s_1 to t_1 or from s_2 to t_2.

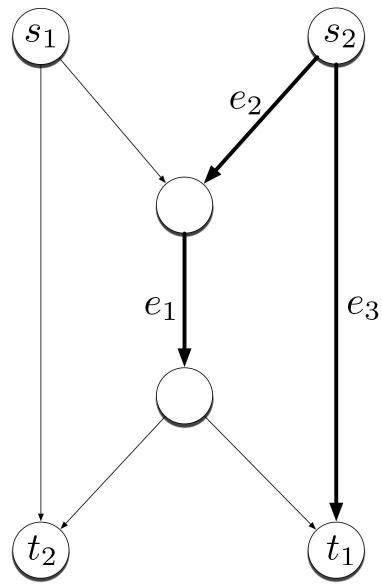


Figure 3.2: The butterfly network

- $S_1 = \{e_1, e_3\}$ is a GNS set. $R_1 + R_2 \leq C_{e_1} + C_{e_3}$ follows from Theorem 2.
- $S_2 = \{e_1, e_2\}$ is not a GNS set, but $R_1 + R_2 \leq C_{e_1} + C_{e_2}$ is an edge-cut bound (this may be established using [10]). However, it is a trivial edge-cut bound since it follows from the cutset bounds: $R_1 \leq C_{e_1}, R_2 \leq C_{e_2}$.
- $S_3 = \{e_1\}$. For all links having unit capacity, we have that $(1, 1)$ is achievable by network coding and this violates $R_1 + R_2 \leq C_{e_3}$. This shows that $R_1 + R_2 \leq C_{e_3}$ is not an edge-cut bound.

Every set $S \subseteq \mathcal{E}$ that is not a GNS set either provides a trivial edge-cut bound on the sum rate or does not provide an edge-cut bound on the sum rate, i.e. the GNS sets together provide all possible non-trivial edge-cut bounds on the sum rate.

3.2 Tightness in special cases

The previous section looked at the GNS outer bound from the perspective of edge-cut bounds [10]. In this section, we consider the GNS outer bound in terms of achievability. If a GNS set S contains a proper GNS subset S', then clearly the outer bound $R_1 + R_2 \leq C(S)$ is "loose" in that, it could be improved to $R_1 + R_2 \leq C(S')$. However, what about a minimal GNS set, i.e. a GNS set with no proper GNS subset? Theorem 6 establishes that every minimal GNS set provides an outer bound that is not "obviously loose", i.e. if all edges outside a minimal GNS set have infinite capacity, then the GNS outer bound matches the capacity region.

Theorem 6 *Let $\mathcal{G}$ be an uncapacitated two-unicast network and let $S \subseteq \mathcal{E}(\mathcal{G})$ be a minimal GNS set. Choose an arbitrary collection of non-negative reals $\{c_e : e \in S\}$. Consider the following link-capacity-vector:*

$$C_e = \begin{cases} c_e & \forall e \in S \\ \infty & \forall e \notin S \end{cases}.$$

Then, for the two-unicast network $(\mathcal{G}, \underline{C})$, the GNS outer bound

$$\{(R_1, R_2) : R_1 \leq c(s_1; t_1), R_2 \leq c(s_2; t_2), R_1 + R_2 \leq c_{\mathsf{gns}}(s_1, s_2; t_1, t_2)\}$$

is identical to the capacity region $\mathcal{C}(\mathcal{G}, \underline{C})$, i.e. the GNS outer bound is tight.

Remark: Note that Theorem 6 does not say that a sum rate of $c_{\mathsf{gns}}(s_1, s_2; t_1, t_2) = C(S)$ is achievable, only that all rate pairs in the region $\{(R_1, R_2) : R_1 \leq c(s_1; t_1), R_2 \leq c(s_2; t_2), R_1 + R_2 \leq c_{\mathsf{gns}}(s_1, s_2; t_1, t_2)\}$ are achievable. A sum rate of $C(S)$ is achievable only when $C(S) \leq c(s_1; t_1) + c(s_2; t_2)$ for the choice of capacities.

Proof: Define $C_i(S) := \min\{C(T) : T \subseteq S,\ T \text{ is an } s_i - t_i \text{ cut}\}$ for $i = 1, 2$ as in the proof of Theorem 5. As S is a minimal GNS set, the GNS outer bound for $(\mathcal{G}, \underline{C})$ is given by

$$R_1 \leq C_1(S), R_2 \leq C_2(S), R_1 + R_2 \leq C(S). \tag{3.2}$$

We will assume that c_e is an integer for each $e \in S$ and describe scalar linear coding schemes over the binary field $\mathbb{F}_2$ with block length $N = 1$ achieving the GNS outer bound. Having done this, it is easy to see that the theorem would also hold for choice of non-negative rational and thus, also non-negative real choice of $c_e, e \in S$. Henceforth, we will imagine a link of capacity c_e as having c_e unit capacity edges connected in parallel. This change could be made in the graph and in this proof, we will use $\mathcal{G}$ to denote the graph with all edges having unit capacity, possibly having multiple edges in parallel connecting two vertices.

Note that if S is a GNS set and a proper subset $S' \subset S$ is also a GNS set, then for any $e \in S \setminus S'$, we have that $S \setminus e$ is a GNS set. Conversely, if $S \setminus e$ is a GNS set for some edge $e \in S$, then there exists a proper subset $S' \subset S$ such that S' is a GNS set, simply choose $S' = S \setminus e$. Thus, a GNS set S is minimal if and only if for each $e \in S$, we have that $S \setminus e$ is not a GNS set. This allows us to partition the edges in S by their connectivity in $\mathcal{G} \setminus \{S \setminus e\}$ as $S_1^1 \cup S_1^2 \cup S_1^{12} \cup S_2^1 \cup S_2^2 \cup S_2^{12} \cup S_{12}^1 \cup S_{12}^2 \cup S_{12}^{12}$ where $e \in S$ lies in S_x^y if, in the graph $\mathcal{G} \setminus \{S \setminus e\}$, $\mathsf{tail}(e)$ is reachable only from source indices x and $\mathsf{head}(e)$ is capable of reaching only destination indices y. Eg. S_{12}^2 contains edge e in S if and only if in $\mathcal{G} \setminus \{S \setminus e\}$, we have that $\mathsf{tail}(e)$ is reachable from s_1, s_2 and $\mathsf{head}(e)$ can reach t_2, but cannot reach t_1.

Define $\hat{S}_1 := S_1^1 \cup S_1^{12} \cup S_{12}^1 \cup S_{12}^{12}$ and $\hat{S}_2 := S_2^2 \cup S_2^{12} \cup S_{12}^2 \cup S_{12}^{12}$. Thus, $\hat{S}_i$, for $i = 1, 2$ is the set of edges in S which have their tails reachable from s_i by a path of infinite capacity and their heads reaching t_i by a path of infinite capacity. We will show $C_i(S) = C(\hat{S}_i) + c_{\mathcal{G}\setminus\hat{S}_i}(s_i; t_i)$, for $i = 1, 2$. By the Max Flow Min Cut Theorem, there exists a flow of value $C_i(S)$ from s_i to t_i in $\mathcal{G}$. At most $C(\hat{S}_i)$ of the flow goes through edges in $\hat{S}_i$. Thus, there exists a flow of value at least $C_i(S) - C(\hat{S}_i)$ in $\mathcal{G} \setminus \hat{S}_i$. So, $c_{\mathcal{G}\setminus\hat{S}_i}(s_i; t_i) \geq C_i(S) - C(\hat{S}_i)$. Now, consider $T_i \subseteq S$ in $\mathcal{G}$ such that T_i is an $s_i - t_i$ cut and $C(T_i) = C_i(S)$. Then, since $\hat{S}_i \subseteq T_i$, we have that $T_i \setminus \hat{S}_i$ is an $s_i - t_i$ cut in $\mathcal{G} \setminus \hat{S}_i$. Thus, $c_{\mathcal{G}\setminus\hat{S}_i}(s_i; t_i) \leq C(T_i \setminus \hat{S}_i) = C(T_i) - C(\hat{S}_i) = C_i(S) - C(\hat{S}_i)$.

