Edge-Cut Bounds on Network Coding Rates

Serap A. Savari
Electrical Engineering and Computer Science Dept.
University of Michigan
Ann Arbor, MI 48109, USA
Email: savari@eecs.umich.edu

Gerhard Kramer
Bell Labs, Lucent Technologies
600 Mountain Ave.
Murray Hill, NJ 07928, USA
Email: gkr@bell-labs.com

***Abstract*— Two bounds on network coding rates are reviewed that generalize edge-cut bounds on routing rates. The simpler bound is a bidirected cut-set bound which generalizes and improves upon a flow cut-set bound that is standard in networking. It follows that routing is rate-optimal if routing achieves the standard flow cut-set bound. The second bound improves on the cut-set bound, and it involves progressively removing edges from a network graph and checking whether certain strengthened d-separation conditions are satisfied.**

I. Introduction

Fifty years ago, several individuals investigated the problem of determining the maximal *flow* from one vertex to another in a graph subject to capacity limitations on arcs or edges [4], [5], [6], [16]. The outcome was the celebrated "max-flow min-cut" theorem that states that the maximal flow is the minimum capacity among all edge cuts separating the source and destination vertices (a related bound additionally partitions the vertex set into two disjoint sets). The result is generally attributed to L. R. Ford and D. R. Fulkerson, who wrote in [7, p. 1] that the problem "comes up naturally in the study of transportation or communication networks." Their work was one of the cornerstones of network optimization theory, which is concerned with finding the most effective ways to send flow over a network, and the max-flow/min-cut theorem was subsequently extended to the multicommodity flow problem [17, p. 1221].

The max-flow/min-cut theorem from network optimization theory can be immediately applied to communication problems where messages are routed, but not to communication problems permitting the generality of network coding. Network coding has been intensely studied since [1] presented a novel coding scheme that attains a cut-set bound for multicasting in networks. We here study the problem of upper bounding network coding rates for the general case of multi-message multicast and describe results that first appeared in [12] and [13]. The standard vertex-partitioning cut-set bound of [3, Sec. 14.10] provides one type of bound. We shall discuss a strengthened version of this bound for undirected networks that is taken from [12].

We will also discuss a more recent improvement of these two bounds that comes from [13]. As pointed out in [7, pp. 16-17], sometimes tighter bounds can be found for routing by considering "disconnecting edge sets". However, this approach cannot be applied to network coding in the obvious way. We address this problem by developing an information-theoretic counterpart to edge-cut bounds that does apply to network coding. We do this by borrowing from the artificial intelligence literature [14] the concept of d-*separation* in Bayesian networks.

Bayesian networks are graphs whose vertices represent random variables, and d-separation is a graphical procedure that establishes the conditional statistical independence of certain sets of these random variables. We consider special types of Bayesian networks known as *functional dependence graphs* (FDGs), and use (a strengthened version of) an extension of d-separation called fd-*separation* that appeared in [10, Ch. 2].

II. A Bidirected Cut-set Bound

Consider an undirected, edge-capacitated graph $\mathcal{N} = (\mathcal{V}, \mathcal{E})$ with vertex and edge sets

$$\mathcal{V} = \{1, 2, \ldots, V\} \quad (1)$$

$$\mathcal{E} = \{(u_1, v_1), (u_2, v_2), \ldots, (u_E, v_E)\} \quad (2)$$

respectively, where $u_e, v_e \in \mathcal{V}$ for $e = 1, 2, \ldots, E$, and where C_e is the capacity of edge e. Consider further a subset $\mathcal{T} = \{t_1, t_2, \ldots, t_T\}$ of $\mathcal{V}$ called *terminals*, some of which are *sources* and some of which are *sinks*. Suppose edge (u, v) represents a two-way channel (TWC)

$$P\left(y_{uv}, y_{vu} | x_{uv}, x_{vu}\right) \quad (3)$$

where x_{uv} and y_{uv} are the respective input to and output from edge (u, v) at vertex u. Each TWC is characterized by a two-dimensional *capacity region* $\mathcal{C}_{uv}$, i.e., a set of rate pairs (R_{uv}, R_{vu}) that specifies at what rates one can transmit in both directions simultaneously and reliably, where R_{uv} is the rate going from u to v (see, Fig. 1 for illustrations of the capacity region of a directed edge, an undirected edge, and a different type of edge). The terminals can further perform *network coding* [1], [9], i.e., each terminal can transmit into each of its TWCs any function of its messages and past received outputs.

The standard network cut-set bound gives a useful outer bound on the set of feasible flow rates. However, this bound is based on flow considerations, and hence may not apply to network coding. Instead, in [12] we employ an information-theoretic cut-set bound that takes coding into account, and we summarize our results here.

