The Multicast Capacity of Acyclic, Deterministic Relay Networks with No Interference

Niranjan Ratnakar and Gerhard Kramer

Abstract—**A class of deterministic relay networks with no interference is considered. Such networks were studied by M. R. Aref, who showed that the unicast capacity has a max-flow, min-cut interpretation. These networks are here called Aref networks. It is shown that the multicast capacity of acyclic Aref networks also has a max-flow, min-cut interpretation.**

I. Introduction

Consider a network represented by a directed graph $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ and $\mathcal{E}$ are the sets of vertices and directed edges respectively. For example, the graph might represent a communication network where the vertices are terminals and the edges are channels. In this paper, we are interested in *deterministic* relay networks with *no interference*. These networks were introduced by Aref [1] and we call such networks Aref networks. Such networks have one input X_u associated with every vertex u, and one output Y_{uv} associated with every edge (u, v). By *deterministic*, we mean that $Y_{u,v}$ is some deterministic function of X_u. This restriction clearly permits *broadcasting*, since the outgoing edges of a vertex share a common input. By *no interference*, we mean that $Y_{u,v}$ is a function of X_u *only*. We are further interested in the *multicast* scenario where one message is to be transmitted from one vertex to one or more other vertices. The maximum rate at which one can transmit is called the *multicast capacity*.

For example, a deterministic wired network with independent channels is a special case of Aref networks by collecting the inputs to all the outgoing edges from a particular vertex as a vector $\underline{X}$, and by viewing $\underline{X}$ as a common input. The multicast capacity of deterministic wired networks networks was determined in [2] and was shown to have a max-flow, min-cut interpretation.

A more general problem is when X_u is the input of an arbitrary broadcast channel. The thesis [1] determined the *unicast* capacity of such networks, i.e., the case when there is exactly one source and one destination. The paper [4] studies another model with similar yet distinct assumptions. Both types of models are an intermediate step towards wireless networks which further suffer from interference at a receiver, fading and other phenomena.

N. Ratnakar is with the Coordinated Science Laboratory, University of Illinois at Urbana Champaign, 1308 West Main St., Urbana IL 61801 (email: ratnakar@ifp.uiuc.edu).

G. Kramer is with Bell Laboratories, Lucent Technologies, 600 Mountain Ave, Murray Hill NJ 07974 (email: gkr@bell-labs.com).

The work of N. Ratnakar was partially supported by the Vodafone foundation. N. Ratnakar wishes to thank Bell Labs for the summer internship opportunity. The work of G. Kramer was partially supported by the Board of Trustees of the University of Illinois Subaward No. 04-217 under NSF Grant No. CCR-0325673.

The main contribution of this paper is to characterize the multicast capacity of acyclic Aref networks.

II. Preliminaries

A. Robust Typicality

In order to prove the main result, we use the notion of robust typicality of sequences [5]. Let X be a random variable distributed over a finite set $\mathcal{X}$ according to a probability distribution p. Let the *support set* S_X be defined as $S_X \overset{def}{=} \{x \in \mathcal{X} : p(x) > 0\}$ and let $\mu_X = \min_{x \in S_X} p(x)$ be its smallest nonzero probability. Let $\mathbf{X} = X_1, X_2, \ldots, X_n$ be a sequence of independent random copies of X. The *empirical frequency* of $x \in \mathcal{X}$ in a sequence $\mathbf{x} = x_1, \ldots, x_n \in \mathcal{X}^n$ is

$$\nu_{\mathbf{x}}(x) \overset{def}{=} \frac{|\{i : x_i = x\}|}{n}.$$

The sequence $\mathbf{x}$ is δ-*robustly typical*(δ-*r.t.*) for $\delta > 0$ if for all $x \in \mathcal{X}$,

$$|\nu_{\mathbf{x}}(x) - p(x)| \leq \delta \cdot p(x)$$

The set of δ-r.t. $\mathbf{x}$'s is denoted $T_\delta(X)$. Let X and Y be random variables. Let the *support set* $S_{X,Y}$ be defined as $S_{X,Y} \overset{def}{=} \{(x, y) \in \mathcal{X} \times \mathcal{Y} : p(x, y) > 0\}$ and let $\mu_{X,Y} = \min_{(x,y) \in S_{X,Y}} p(x, y)$ be its smallest nonzero probability.

Let $\mathbf{x} \in \mathcal{X}^n$ and $\delta > 0$ let

$$T_\delta(Y|\mathbf{x}) \overset{def}{=} \{\mathbf{y} : (\mathbf{x}, \mathbf{y}) \in T_\delta(X, Y)\}$$

We list the following lemmas which are proved in the appendix of [5].