Case I: S is a minimal GNS set such that $\mathcal{G} \setminus S$ has no paths from either of s_1, s_2 to t_1, t_2. In this case, $S_1^2, S_2^1 = \emptyset$ by minimality of S. Thus, $C_1(S) + C_2(S) \geq C(\hat{S}_1) + C(\hat{S}_2) = C(S) + C(S_{12}^{12}) \geq C(S)$. So, in this case, the GNS outer bound (3.2) is a pentagonal region and we have to show achievability of the two corner points $(C_1(S), C(S) - C_1(S))$ and $(C(S) - C_2(S), C_2(S))$.

Consider the following scheme. Edges in $S_1^1, S_1^{12}, S_{12}^1, S_{12}^{12}$ forward s_1's message bits to t_1 and edges in $S_2^2, S_{12}^2, S_2^{12}$ forward s_2's message bits to t_2. This achieves

$$R_1 = C(\hat{S}_1) = C(S_1^1) + C(S_1^{12}) + C(S_{12}^1) + C(S_{12}^{12}),$$
$$R_2 = C(S_2^2) + C(S_{12}^2) + C(S_2^{12}).$$

Note that we have $R_1 + R_2 = C(S)$ for this rate pair. Now, we will increase R_1 up to $C_1(S)$ while preserving this sum rate. Construct $c_{\mathcal{G}\setminus\hat{S}_1}(s_1; t_1)$ unit capacity "edge-disjoint" paths from s_1 to t_1 in $\mathcal{G} \setminus \hat{S}_1$. This gives us $c_{\mathcal{G}\setminus\hat{S}_1}(s_1; t_1)$ paths in $\mathcal{G}$ such that none of them use any edge in $\hat{S}_1$. Any such path encounters a first finite capacity edge from S_{12}^2 and a last finite capacity edge from S_2^{12}. The intermediate finite capacity edges, if any, may be assumed to lie in S_2^2 only. If intermediate finite capacity edges lie in S_{12}^2 or S_2^{12}, we can modify the path so that this is not the case, while preserving the edge-disjointness property. This can be done as follows. Suppose we have a path $\mathcal{P}$. Consider the last finite capacity edge $e \in S_{12}^2$ along the path. Now, use an infinite capacity path $\mathcal{P}_1$ from s_1 to reach $\mathsf{tail}(e)$. Find the earliest finite capacity edge $e' \in S_2^{12}$ after e on the path $\mathcal{P}$. Find an infinite capacity path $\mathcal{P}_2$ from head(e') to t_1. Then, the path generated from concatenating $\mathcal{P}_1$, the part of the path $\mathcal{P}$ between and including e and e' and finally, $\mathcal{P}_2$ gives a path from s_1 to t_1 with the desired property.

Consider any such path. A simple XOR coding scheme as shown in Fig. 3.3(a) shows that it is possible to modify the earlier scheme so that R_1 is improved by one bit and R_2 is reduced by one bit. We have $e_1 \in S_{12}^2, e_2 \in S_2^2, e_3 \in S_2^{12}$. s_2 was sending three bits to t_2 when s_1 wasn't transmitting along the chosen unit capacity path. Now, s_1 gets one bit delivered to t_1 while s_2 gets two bits delivered to t_2 instead of three, since s_2 has to set $b_1 \oplus b_2 \oplus b_3 = 0$ for allowing t_1 to decode a.

In the general case, we have an arbitrary number of finite capacity edges from S_2^2 along the path (possibly none). Perform a similar XOR scheme to improve R_1 by one bit and lower R_2 by one bit, as shown in Fig. 3.3(b) and Fig. 3.3(c). Because the paths are "edge-disjoint", the finite capacity edges on those paths are all distinct, so the imposed constraints can all be met by reducing R_2 by one bit for each such path. When this is carried out for each of the $c_{\mathcal{G}\setminus\hat{S}_1}(s_1;t_1)$ paths, we have a scheme achieving $(C_1(S), C(S)-C_1(S))$. Similarly, the other corner point $(C(S)-C_2(S), C_2(S))$ may be shown to be achievable.

Case II: S is a minimal GNS set such that $\mathcal{G} \setminus S$ has no paths from s_1 to t_1, s_2 to t_2, or s_2 to t_1 but it has paths from s_1 to t_2. As S is a minimal GNS set, we have $S_1^2 = \emptyset$. In this case, the GNS outer bound (3.2) is not necessarily a pentagonal region. We first show achievability of the rate pair $R_1 = C_1(S), R_2 = \min\{C_2(S), C(S) - C_1(S)\}$.

Stage I - Basic Scheme: It is easy to see that we can achieve the rate pair given by

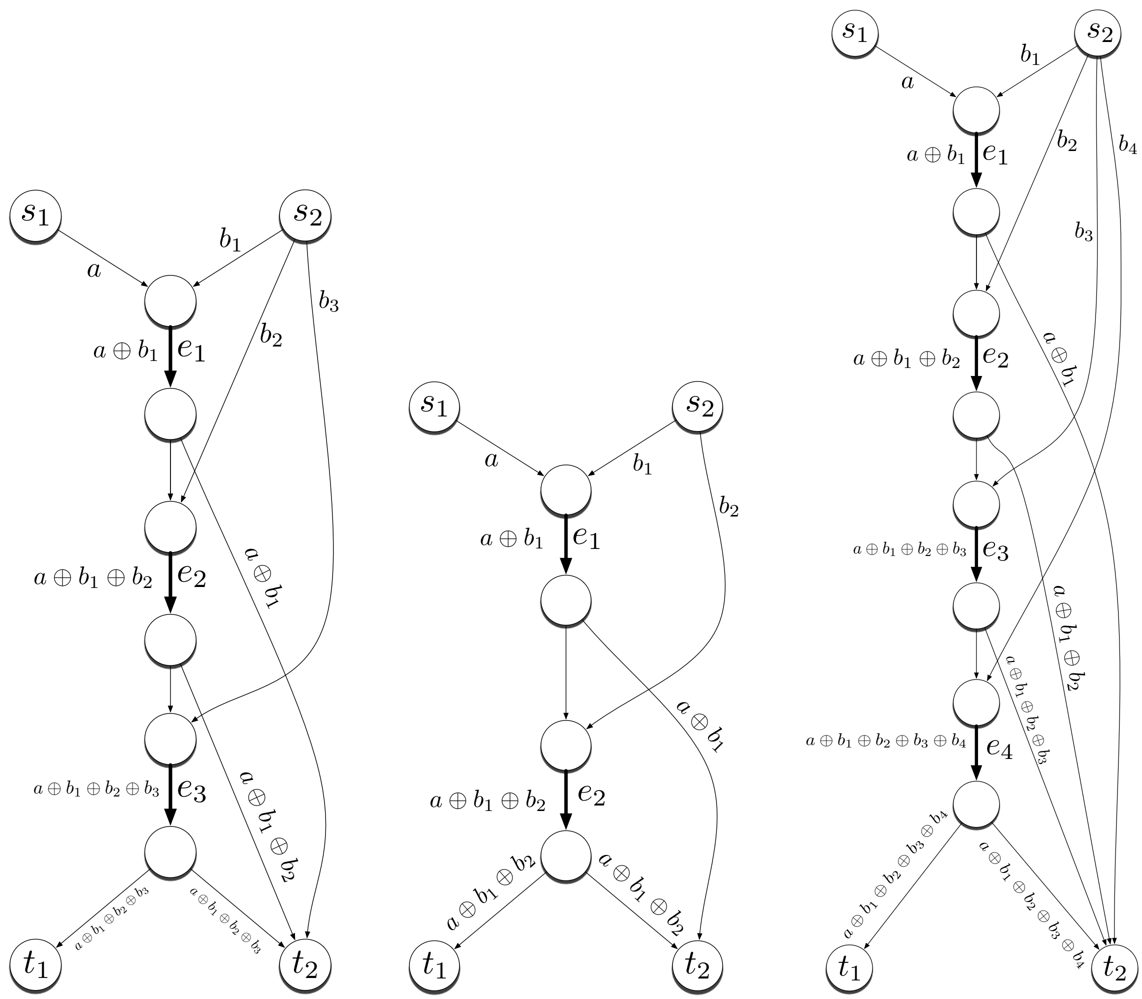
$$R_1 = C(\hat{S}_1) = C(S_1^1) + C(S_{12}^1) + C(S_1^{12}) + C(S_{12}^{12}),$$
$$R_2 = C(S_2^2) + C(S_2^{12}) + C(S_{12}^2) + \min\{C(S_2^1), C(S_{12}^{12})\},$$

by a routing + butterfly coding approach as follows.