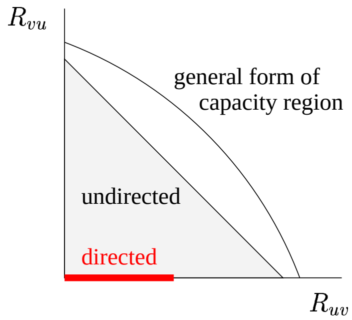

Fig. 1. Capacity regions of different types of edges.

For networks of bidirectional TWCs in which Shannon's outer bound [15] for each constituent channel, i.e., edge, is its capacity region, we can reduce the bound of [3, Sec. 14.10] to the following three steps [11].

1) Convert every network edge into a pair of directed edges whose capacity pair is a boundary point of the capacity region of this edge.
2) Apply the flow cut set bound to get a rate region $\mathcal{R}_{cut}$.
3) Repeat the above two steps for all capacity boundary points on all edges. The union of the regions $\mathcal{R}_{cut}$ is a an outer bound on the set of achievable rates.

Note that for directed networks, the above bound is the same as the flow cut set bound. However, the flow cut-set bound and the bidirected cut-set bound can differ when there are two-way edges, e.g., undirected edges having triangular capacity regions. We call this bound a *bidirected* cut-set bound.

We have the following consequences of our cut-set bound.

1) Let $\mathcal{R}_{flow}$ be the feasible routing (flow) rates, $\mathcal{R}_{NC}$ the achievable network coding rates, $\mathcal{R}_{cut}^{2D}$ the bidirected cut set rates, and $\mathcal{R}_{cut}$ the standard flow cut set rates for directed or undirected graphs. We have the ordering

$$\mathcal{R}_{flow} \subseteq \mathcal{R}_{NC} \subseteq \mathcal{R}_{cut}^{2D} \subseteq \mathcal{R}_{cut}. \tag{4}$$

This ordering implies that a network flow problem for which $\mathcal{R}_{flow} = \mathcal{R}_{cut}$ also has $\mathcal{R}_{flow} = \mathcal{R}_{NC}$. Routing is therefore optimal in terms of rates. A survey of cases where $\mathcal{R}_{flow} = \mathcal{R}_{cut}$ can be found in [17, Part VII].

2) The bound converts a network of two-way channels into a set of *directed* networks. Thus, any problem for which one achieves the flow cut-set bound for directed networks, one also achieves the bidirected cut-set bound for networks of TWCs. For example, it is known that linear network coding is optimal for multicasting a single source in directed networks [1], [9]. Linear network coding is therefore also optimal for multicasting a single source in networks of TWCs. Furthermore, for such problems one can *separate* channel and network coding.

III. Review of the PdE Bound for Network Coding

The bidirected cut-set bound is often loose. However, for directed networks one can sometimes determine improved outer bounds based on edge cuts. An *edge cut* is a set $\mathcal{E}_d$ of edges that disconnects sources from sinks. (Edge cuts in directed graphs are sometimes called *directed cuts* or *disconnecting edge sets*.) Rather intuitively, the sum of the routing rates of the source-destination pairs that are disconnected by $\mathcal{E}_d$ is upper bounded by the sum of the capacities of the edges in $\mathcal{E}_d$.

We would like to apply edge-cut bounds to network coding. Such bounds clearly apply to *undirected* graphs, and one can prove this by using the techniques of [12]. Unfortunately, edge-cut bounds do not necessarily apply to *directed* graphs [13]. In [13], we developed an alternative to edge-cut bounds that does apply to network coding and further used the bound to derive new capacity theorems for network information flow.

A. Network Model

We adopted in [13] the model of [2], [12] where the network is clocked, i.e, there is a universal clock that ticks N times. Vertex u transmits symbols $X_{uv}^{(n)}$, $(u,v) \in \mathcal{E}$, *after* clock tick $n-1$ and *before* clock tick n for $n = 1, 2, \ldots, N$. Vertex u receives symbols $Y_{vu}^{(n)}$, $(v,u) \in \mathcal{E}$, *at* clock tick n. We refer to [12] for more details on the remaining assumptions. For simplicity of exposition, we will here model the edge channels as being noise-free, i.e., we will mostly consider channels with $Y_{vu}^{(n)} = X_{vu}^{(n)}$ for all u and v. However, our results do extend (see [13]) to noisy channels with $Y_{uv}^{(n)} = f_{uv}(X_{uv}^{(n)}, Z_{uv}^{(n)})$, where the $Z_{uv}^{(n)}$, $(u,v) \in \mathcal{E}$, are noise random variables that are statistically independent of each other and the $X_{uv}^{(n)}$.