Lemma 1: Let $0 < \delta \leq 1$ and

$$\epsilon_\delta(n) \overset{def}{=} 2|S_X| e^{-\delta^2 \mu_X n/3}. \tag{1}$$

We have,

$$(1 - \epsilon_\delta(n)) \cdot 2^{(1-\delta)H(X)n} \leq |T_\delta(X)| \leq 2^{(1+\delta)H(X)n}$$

Lemma 2: Let $0 < \delta_1 < \delta_2 \leq 1$. Let $\epsilon_{\delta_1,\delta_2}(n) \overset{def}{=} 2|S_{X,Y}| e^{-\frac{(\delta_2-\delta_1)^2}{1+\delta_1} \cdot \mu_{X,Y} n/3}$. In the following, the upper bound holds for every $\mathbf{x} \in \mathcal{X}^n$ and the lower bound holds for every $\mathbf{x} \in T_{\delta_1}(X)$.

$$\begin{aligned}(1 - \epsilon_{\delta_1,\delta_2}(n)) \cdot 2^{(1-\delta_2)H(Y|X)n} &\leq |T_{\delta_2}(Y|\mathbf{x})| \\ &\leq 2^{(1+\delta_2)H(Y|X)n}\end{aligned}$$

Note that $\epsilon_\delta(n)$ and $\epsilon_{\delta_1,\delta_2}(n)$ diminish to zero exponentially with n.

Lemma 3: Let $y \in \mathcal{Y}$ be a random variable such that $y = f(x)$ where f is a deterministic function. Let $\mathbf{x} \in \mathcal{X}^n$ such that $\mathbf{x} \in T_\delta(X)$. Let $\mathbf{y} = (f(x_1), f(x_2), \ldots, f(x_n))$. We have $\mathbf{y} \in T_\delta(Y)$ and $(\mathbf{x}, \mathbf{y}) \in T_\delta(X, Y)$

Proof: For every y in $\mathcal{Y}$, define the set inv(y) as follows:

$$\text{inv}(y) = \{x \in \mathcal{X} : f(x) = y\}$$

Now, $p(y) = \sum_{x \in \text{inv}(y)} p(x)$. Similarly $\nu_{\mathbf{y}}(y) = \sum_{x \in \text{inv}(y)} \nu_{\mathbf{x}}(x)$. Since $\mathbf{x} \in T_\delta(X)$, we have, $|\nu_{\mathbf{x}}(x) - p(x)| \leq \delta \cdot p(x)$ for all $x \in \mathcal{X}$. For any $y \in \mathcal{Y}$, consider

$$\begin{aligned}
|\nu_{\mathbf{y}}(y) - p(y)| &= \left| \sum_{x \in \text{inv}(y)} (\nu_{\mathbf{x}}(x) - p(x)) \right| \\
&\leq \sum_{x \in \text{inv}(y)} |\nu_{\mathbf{x}}(x) - p(x)| \\
&\leq \sum_{x \in \text{inv}(y)} \delta \cdot p(x) \\
&= \delta \cdot p(y)
\end{aligned}$$

Thus, it follows that $\mathbf{y} \in T_\delta(Y)$.

Consider a pair $(x, y) \in \mathcal{X} \times \mathcal{Y}$. If $y = f(x)$, we have, $\nu_{(\mathbf{x},\mathbf{y})}(x, y) = \nu_{\mathbf{x}}(x)$ and $p(x, y) = p(x)$. Thus, it follows that $|\nu_{(\mathbf{x},\mathbf{y})}(x, y) - p(x, y)| \leq \delta \cdot p(x, y)$.

If $y \neq f(x)$, $\nu_{(\mathbf{x},\mathbf{y})}(x, y) = p(x, y) = 0$. Thus, it follows that $(\mathbf{x}, \mathbf{y}) \in T_\delta(X, Y)$. ■

B. Network Model

We describe the network model. Consider the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ described above. The random variable X_u associated with vertex u has discrete and finite alphabet $\mathcal{X}_u$. We write $p_{X_1}(\cdot)$ for the distribution of X_1, and $p_{X_1}(x_1)$ or $p(x_1)$ for the probability that $X_1 = x_1$. Similarly, the random variable $Y_{u,v}$ associated with edge (u, v) has discrete and finite alphabet $\mathcal{Y}_{u,v}$. We write $Y_{u,v} = h_{u,v}(X_u)$, where the function $h_{u,v}(\cdot)$ has domain $\mathcal{X}_u$ and range $\mathcal{Y}_{u,v}$. The network is *clocked*, i.e., each vertex and edge is activated simultaneously N times, and at every time instant vertex u transmits a symbol x_u, $x_u \in \mathcal{X}_u$, and receives symbols $y_{w,u} = h_{w,u}(x_w)$, where w is any vertex for which there is an edge (w, u) in $\mathcal{E}$.