- Edges in $S_1^1, S_1^{12}, S_{12}^1$ forward s_1's message bits to t_1 and edges in $S_2^2, S_2^{12}, S_{12}^2$ forward s_2's message bits to t_2.

- Edges in S_{12}^{12} and S_2^1 along with an infinite capacity path from s_1 to t_2 perform "preferential routing for s_1 with butterfly coding for s_2," i.e.

  - if $C(S_2^1) < C(S_{12}^{12})$, then an amount of $C(S_{12}^{12}) - C(S_2^1)$ of the capacity of edges in S_{12}^{12} is used for routing s_1's message bits, while the rest is used for butterfly coding, i.e.



(a) Three finite capacity edges, s_2 sets $b_1 \oplus b_2 \oplus b_3 = 0$

(b) Two finite capacity edges, s_2 sets $b_1 \oplus b_2 = 0$

(c) Four finite capacity edges, s_2 sets $b_1 \oplus b_2 \oplus b_3 \oplus b_4 = 0$

Figure 3.3: Case I: Improving R_1 by one bit while reducing R_2 by one bit

a bitwise XOR operation is performed over $C(S_2^1)$ bits from source s_1 with $C(S_2^1)$ bits from source s_2 to be transmitted over the edges in S_{12}^{12}. Edges in S_2^1 provide $C(S_2^1)$ bits of side-information from s_2 to t_1, while the infinite capacity path from s_1 to t_2 provides side-information to t_2.

– if $C(S_2^1) \geq C(S_{12}^{12})$, then all of the capacity of edges in S_{12}^{12} is used for butterfly coding. An amount of $C(S_2^1) - C(S_{12}^{12})$ of the capacity of edges in S_2^1 is left unused.

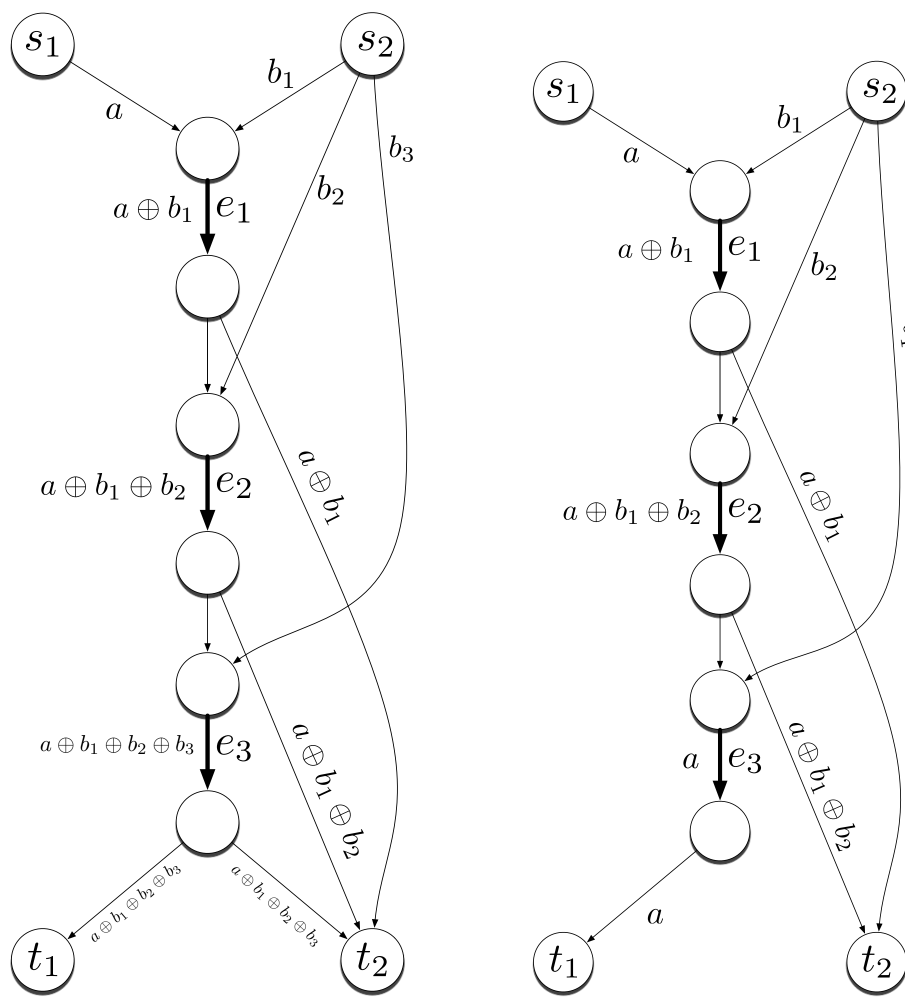
Stage II - Improving R_1 up to $C_1(S)$: We know $c_{\mathcal{G}\setminus\hat{S}_1}(s_1;t_1) = C_1(S) - C(\hat{S}_1)$. Find $c_{\mathcal{G}\setminus\hat{S}_1}(s_1;t_1)$ unit capacity "edge-disjoint" paths from s_1 to t_1 in $\mathcal{G}$ such that none of them use any edge in $\hat{S}_1$. Each such unit capacity path from s_1 to t_1 in $\mathcal{G}$ starts with a first finite capacity edge in S_{12}^2, ends with the last finite capacity edge in S_2^{12} or S_2^1 and with all intermediate edges lying, without loss of generality, in S_2^2. Use each such path to increase R_1 by one bit as follows.

- If the last finite capacity edge lies in S_2^{12}, perform coding as in Fig. 3.4(a). If the capacity of S_2^1 edges is not fully used, use free unit capacity of some edge $e \in S_2^1$ to relay the XOR value of $b_1 \oplus b_2 \oplus b_3$ from s_2 to t_1. Use the infinite capacity path from s_1 to t_2 to send the symbol a. Now, R_1 is improved by 1 bit while R_2 remains unchanged. If the capacity of S_2^1 edges is fully used, note that we have achieved a sum rate of $C(S)$ as all links are carrying independent linear combinations of source message bits. Then, s_2 sets $b_1 \oplus b_2 \oplus b_3 = 0$. This increases R_1 by one bit and reduces R_2 by one bit.

- Suppose the last finite capacity edge, call it e_3, lies in S_2^1.

  If e_3 is not being used, perform coding as in Fig. 3.4(b). Use the infinite capacity path from s_1 to t_2 to relay the symbol a. Then we use free edge $e_3 \in S_2^1$ to carry a which will improve R_1 by one bit while R_2 remains unchanged. If e_3 is being used, it must be used as a conduit for side-information to t_1, as part of the butterfly coding. If there is some edge $e \in S_2^1$ that is not being used, then we could use e to serve as a conduit for the side-information to t_1. This frees up e_3 and we can use the same scheme described in Fig. 3.4(b).

  If all edges in S_2^1 are used, then we must have achieved a sum rate of $C(S)$. Edge e_3 now relays a to t_1 improving R_1 by one bit. However, the edge e_3 must have been assisting in butterfly coding using some edge in S_{12}^{12} and the infinite capacity $s_1 - t_2$ path. Now, the edge e_3 can no longer provide side-information to t_1. So, the corresponding unit capacity in some edge in S_{12}^{12} now performs routing of s_1's message bit as opposed to XOR mixing of one bit



(a) Last finite capacity edge in S_2^{12} (b) Last finite capacity edge in S_2^{1}

Figure 3.4: Improving R_1 up to $C_1(S)$

of s_1's message and one bit of s_2's message. This reduces R_2 by one bit.