As an example, consider the Aref network in Figure 1. Suppose vertex 1 represents a source and $T = \{6, 7\}$ a set of destination vertices. The main difference between an Aref network and a network of point to point links considered in [2, 3] is captured by the output observed at vertices 2 and 3. In the model discussed in this paper, vertex 1 transmits x_1 and vertices 2 and 3 observe $h_{1,2}(x_1)$ and $h_{1,3}(x_1)$ respectively, whereas in the model discussed in [2, 3], vertex 1 can be thought of as transmitting $\underline{x}_1 = [x_{1,1}, x_{1,2}]$ and vertices 2 and 3 observe $h_{1,2}(x_{1,2})$ and $h_{1,3}(x_{1,3})$ respectively. We can, of course, also write this as $y_{12} = h_{1,2}(\underline{x}_1)$ and $y_{13} = h_{1,3}(\underline{x}_1)$. Thus, the model discussed in [2, 3] is a special case of the model discussed here.

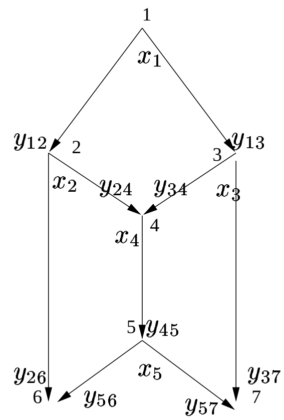

Fig. 1. An example

C. Coding

The multicasting problem has one source vertex (vertex 1), and several destination vertices that we collect in a set $\mathcal{T}$. The source vertex has a message M that is uniformly distributed over $\{1, 2, \ldots, 2^{NR}\}$, where R is the *rate* and where we assume that NR is an integer for simplicity. A communication strategy consists of encoding functions $f_u^{(i)}(\cdot)$, $u \in \mathcal{V}$, $i = 1, 2, \ldots, N$, and decoding functions $\hat{m}_t(\cdot)$, $t \in \mathcal{T}$. We write $\underline{x}_{\mathcal{S}} = [x_u : u \in \mathcal{S}]$ and $\underline{y}_{\mathcal{S},\mathcal{S}'} = [y_{u,v} : u \in \mathcal{S}, v \in \mathcal{S}']$. We similarly write $\underline{y}_{u,\mathcal{S}'} = [y_{u,v} : v \in \mathcal{S}']$.

- *Encoders.* Suppose $M = m$. At time i, vertex 1 transmits $x_1^{(i)} = f_1^{(i)}(m)$ and every other vertex u receives $\underline{y}_{\mathcal{V},u}^{(i)}$. Vertex u transmits $x_u^{(i)} = f_u^{(i)}(\underline{y}_{\mathcal{V},u}^{(1)}, \underline{y}_{\mathcal{V},u}^{(2)}, \ldots, \underline{y}_{\mathcal{V},u}^{(i-1)})$.
- *Decoders.* After time N, each destination vertex t puts out an estimate $\hat{m}_t(\underline{y}_{\mathcal{V},t}^{(1)}, \underline{y}_{\mathcal{V},t}^{(2)}, \ldots, \underline{y}_{\mathcal{V},t}^{(N)})$.

The error probability is

$$P_e = \Pr\left[\bigcup_{t \in T} \left\{\hat{m}_t\left(\underline{Y}_{\mathcal{V},t}^{(1)}, \underline{Y}_{\mathcal{V},t}^{(2)}, \ldots, \underline{Y}_{\mathcal{V},t}^{(N)}\right) \neq M\right\}\right]. \tag{2}$$

The rate R is said to be *achievable* if, for any $\epsilon > 0$, there exist encoders and decoders that make $P_e \leq \epsilon$ for some N. The *multicast capacity* C is the supremum of the achievable rates.

D. Definitions

A set $S \subset V$ is called a *cut* if $1 \in S$ and the set $\overline{S}$ (the complement of S) contains one or more destination vertices, i.e., $\overline{S} \bigcap T \neq \phi$. We denote the set of all cuts $\{S : S \subset V, 1 \in S, \overline{S} \bigcap T \neq \phi\}$ as Λ. The *boundary* of a cut S, $\beta(S)$ is defined as

$$\beta(S) = \{u : (u, v) \in E, u \in S, v \in \overline{S}\}$$

For fixed input distributions $p_{X_1}(\cdot), p_{X_2}(\cdot), \ldots, p_{X_{|V|}}(\cdot)$, we define the *value* of a cut as follows.

$$\text{Value}(S) = \sum_{u \in \beta(S)} H(Y_{u,\overline{S}}) \tag{3}$$

where $Y_{u,\overline{S}} = [Y_{u,v} : v \in \overline{S}]$. The value of a cut indeed depends on the input distributions, but for economy of notation we do not explicitly include these distributions as arguments in (3). Let $X(A)$ denote the vector $[X_u : u \in A]$.

III. Main Result

Suppose the min-cut is positive, i.e., all destinations are connected to the source. Since the network is acyclic, without loss of generality, we can consider only the subgraph having those vertices and edges on the paths from the source to the destinations. We can further number the vertices so that $(u, v) \in \mathcal{E}$ implies that $u < v$. The following theorem is our main result.

Theorem 1: The multicast capacity of an acyclic Aref network is

$$C = \max_{p_{X_1}(\cdot), p_{X_2}(\cdot), \ldots, p_{X_{|V|}}(\cdot)} (\min_{S \in \Lambda}) \text{Value}(S) \tag{4}$$

where $\Lambda = \{S : S \subset V, 1 \in S, \overline{S} \bigcap T \neq \phi\}$.
This result has a max-flow, min-cut interpretation. For specified probability distributions $p_{X_1}(\cdot), p_{X_2}(\cdot), \ldots, p_{X_{|V|}}(\cdot)$ and a $t \in T$, the min-cut is given by $\min_{S \in \Lambda_t} \text{Value}(S)$ where $\Lambda_t = \{S : 1 \in S, t \in \overline{S}\}$. The minimum value of a cut over all vertices $t \in T$ would then be $\min_{S \in \bigcup_{t \in T} \Lambda_t} \text{Value}(S)$. However, note that $\Lambda = \bigcup_{t \in T} \Lambda_t$. Thus, the result says that the maximum achievable data rate is the maximum value of the minimum cut in the information theoretic sense.

A. Achievability

We code in $L + |\mathcal{V}| - 2$ blocks of length n, i.e., we set $N = (L+|\mathcal{V}|-2) \cdot n$. We divide the message m into L parts that each take on values in $\{1, 2, \ldots, 2^{nR}\}$. The l-th part of m is denoted m_l. The overall rate is $R \cdot L / (L + |\mathcal{V}| - 2)$, but since $|\mathcal{V}|$ is finite one can approach R by increasing L.

Let $f_v^n(\cdot) = [f_v^{(i)}(\cdot) : i = 1, 2, \ldots, n]$, and set $f_v^{(i+l\cdot n)}(\cdot) = f_v^{(i)}(\cdot)$ for all i, l, and v, i.e., we use the same encoding function(s) for each block. We associate $\underline{x}_u^n(l)$ and $\underline{y}_{\mathcal{V},u}^n(l)$ with the transmissions for the l-th message m_l. That is, we write the $(l+u-1)$-th transmitted and $(l+u-2)$-th received vectors at vertex u as the respective

$$\underline{x}_u^n(l) = [\underline{x}_u^{(1)}(l), \underline{x}_u^{(2)}(l), \ldots, \underline{x}_u^{(n)}(l)]$$
$$\underline{y}_{\mathcal{V},u}^n(l) = [\underline{y}_{\mathcal{V},u}^{(1)}(l), \underline{y}_{\mathcal{V},u}^{(2)}(l), \ldots, \underline{y}_{\mathcal{V},u}^{(n)}(l)].$$

We sometimes drop the index l if it does not play an important role. We fix the distributions $p_{X_1}(\cdot), p_{X_2}(\cdot), \ldots, p_{X_{|\mathcal{V}|}}(\cdot)$.

Codebooks. At vertex 1, choose $f_1^n(\cdot)$ to map each of the indices in $\{1, 2, \ldots, 2^{nR}\}$ to a sequence $\underline{x}_1^n$ drawn uniformly from $T_\delta(X_1)$. At vertex u, choose $f_u^n(\cdot)$ to map each sequence in $T_\delta(\underline{Y}_{\mathcal{V},u})$ to a sequence drawn uniformly from $T_\delta(X_v)$. Note that some $\underline{y}_{\mathcal{V},u}^n$ in $T_\delta(\underline{Y}_{\mathcal{V},u})$ might never be used. Note also that we have $\underline{y}_{u,v}^n \in T_\delta(Y_{u,v})$ for all (u, v) because $\underline{x}_u^n \in T_\delta(X_v)$ for all u (see Lemma 3).