This can be carried out for each of the $c_{\mathcal{G}\setminus\hat{S}_1}(s_1;t_1)$ paths sequentially because they are "edge-disjoint". This stage achieves $R_1 = C_1(S)$. If the capacity of S_2^1 edges is all used up, we have achieved a sum rate $R_1 + R_2 = C(S)$ and so, $R_2 = C(S) - C_1(S)$ and we are done. So, suppose there is some residual capacity left in edges of S_2^1. This means that R_1 was improved to $C_1(S)$ while R_2 was not reduced from $C(\hat{S}_2)$.

Stage III - Improving R_2 up to $\min\{C(S) - C_1(S), C_2(S)\}$: We have $C_2(S) = C(\hat{S}_2) + c_{\mathcal{G}\setminus\hat{S}_2}(s_2;t_2)$. Similar to before, we find $c_{\mathcal{G}\setminus\hat{S}_2}(s_2;t_2)$ unit capacity "edge-disjoint" paths from s_2 to t_2 in $\mathcal{G}$ such that the paths don't use any edge in $\hat{S}_2$. Each such unit capacity path encounters a first finite capacity edge from S_{12}^1 or S_2^1 and a last finite capacity edge from S_1^{12} while all intermediate finite capacity edges may be assumed to lie in S_1^1. Note that edges in $S_1^1, S_1^{12}, S_{12}^1$ are all performing pure routing of s_1's message. At any point, if the capacity of edges in S_2^1 is fully used, we have reached $R_1 = C_1(S), R_2 = C(S) - C_1(S)$. If the capacity is not fully used, perform the modification as described below for the $c_{\mathcal{G}\setminus\hat{S}_2}(s_2;t_2)$ unit capacity "edge-disjoint" paths from s_2 to t_2, one at a time, till we hit $R_2 = C_2(S)$ or till all S_2^1 edges are used, upon which we have $R_2 = C(S) - C_1(S)$.

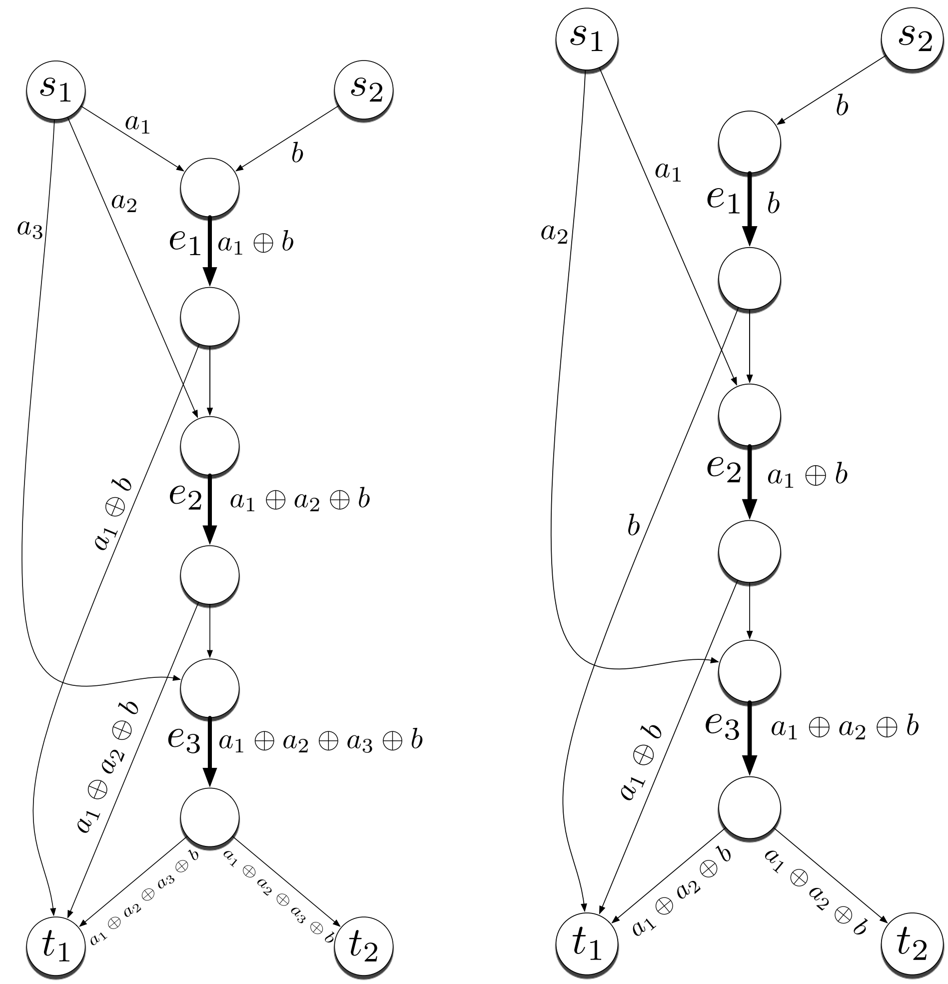
- If the first finite capacity edge lies in S_{12}^1, perform coding as in Fig. 3.5(a). Use unit capacity of some free edge in S_2^1 to relay symbol b from s_2 to t_1 and use the s_1 to t_2 infinite capacity path to send the XOR value of $a_1 \oplus a_2 \oplus a_3$ to t_2. This leaves R_1 unaffected and improves R_2 by one bit.

- Suppose the first finite capacity edge, call it e_1, lies in S_2^1.

  If e_1 is not being used, perform coding as in Fig. 3.5(b). Use unit capacity of edge $e_1 \in S_2^1$ to send a symbol b from s_2 to t_1. The infinite capacity s_1 to t_2 path is used to send $a_1 \oplus a_2$ from s_1 to t_2. This allows t_2 to decode b and improves R_2 by one bit while leaving R_1 unaffected.

  If e_1 is being used for sending side-information to t_1 (as part of the butterfly coding or in Stage II for providing side-information when the last finite capacity edge lies in S_2^{12}), then pick some edge $e \in S_2^1$ that is not being used and use e for sending the side-information to t_1. This frees up e_1 and we can use the scheme described in Fig. 3.5(b).