Encoding. During the $(l + u - 1)$-th block:

- Vertex $u = 1$ transmits $\underline{x}_1^n(l) = f_1^n(m_l)$ for $l = 1, 2, \ldots, L$ and $\underline{x}_1^n(l) = f_1^n(1)$ otherwise.
- Vertex u, $u \neq 1$, observes $\underline{y}_{\mathcal{V},u}^n(l+1)$ and transmits

$$\underline{x}_u(l) = f_u^n\left(\underline{y}_{\mathcal{V},u}(l-1)\right).$$

For instance, the coding strategy for the network shown in Fig. 2 is given in Table III-A. Observe how the transmissions are "pipelined" and that over $L + |V| - 2$ blocks, all vertices have finished transmitting data.

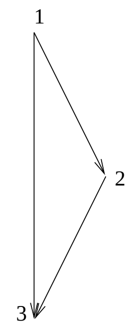

Fig. 2. This figure shown a network with 3 vertices. Vertex 1 is the source and vertex 3 is the destination. The following table shows the transmission and reception at various nodes for $L = 3$ messages.

Decoding. Since $\underline{y}_{\mathcal{V},t}^n(l)$ is a function of m_l, we abuse notation and write this sequence of vectors as $\underline{y}_{\mathcal{V},t}^n(m_l)$. Vertex t decodes m_l after block $t + l - 2$ by using the function $\hat{m}_t(\cdot)$, where we again abuse notation by using the same expression as in (2). We further define

$$\hat{m}_t\left(\underline{y}_{\mathcal{V},t}^n(m_l)\right) = \begin{cases} \text{error} & \text{if } \underline{y}_{\mathcal{V},t}^n(m_l') = \underline{y}_{\mathcal{V},t}^n(m_l) \text{ for some } m_l' \neq m_l \\ m_l & \text{otherwise.} \end{cases} \tag{5}$$

Analysis: In the following, we consider only transmissions that pertain to the message m_l. We thus drop the time indices for convenience. For example, we write m, $\underline{x}_u^n$, and $\underline{y}_{u,v}^n$ for m_l, $\underline{x}_u^n(l)$ and $\underline{y}_{u,v}^n(l)$, respectively.

Consider a destination vertex t. Let $\overline{P}_e(t, m, m')$ be the average probability that vertex t cannot distinguish between m and m', where the average is over the ensemble of encoding functions. Let $\mathcal{S}(m, m')$ be the set of vertices u for which

$$\underline{y}_{\mathcal{V},u}^n(m) \neq \underline{y}_{\mathcal{V},u}^n(m') \tag{6}$$

TABLE I
CODING STRATEGY FOR THE NETWORK OF FIG. 2

Block	Message	1 Transmits	2 Receives	2 Transmits	3 Receives	Decoder output
1	$m(1)$	$\underline{x}_1^n(1)$	$\underline{y}_{1,2}^n(1)$	$\cdot$	$\underline{y}_{1,3}^n(1)$	$\cdot$
2	$m(2)$	$\underline{x}_1^n(2)$	$\underline{y}_{1,2}^n(2)$	$\underline{x}_2^n(1)$	$\underline{y}_{1,3}^n(2), \underline{y}_{2,3}^n(1)$	$\hat{m}_t(\underline{y}_{1,3}^n(1), \underline{y}_{2,3}^n(1))$
3	$m(3)$	$\underline{x}_1^n(3)$	$\underline{y}_{1,2}^n(3)$	$\underline{x}_2^n(2)$	$\underline{y}_{1,3}^n(3), \underline{y}_{2,3}^n(2)$	$\hat{m}_t(\underline{y}_{1,3}^n(2), \underline{y}_{2,3}^n(2))$
4	$\cdot$	$\cdot$	$\cdot$	$\underline{x}_2^n(3)$	$\underline{y}_{2,3}^n(3)$	$\hat{m}_t(\underline{y}_{1,3}^n(3), \underline{y}_{2,3}^n(3))$