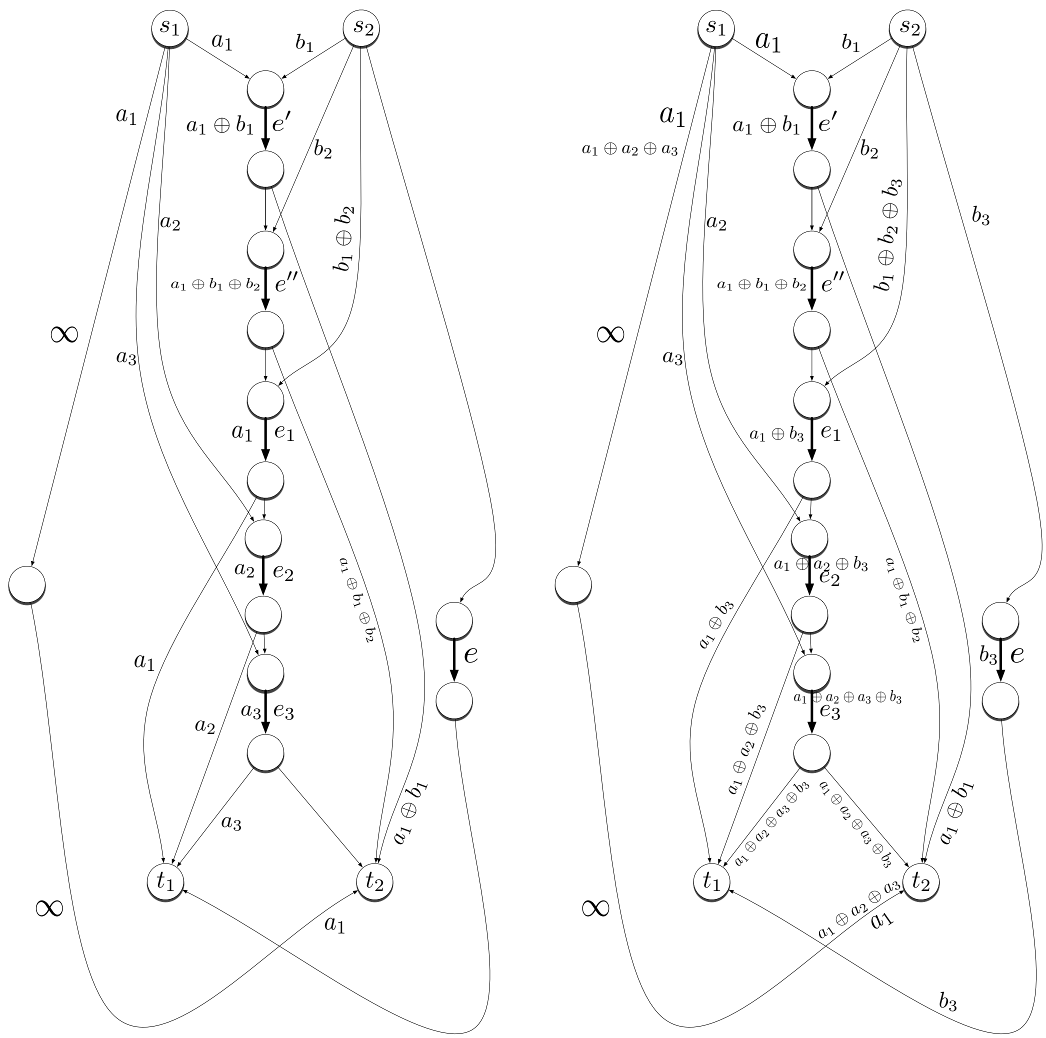
  If e_1 is being used but not for sending side-information, it must have gotten used in Stage II as the last finite capacity edge on an $s_1 - t_1$ path. It could not have been used in Stage III as in



(a) First finite capacity edge in S^1_{12}

(b) First finite capacity edge in S^1_2 and the edge not used in Stage II

Figure 3.5: Improving R_2 up to $\min\{C(S) - C_1(S), C_2(S)\}$



(a) e', e'', e_1 are being used in Stage II. e_2, e_3 serve to route s_1's bits to t_1.

(b) Chosen s_2-t_2 path uses edges e_1, e_2, e_3. Modified scheme uses some free edge $e \in S_2^1$.

Figure 3.6: Improving R_2 up to $\min\{C(S) - C_1(S), C_2(S)\}$ in the case when e_1 was already being used in Stage II.

Fig. 3.5(b), because all the paths chosen in Stage III are "edge-disjoint". In this case, we use some free edge $e \in S_2^1$ and superimpose scheme shown in Fig. 3.5(b) with already existing scheme Fig. 3.4(b). This modification is shown via Fig. 3.6(a) and Fig. 3.6(b). This improves R_2 by one bit while R_1 remains unchanged.

This stage terminates achieving $R_1 = C_1(S), R_2 = \min\{C_2(S), C(S) - C_1(S)\}$. Because the GNS set is not symmetric in indices 1 and 2, we also have to show achievability of the rate pair $R_1 = \min\{C_1(S), C(S) - C_2(S)\}, R_2 = C_2(S)$.

Stage I - Basic Scheme: It is easy to see achievability of the rate pair given by

$$R_1 = C(S_1^1) + C(S_{12}^1) + C(S_1^{12}) + \min\{C(S_{12}^{12}), C(S_2^1)\},$$
$$R_2 = C(\hat{S}_2) = C(S_2^2) + C(S_2^{12}) + C(S_{12}^2) + C(S_{12}^{12})$$

by a routing + butterfly coding approach as follows.

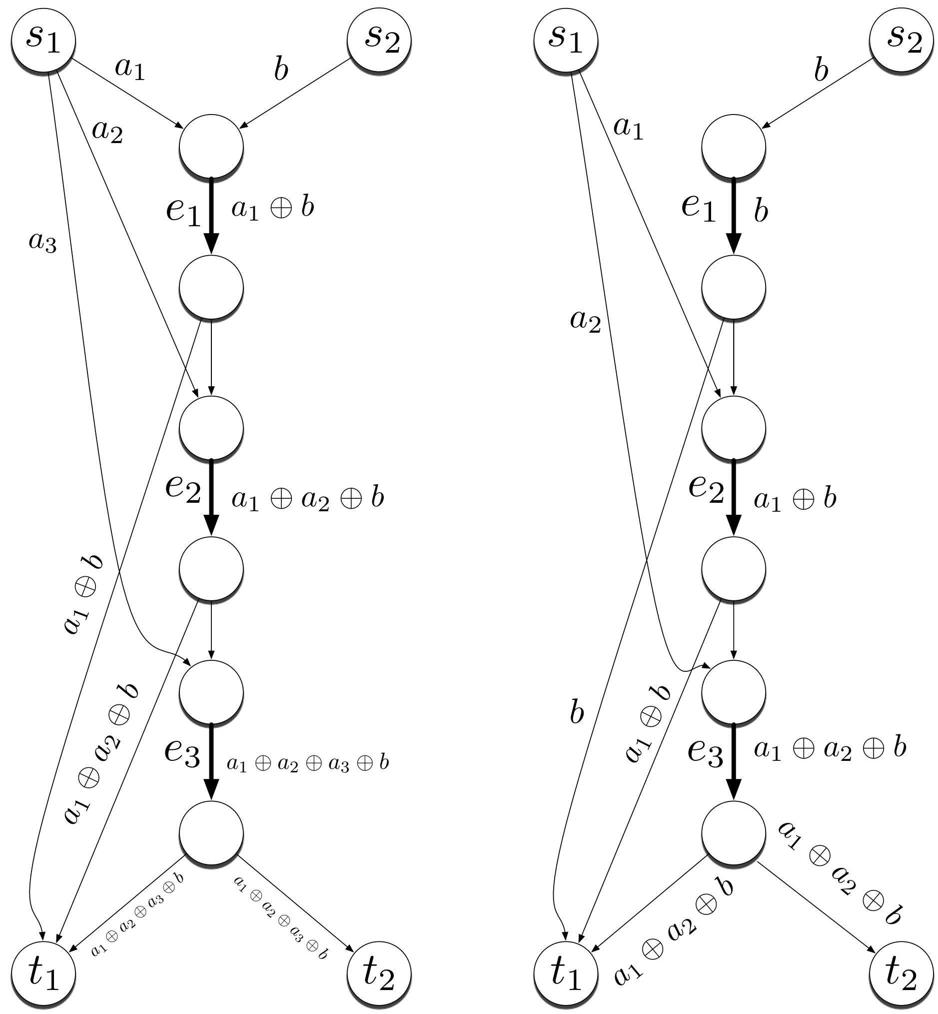
- Edges in $S_2^2, S_2^{12}, S_{12}^2$ forward s_2's message bits to t_2.
- Edges in $S_1^1, S_1^{12}, S_{12}^1$ forward s_1's message bits to t_1.
- Edges in S_{12}^{12} and S_2^1 along with an infinite capacity path from s_1 to t_2 perform "preferential routing for s_2 with butterfly coding for s_1."

Stage II - Improving R_2 up to $C_2(S)$: As before, we have $c_{\mathcal{G}\backslash\hat{S}_2}(s_2; t_2) = C_2(S) - C(\hat{S}_2)$. Find $c_{\mathcal{G}\backslash\hat{S}_2}(s_2; t_2)$ unit capacity "edge-disjoint" paths from s_2 to t_2 in $\mathcal{G}$ such that none of them use any edge in $\hat{S}_2$. Now, we will modify the above scheme and use each such path to increase R_2 by one bit in a way so as to inflict minimum damage to R_1. Each such unit capacity path from s_2 to t_2 in $\mathcal{G}$ starts with a first finite capacity edge in S_{12}^1 or S_2^1 and ends with the last finite capacity edge in S_1^{12}, with all intermediate edges lying, without loss of generality, in S_1^1.

- If the first finite capacity edge lies in S_{12}^1, perform coding as in Fig. 3.7(a).

  If the capacity of S_2^1 edges is not fully used, use free unit capacity of some edge $e \in S_2^1$ to relay the bit b from s_2 to t_1. Use the infinite capacity path from s_1 to t_2 to send the XOR value of $a_1 \oplus a_2 \oplus a_3$. Now, R_2 is improved by 1 bit while R_1 remains unchanged.

  If the capacity of S_2^1 edges is fully used, we have achieved a sum rate of $C(S)$. Then s_1 sets $a_1 \oplus a_2 \oplus a_3 = 0$. This increases R_2 by one bit and reduces R_1 by one bit.



(a) First finite capacity edge in S_{12}^1

(b) First finite capacity edge in S_2^1

Figure 3.7: Improving R_2 up to $C_2(S)$

- Suppose the first finite capacity edge, call it e_1, lies in S_2^1.

  If e_1 is not being used, perform coding as in Fig. 3.7(b). Use free edge $e_1 \in S_2^1$ to relay b and the infinite capacity path from s_1 to t_2 to send $a_1 \oplus a_2$. This improves R_2 by one bit while R_1 remains unchanged. If e_1 is being used, it must be used for providing side-information to t_1, as part of the butterfly coding. If there is some edge $e \in S_2^1$ that is not being used, then we could use e to convey this side-information to t_1. This frees up e_1 and we can use the same scheme described in Fig. 3.7(b).

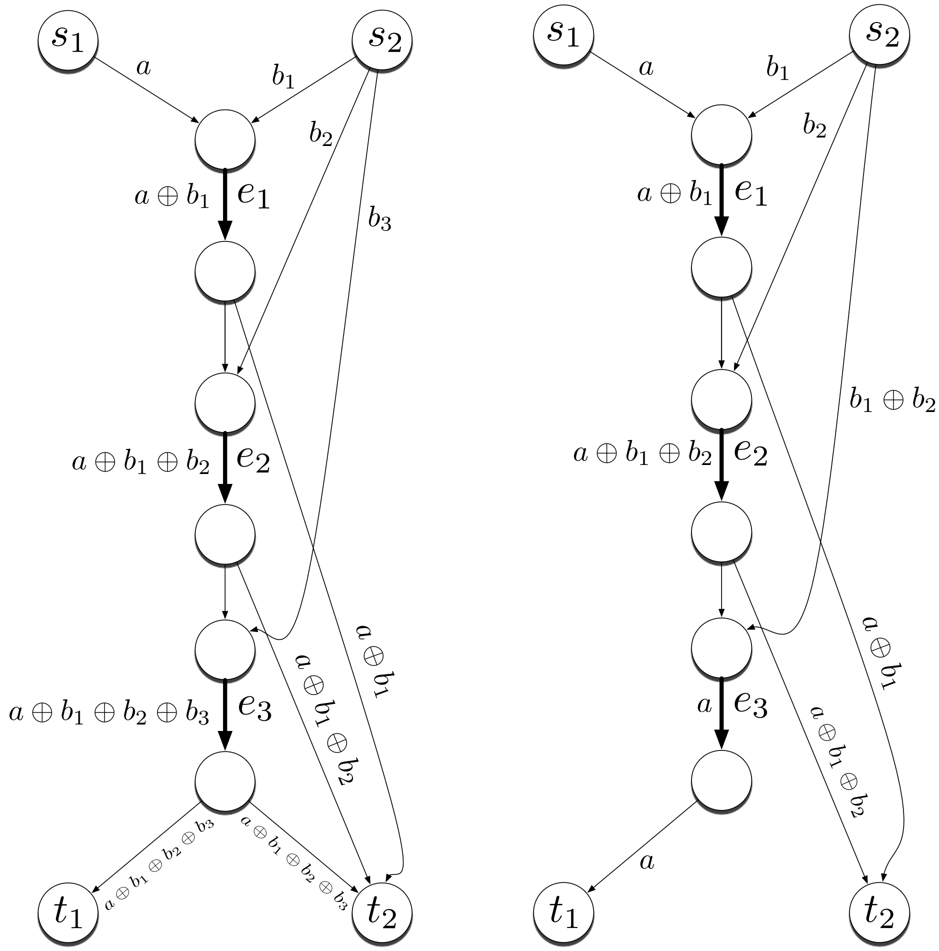
  If the capacity of S_2^1 edges is fully used, we must have achieved a sum rate of $C(S)$. Edge e_1 now relays b to t_1 improving R_2 by one bit. However, the edge e_1 must have been assisting in butterfly coding using some edge in S_{12}^{12} and the infinite capacity $s_1 - t_2$ path. Now, the edge e_1 can no longer provide side-information to t_1. So, the corresponding unit capacity in some edge in S_{12}^{12} now performs routing of s_2's message bit as opposed to XOR mixing of one bit of s_1's message and one bit of s_2's message. This reduces R_1 by one bit.

This stage achieves $R_2 = C_2(S)$. If the capacity of S_2^1 edges is all used up, we have achieved a sum rate of $R_1 + R_2 = C(S)$ and so, $R_1 = C(S) - C_2(S)$ and we are done. So, suppose there is some residual capacity left in edges of S_2^1. This means that R_2 was improved to $C_2(S)$ while R_1 was not reduced from $C(\hat{S}_1)$.

Stage III - Improving R_1 up to $\min\{C(S) - C_2(S), C_1(S)\}$:

We know $C_1(S) = C(\hat{S}_1) + c_{\mathcal{G}\setminus\hat{S}_1}(s_1; t_1)$. Find $c_{\mathcal{G}\setminus\hat{S}_1}(s_1; t_1)$ unit capacity "edge-disjoint" paths from s_1 to t_1 in $\mathcal{G}$ such that the paths don't use any edge in $\hat{S}_1$. Each such unit capacity path encounters a first finite capacity edge from S_{12}^2 and a last finite capacity edge from S_2^{12} or S_2^1 while all intermediate finite capacity edges may be assumed to lie in S_2^2. Note that edges in $S_2^2, S_2^{12}, S_{12}^2$ are all performing pure routing of s_2's message bits. At any point, if the capacity of S_2^1 edges is fully used, we have reached $R_1 = C(S) - C_2(S), R_2 = C_2(S)$. If the capacity is not fully used, perform the modification as described below for the $c_{\mathcal{G}\setminus\hat{S}_1}(s_1; t_1)$ unit capacity "edge-disjoint" paths from s_1 to t_1, one at a time, till we hit $R_1 = C_1(S)$ or till all S_2^1 edges are used, upon which we have $R_1 = C(S) - C_2(S)$.