i.e., $\mathcal{S}(m,m')$ is the set of the vertices that *can* distinguish between m and m'. We view $\mathcal{S}(m,m')$ as a random variable that is a function of the encoding functions. We clearly have $1 \in \mathcal{S}(m,m')$. Suppose vertex t cannot distinguish between m and m', so that $t \in \overline{\mathcal{S}}(m,m')$ and $\mathcal{S}(m,m')$ is a cut between vertices 1 and t. Let Λ_t be the set of such cuts, i.e., we define $\Lambda_t = \{\mathcal{S} \subset \mathcal{V} : 1 \in \mathcal{S}, t \in \overline{\mathcal{S}}\}$. We can write

$$\begin{aligned}\overline{P}_e(t,m,m') &= \Pr\left[\bigcup_{\mathcal{S}\in\Lambda_t} \{\mathcal{S}(m,m') = \mathcal{S}\}\right] \\ &\le \sum_{\mathcal{S}\in\Lambda_t} \Pr\left[\mathcal{S}(m,m') = \mathcal{S}\right]. \end{aligned} \tag{7}$$

Furthermore, we claim that a necessary condition for the event $\mathcal{S}(m,m') = \mathcal{S}$ is one must have

$$\underline{y}_{u,\overline{\mathcal{S}}}^n(m) = \underline{y}_{u,\overline{\mathcal{S}}}^n(m') \tag{8}$$

for *all* $u \in \mathcal{S}(m,m')$. To see this, note that if (8) was not true for some $u \in \mathcal{S}(m,m')$, then there is a vertex $v \in \overline{\mathcal{S}}$ that can distinguish between m and m', contradicting our original hypothesis. We can thus write

$$\begin{aligned}\Pr\left[\mathcal{S}(m,m') = \mathcal{S}\right] &\le \Pr\left[\bigcap_{u\in\beta(\mathcal{S})} \left\{\underline{y}_{u,\overline{\mathcal{S}}}^n(m) = \underline{y}_{u,\overline{\mathcal{S}}}^n(m')\right\}\right] \\ &= \prod_{u\in\beta(\mathcal{S})} \Pr\left[\underline{y}_{u,\overline{\mathcal{S}}}^n(m) = \underline{y}_{u,\overline{\mathcal{S}}}^n(m')\right] \end{aligned} \tag{9}$$

where the equality follows because the $\underline{x}_u^n$, $u \in \mathcal{V}$, were chosen independently.

We proceed to upper bound the probabilities in the product in (9). We have $(\underline{x}_u^n(m'), \underline{y}_{u,\overline{\mathcal{S}}}^n(m')) \in T_\delta(X_u, Y_{u,\mathcal{S}})$ by Lemma 3. The event (8) thus implies

$$\left(\underline{x}_u^n(m'), \underline{y}_{u,\overline{\mathcal{S}}}^n(m)\right) \in T_\delta(X_u, Y_{u,\overline{\mathcal{S}}}). \tag{10}$$

But note that $\underline{x}_u^n(m')$ was chosen independent of $\underline{y}_{u,\overline{\mathcal{S}}}^n(m)$. The probability of (10) occurring is thus

$$\left|T_\delta(X_u|\, \underline{y}_{u,\overline{\mathcal{S}}}^n(m))\right| \Big/ \left|T_\delta(X_u)\right|. \tag{11}$$

We use Lemmas 1 and 2 to bound

$$|T_\delta(X_u)| \ge (1-\epsilon_\delta(n)) \cdot 2^{n(1-\delta)H(X_u)} \tag{12}$$

$$|T_\delta(X_u|\underline{y}_{u,\overline{\mathcal{S}}}^n(m))| \le 2^{n(1+\delta_2)H\left(X_u|Y_{u,\overline{\mathcal{S}}}\right)} \tag{13}$$

where $\epsilon_\delta(n) \to 0$ as $n \to \infty$. Inserting (12) and (13) into (11), we have

$$\begin{aligned}&\Pr\left[\underline{y}_{u,\overline{\mathcal{S}}}^n(m) = \underline{y}_{u,\overline{\mathcal{S}}}^n(m')\right] \\ &\quad \le (1-\epsilon_\delta(n))^{-1} \cdot 2^{n(\delta+\delta_2)} \cdot 2^{-nH\left(Y_{u,\overline{\mathcal{S}}}\right)} \end{aligned} \tag{14}$$

where we have used $H(Y_{u,\overline{\mathcal{S}}}|X_u) = 0$. Inserting (14) into (9), and using $|\beta(\mathcal{S})| \le |\mathcal{E}|$, we have

$$\begin{aligned}&\Pr\left[\mathcal{S}(m,m') = \mathcal{S}\right] \\ &\quad \le (1-\epsilon_\delta(n))^{-|\mathcal{E}|} \cdot 2^{n|\mathcal{E}|(\delta+\delta_2)} \cdot 2^{-n\mathrm{Value}(\mathcal{S})}. \end{aligned} \tag{15}$$

Inserting (15) into (7), and using the fact that the number of cuts is less than $2^{|\mathcal{V}|}$, we have

$$\begin{aligned}&\overline{P}_e(t,m,m') \\ &\le (1-\epsilon_\delta(n))^{-|\mathcal{E}|} \cdot 2^{|\mathcal{V}|+n|\mathcal{E}|(\delta+\delta_2)} \cdot 2^{-n\min_{\mathcal{S}\in\Lambda_t}\mathrm{Value}(\mathcal{S})}. \end{aligned} \tag{16}$$

The above applies to the l-th block of transmission. We now add the index l to m_l. Let $\overline{P}_e(m)$ be the average probability of error when the (overall) message m was transmitted. We use the union bound over all L blocks, all destinations t, and all $m' \neq m$ to write

$$\begin{aligned}\overline{P}_e(m) &\le \sum_{l=1}^{L}\sum_{t\in\mathcal{T}}\sum_{m_l'\neq m_l} \overline{P}_e(t,m_l,m_l') \\ &\le L\cdot|\mathcal{T}|\cdot(2^{nR}-1)\cdot(1-\epsilon_\delta(n))^{-|\mathcal{E}|} \\ &\quad \cdot 2^{|\mathcal{V}|+n|\mathcal{E}|(\delta+\delta_2)} \cdot 2^{-n\min_{\mathcal{S}\in\Lambda_t}\mathrm{Value}(\mathcal{S})}. \end{aligned} \tag{17}$$

We thus find that the average error probability for any message can be made small if n is large and

$$R < -|\mathcal{E}|(\delta+\delta_2) + \min_{\mathcal{S}\in\Lambda_t} \mathrm{Value}(\mathcal{S}). \tag{18}$$

Finally, we optimize over all input distributions, choose δ and δ_2 small, and choose n and L large. The result is that we can make the overall rate approach C in (4) while at the same time ensuring that $P_e \le \epsilon$ for any positive ϵ.

B. Converse

In this section, we show a converse for the result in Theorem 1. (We begin by mentioning the following result which is a multicast version of a similar result proved in Section 14.10 of [6]. We omit the proof.)

$$C \leq \max_{p_{X_1}(\cdot), p_{X_2}(\cdot), \ldots, p_{X_{|V|}}(\cdot)} (\min_{S \in \Lambda} I(X(S); Y(\overline{S}) | X(\overline{S})))$$

Consider the term $I(X(S); Y(\overline{S}) | X(\overline{S}))$. We have,

$$\begin{aligned} I(X(S); Y(\overline{S}) | X(\overline{S})) &= H(Y(\overline{S}) | X(\overline{S})) \qquad (19) \\ &\quad - H(Y(\overline{S}) | X(\overline{S}), X(S)) \\ &= H(Y(\overline{S}) | X(\overline{S})) \qquad (20) \\ &= H(Y_B(\overline{S}) | X(\overline{S})) \\ &\quad + H(Y_I(\overline{S}) | X(\overline{S}), Y_B(\overline{S})) \qquad (21) \\ &= H(Y_B(\overline{S}) | X(\overline{S})) \qquad (22) \\ &= H(Y_B(\overline{S})) \qquad (23) \\ &= \sum_{u \in \beta(S)} H(Y_{u,\overline{S}}) = \text{Value}(S) \qquad (24) \end{aligned}$$

where

- (20) follows from the fact that $Y(\overline{S})$ is specified completely by $X(\overline{S})$ and $X(S)$.
- (21) follows by defining $Y_B(\overline{S}) = [Y_{u,v} : u \in S, v \in \overline{S}]$ and $Y_I(\overline{S}) = [Y_{u,v} : u \in \overline{S}, v \in \overline{S}, (u,v) \in E]$.
- (22) follows from the fact that $Y_I(\overline{S})$ is specified completely by $X(\overline{S})$.
- (23) follows from the fact that $X(\overline{S})$ and $X(S)$ are independent and $X(S)$ specifies $Y_B(\overline{S})$.
- (24) follows from the fact that $Y_{u,\overline{S}}$ are independent of each other for each $u \in \beta(S)$.