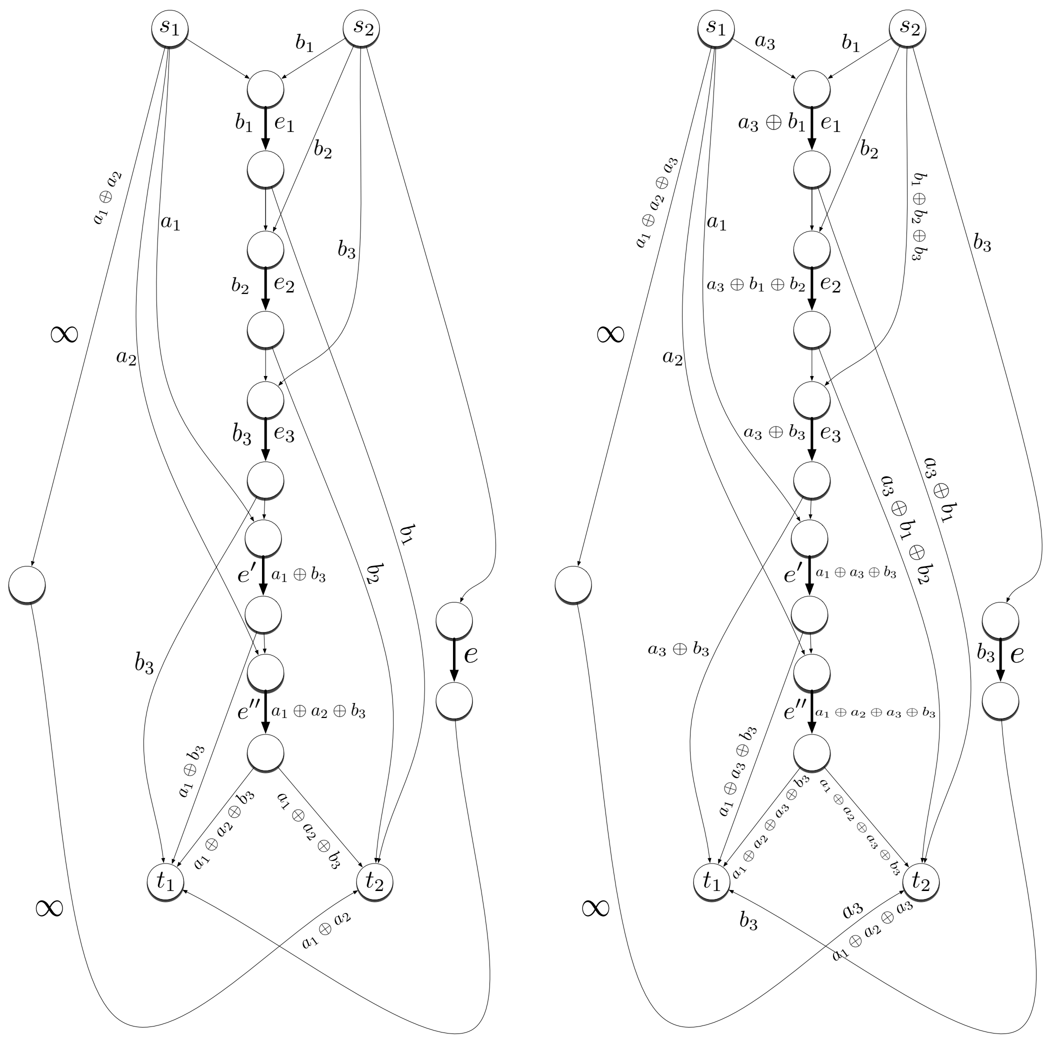
- If the last finite capacity edge lies in S_2^{12}, perform coding as in Fig. 3.8(a). Use unit capacity of an edge in S_2^1 to relay the XOR value of $b_1 \oplus b_2 \oplus b_3$ from s_2 to t_1 and use the s_1 to t_2 infinite capacity path to send the bit a to t_2. This leaves R_2 unaffected and improves R_1 by one bit.



(a) Last finite capacity edge in S_{12}^1

(b) Last finite capacity edge in S_2^1

Figure 3.8: Improving R_1 up to $\min\{C(S) - C_2(S), C_1(S)\}$



(a) e_3, e', e'' are being used in Stage II. e_1, e_2 serve to route s_2's bits to t_2.

(b) Chosen s_1-t_1 path uses edges e_1, e_2, e_3. Modified scheme uses some free edge $e \in S_2^1$.

Figure 3.9: Improving R_1 up to $\min\{C(S) - C_2(S), C_1(S)\}$ in the case when e_3 was already being used in Stage II.

- Suppose the last finite capacity edge, call it e_3, lies in S_2^1.

  If e_3 is not being used, perform coding as in Fig. 3.8(b). The infinite capacity s_1 to t_2 path is used to send a from s_1 to t_2. This improves R_1 by one bit while leaving R_2 unaffected.

  If e_3 is being used for sending side-information to t_1 (as part of the butterfly coding or in Stage II for providing side-information when the first finite capacity edge lies in S_{12}^1), then pick some edge $e \in S_2^1$ that is not being used and use e for sending the side-information to t_1. This frees up e_3 and we can use the scheme described in Fig. 3.8(b).

  If e_3 is being used but not for sending side-information, it must have gotten used in Stage II as the first finite capacity edge on an $s_2 - t_2$ path. It could not have been used in Stage III as in Fig. 3.8(b), because all the paths chosen in Stage III are "edge-disjoint". In this case, we use some free edge $e \in S_2^1$ and superimpose scheme shown in Fig. 3.8(b) with already existing scheme Fig. 3.7(b). This modification is shown via Fig. 3.9(a) and Fig. 3.9(b). This improves R_1 by one bit while R_2 remains unchanged.

This stage terminates achieving $R_1 = \min\{C_1(S), C(S) - C_2(S)\}, R_2 = C_2(S)$.

Case III: S is a minimal GNS set such that $\mathcal{G} \setminus S$ has no paths from s_1 to t_1, s_2 to t_2, or s_1 to t_2 but it has paths from s_2 to t_1. This case is identical to Case II. ∎

Remark: Theorem 6 also holds when C_e for $e \notin S$ are all finite but sufficiently large, i.e. when $C_e \geq C(S)\ \forall e \notin S$. This can be concluded from the proof by using the fact that the scheme is linear over the binary field $\mathbb{F}_2$.

3.3 GNS outer bound is not tight

We now provide an example of a two-unicast network showing that,

- the GNS outer bound is not tight, so edge-cut bounds do not suffice to characterize the capacity region,
- the trade-off between rates on the boundary of the capacity region need not be 1:1,
- the capacity region may have a non-integral corner point even if all links have integer capacity, and thus,
- scalar linear coding is not sufficient to achieve capacity.

Consider the network shown in Fig. 3.10(a). $\{e_1, e_2, e_4\}$ and $\{e_1, e_3, e_4\}$ are the two GNS sets. The GNS sum-rate bound is $c_{\mathsf{gns}}(s_1, s_2; t_1, t_2) = 3$. The GNS outer bound is therefore,

$$\mathcal{C}_{\mathsf{GNS}} = \{(R_1, R_2) : R_1 \leq 1, R_2 \leq 2\}.$$

$c(s_2; t_2) = 2$, so the rate pair $(0, 2)$ is achievable. Fig. 3.10(b) shows a two time step vector linear coding scheme over $\mathbb{F}_2$ that achieves $(1, 1.5)$.

Next, we will prove the inequality $R_1 + 2R_2 \leq 4$ for any rate pair (R_1, R_2) in the capacity region of this network. Consider a scheme of block length N over alphabet $\mathcal{A}$ achieving the rate pair (R_1, R_2). Let W_1, W_2 be independent and distributed uniformly over the sets $\mathcal{A}^{\lceil NR_1 \rceil}$ and $\mathcal{A}^{\lceil NR_2 \rceil}$ respectively. For edge $e = e_1, e_2, e_3, e_4$, define X_e as the concatenated evaluation of the functions specified by the scheme for edge e.

$$H(W_1)$$

$$= I(X_{e_1}, X_{e_2}, X_{e_4}; W_1) + H(W_1 | X_{e_1}, X_{e_2}, X_{e_4}) \tag{3.3}$$

$$= I(X_{e_1}, X_{e_2}; W_1) + I(X_{e_4}; W_1 | X_{e_1}, X_{e_2}) + 0 \tag{3.4}$$

$$I(X_{e_1}, X_{e_2}; W_1)$$

$$= I(X_{e_1}, X_{e_2}; W_1, W_2) - I(X_{e_1}, X_{e_2}; W_2 | W_1) \tag{3.5}$$

$$= H(X_{e_1}, X_{e_2}) - H(W_2 | W_1) + H(W_2 | W_1, X_{e_1}, X_{e_2}) \tag{3.6}$$