Thus, we now have the result that

$$C \leq \max_{p_{X_1}(\cdot), p_{X_2}(\cdot), \ldots, p_{X_{|V|}}(\cdot)} (\min_{S \in \Lambda} \text{Value}(S)) \qquad (25)$$

However, note that the maximum achievable rate derived in Section III-A is

$$R = \max_{p_{X_1}(\cdot), p_{X_2}(\cdot), \ldots, p_{X_{|V|}}(\cdot)} (\min_{S \in \Lambda} \text{Value}(S))$$

Note that the converse is tight provided that the maximization in (25) is achieved at a product distribution. In the rest of the section, we prove that this indeed is the case.

Lemma 4: For Aref networks, the bound (25) is optimized by independent inputs X_u, $u \in \mathcal{V}$.

Proof: For any fixed $p_{X_1 X_2 \cdots X_{|\mathcal{V}|}}(\cdot)$, we have

$$\begin{aligned} I(\underline{X}_{\mathcal{S}} \,;\, \underline{Y}_{\mathcal{V},\overline{\mathcal{S}}} \mid \underline{X}_{\overline{\mathcal{S}}}) &= H(\underline{Y}_{\mathcal{V},\overline{\mathcal{S}}} \mid \underline{X}_{\overline{\mathcal{S}}}) \\ &\leq H(\underline{Y}_{\mathcal{V},\overline{\mathcal{S}}}) \\ &\leq \sum_{u \in \beta(\mathcal{S})} H(\underline{Y}_{u,\overline{\mathcal{S}}}) \qquad (26) \end{aligned}$$

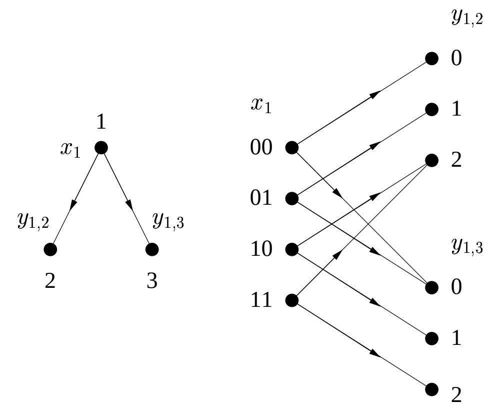

Fig. 3. An Aref network for which $C_{1,T} < \min_{t \in T} C_{1,t}$.

where the first step follows because the network is deterministic, and the other steps because conditioning cannot increase entropy. Furthermore, by replacing the joint distribution by the product of its marginals, the mutual information in (25) is exactly the sum of the entropies in (26). That is, one can restrict attention to independent inputs. ■

Since replacing any joint distribution by the product of its marginals cannot decrease the value of a cut, it follows that the maximization is achieved at a product distribution. Thus it follows that, for Aref networks, it suffices to perform the maximization over all the product distributions, which proves the converse.

C. Discussion

For the usual deterministic networks without broadcasting, one can show that the multicasting capacity is

$$C = \min_{t \in T} C_{1,t}$$

where $C_{1,t}$ is the *unicast* capacity from vertex 1 to vertex t. However, for Aref networks such a relationship is not necessarily true. Consider the network shown on the left in Fig. 3. Suppose the channel from x_1 to $y_{1,2}$ and $y_{1,3}$ is the broadcast channel shown on the right in Fig. 3. It is easy to see that $C_{1,2} = C_{1,3} = \log_2(3)$ bits by choosing 3 out of 4 of the input letters to have probability 1/3. The multicast capacity is, however, only $C = 1.5$, and is achieved only if all 4 inputs have probability 1/4.

IV. Acknowledgements

The authors would wish to thank Tracey Ho for valuable suggestions.

References

[1] Mohammad Reza Aref, *Information flow in relay networks*, PhD thesis, Stanford Univeristy, 1981.

[2] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow", *IEEE Transactions on Information Theory*, vol. IT-46, pp. 1204–1216, 2000.

[3] R. Koetter and M. Medard, "An algebraic approach to network coding", *IEEE/ACM transactions on networking*, vol. 11, no. 5, October 2003.

[4] Radhika Gowaikar, Amir F. Dana, Ravi Palanki, Babak Hassibi, and Michelle Effros, "On the capacity of wireless erasure networks", *Proc. ISIT,Chicago*, p. 401, 2004.

[5] Alon Orlitsky and James R. Roche, "Coding for computing", in *IEEE Symposium on Foundations of Computer Science*, 1995, pp. 502–511.
[6] Thomas M Cover and Joy A Thomas, *Elements of Information Theory*, John Wiley & Sons, Inc., New York, New York, 1991.