$$= H(X_{e_1}, X_{e_2}) - H(W_2) + 0 \tag{3.7}$$

$$I(X_{e_4}; W_1 | X_{e_1}, X_{e_2})$$

$$= I(X_{e_4}; W_1, X_{e_1}, X_{e_2}) - I(X_{e_4}; X_{e_1}, X_{e_2}) \tag{3.8}$$

$$\leq H(X_{e_4}) - I(X_{e_4}; W_2) \tag{3.9}$$

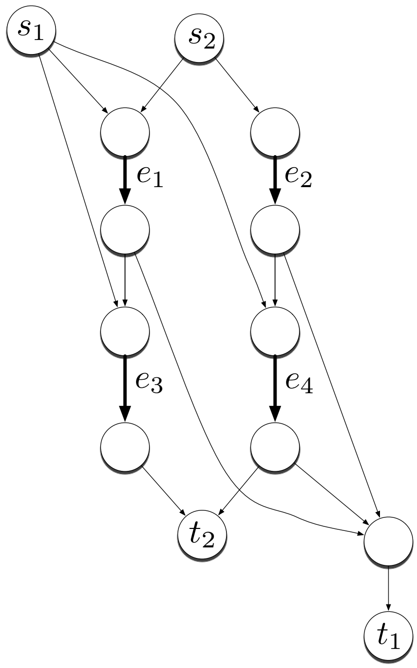
$$= H(X_{e_4}) - I(X_{e_3}, X_{e_4}; W_2) + I(X_{e_3}; W_2 | X_{e_4}) \tag{3.10}$$

$$\leq H(X_{e_4}) - H(W_2) + H(X_{e_3} | X_{e_4}) \tag{3.11}$$

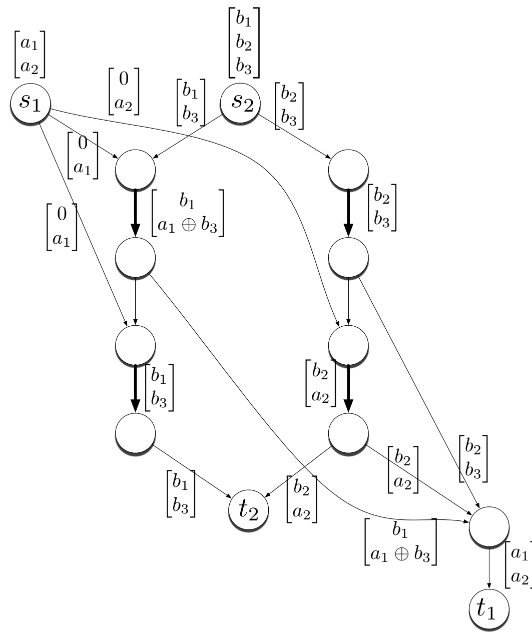
$$= H(X_{e_3}, X_{e_4}) - H(W_2) \tag{3.12}$$

(3.4) follows from $\{e_1, e_2, e_4\}$ being an $s_1, s_2 - t_1$ cut, (3.7) follows from $\{e_1, e_2\}$ being an $s_2 - t_2$ cut.

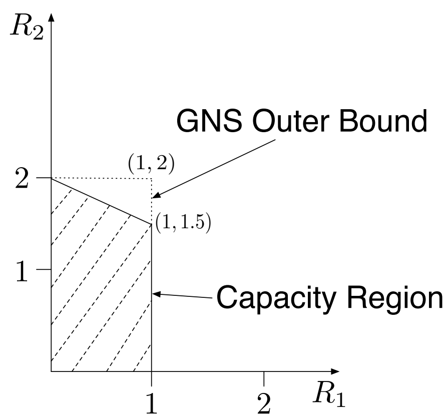
Thus, we have $N \cdot \log |\mathcal{A}| \cdot (R_1 + 2R_2) \leq H(W_1) + 2H(W_2) \leq H(X_{e_1}, X_{e_2}) + H(X_{e_3}, X_{e_4}) \leq 4N \cdot \log |\mathcal{A}|$.



(a) GNS counterexample: e_1, e_2, e_3, e_4 have unit capacity and the rest have infinite capacity



(b) Vector linear scheme over $\mathbb{F}_2$ achieving (1,1.5)



(c) GNS outer bound and capacity region of the network in Fig (a)

Figure 3.10: Counterexample to tightness of the GNS outer bound

Thus, the network has a capacity region as shown in Fig. 3.10(c). The capacity region of this network is strictly contained within the GNS outer bound.

Chapter 4

Conclusion

We described an improvement over the Network Sharing Outer Bound [9] which we called the Generalized Network Sharing outer bound. From the point of view of outer bounds, we showed that the GNS outer bound is the tightest among the edge-cut bounds for the two-unicast problem. From the achievability point of view, we showed that the GNS outer bound is achievable when all edges, except those from a minimal GNS set, have sufficiently large capacities. Finally, we showed that the GNS outer bound is not tight for two-unicast networks by a counterexample.

4.1 Interesting open questions

For a general n-unicast network, the LP bound [11] (denoted by, say $\mathcal{C}_{\mathsf{LP}}$) is the tightest outer bound that can be obtained only using Shannon information inequalities. The GNS outer bound $\mathcal{C}_{\mathsf{GNS}}$ for the n-unicast network is the collection of all possible inequalities that may be obtained using Theorem 3. The GNS outer bound is a special case of the LP bound and so we have

$$\mathcal{C}_{\mathsf{scalar}} \subseteq \mathcal{C}_{\mathsf{vector}} \subseteq \mathcal{C} \subseteq \mathcal{C}_{\mathsf{LP}} \subseteq \mathcal{C}_{\mathsf{GNS}}.$$

[4] shows that $\mathcal{C}_{\mathsf{vector}} \subsetneq \mathcal{C}$ and [13] shows $\mathcal{C} \subsetneq \mathcal{C}_{\mathsf{LP}}$ for general n-unicast networks. The network in Fig. 3.10(a) shows that for two-unicast networks, $\mathcal{C}_{\mathsf{scalar}} \subsetneq \mathcal{C}_{\mathsf{vector}}$ and $\mathcal{C}_{\mathsf{LP}} \subsetneq \mathcal{C}_{\mathsf{GNS}}$ in general. It would be interesting to know whether or not

- $\mathcal{C}_{\mathsf{vector}} \subsetneq \mathcal{C}$
- $\mathcal{C} \subsetneq \mathcal{C}_{\mathsf{LP}}$

for a general two-unicast network.

It would also be interesting to know whether the GNS outer bound for the n-unicast problem obtained using Theorem 3, is the tightest that can be realized only using edge-cut bounds, i.e. whether an analog of Theorem 5 holds for n-unicast networks for $n \geq 3$.

Bibliography

[1] P. Elias, A. Feinstein, and C.E. Shannon, "A note on the maximum flow through a network", *IEEE Transactions on Information Theory*, vol. IT-2, pp. 117–119, December 1956.

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

[3] 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.

[4] 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.

[5] Z. Zhang and R.W. Yeung, "On characterization of entropy function via information inequalities", *IEEE Transactions on Information Theory*, vol. 44, no. 4, pp. 1440–1452, July 1998.

[6] T. Chan and A. Grant, "Mission impossible: Computing the network coding capacity region", in *Proc. of IEEE ISIT*, Toronto, Canada, July 2008.

[7] Chih-Chun Wang and Ness B. Shroff, "Pairwise intersession network coding on directed networks", *IEEE Transactions on Information Theory*, vol. 56, no. 8, pp. 3879–3900, August 2010.

[8] Kai Cai, K.B. Letaief, Pingyi Fan, and Rongquan Feng, "On the solvability of 2-pair unicast networks: A cut-based characterization", *arXiv:1007.0465 [cs.IT]*, July 2010.

[9] X. Yan, J. Yang, and Z. Zhang, "An outer bound for multisource multisink network coding with minimum cost consideration", *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2373–2385, June 2006.

[10] G. Kramer and S. Savari, "Edge-cut bounds on network coding rates", *Journal of Network and Systems Management*, vol. 14, no. 1, pp. 49–67, March 2006.

[11] R.W. Yeung and Z. Zhang, "Distributed source coding for satellite communications", *IEEE Transactions on Information Theory*, vol. 45, no. 4, pp. 1111–1120, May 1999.

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

[13] R. Dougherty, C. Freiling, and K. Zeger, "Networks, matroids and non-shannon information inequalities", *IEEE Transactions on Information Theory*, vol. 53, no. 6, pp. 1949–1969, June 2007.