A widely studied networking problem is the *multicommodity flow* problem [17, p. 1221]. This problem has a set of K commodities that must be routed through $\mathcal{N}$. In a communication network, the commodities are *messages*, and we denote these by $W_1, W_2, \ldots, W_K$. Message W_k is associated with a vertex pair (s_k, t_k), $s_k \neq t_k$, and one wishes to transmit R_k units of data from s_k to t_k simultaneously for all k. The meaning is that s_k is the *source* vertex and t_k is the *sink* or destination vertex. In communications, R_k refers to the *rate* of source k.

A more general problem is the multi-message multicasting problem, where several destinations decode each message W_k. More precisely, message W_k is associated with the vertices $(s_k, t_k(1), t_k(2), \ldots, t_k(D_k))$, and one wishes to transmit R_k units of data from s_k to $t_k(i)$, $s_k \neq t_k(i)$, simultaneously for all $k = 1, 2, \ldots, K$ and $i = 1, 2, \ldots, D_k$. We denote the estimate of W_k at destination $t_k(i)$ by $\hat{W}_k^{(i)}$.

B. Functional Dependence Graphs and d-Separation

To arrive at the results of [13], we used the calculus of d-separation and fd-separation in FDGs. FDGs are graphs where the vertices represent random variables, and the edges represent the functional dependencies between the random variables [10], [11]. For instance, suppose we have N_{RV} random variables that are defined by S_{RV} independent (or source) random variables by N_{RV} functions. An FDG $\mathcal{G}$ is a directed graph having $N_{RV} + S_{RV}$ vertices representing the random variables, and in which edges are drawn from one vertex to another if the random variable of the former vertex is an argument of the function defining the random variable of the latter vertex.

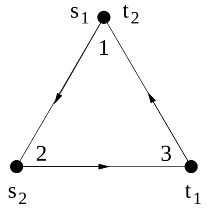

Fig. 2. A two-commodity fbw problem on a directed graph.

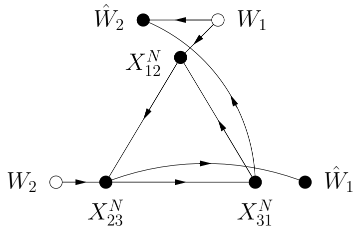

Fig. 3. FDG for the two-commodity problem in Fig. 2.

For example, suppose we have the 2-commodity problem in a noise-free triangular network depicted in Fig. 2. A corresponding FDG is shown in Fig. 3. In this graph, X_{12}^N is a function of the message W_1 and X_{31}^N (in fact, $X_{12}^{(n)}$ is a function of W_1 and the *past* X_{31}^{n-1} only). The message estimate $\hat{W}_2$ of W_2 at vertex 1 is also a function of W_1 and X_{31}^N. The channel inputs X_{23}^N are a function of W_2 and X_{12}^N, and the estimate $\hat{W}_1$ is a function of X_{23}^N. The $S_{RV} = 2$ vertices representing the independent W_1 and W_2 are distinguished by drawing them with a hollow circle. Note that Fig. 3 is the line graph of Fig. 2 with the addition of vertices representing the messages and their estimates, and edges representing the functional relations of these new vertices to the existing ones.

By d-separation we mean the following reformulation of a definition in [14, p. 117] that is described in [10], [11].

Definition 1: Let $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{Z}$ be disjoint subsets of the vertices of a FDG $\mathcal{G}$. $\mathcal{Z}$ is said to *d-separate* $\mathcal{X}$ from $\mathcal{Y}$ if there is no path between a vertex in $\mathcal{X}$ and a vertex in $\mathcal{Y}$ after the following manipulations of the graph have been performed.

1) Consider the subgraph $\mathcal{G}_{\mathcal{X}\mathcal{Y}\mathcal{Z}}$ of $\mathcal{G}$ consisting of the vertices in $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{Z}$, as well as the edges and vertices encountered when moving *backward* one or more edges starting from any of the vertices in $\mathcal{X}$ or $\mathcal{Y}$ or $\mathcal{Z}$.
2) In $\mathcal{G}_{\mathcal{X}\mathcal{Y}\mathcal{Z}}$ delete all edges coming *out* of the vertices in $\mathcal{Z}$. Call the resulting graph $\mathcal{G}_{\mathcal{X}\mathcal{Y}|\mathcal{Z}}$.
3) Remove the arrows on the remaining edges of $\mathcal{G}_{\mathcal{X}\mathcal{Y}|\mathcal{Z}}$ to obtain an undirected graph.

C. The PdE Bound

The bound we developed in [13] begins with a set of edges $\mathcal{E}_d$ like the edge-cut bound. However, in addition to computing the sum of the capacities of these edges, we must perform a series of verification steps. Consider a set $\mathcal{S}_d$ of source indices, and an ordering of these indices via a one-to-one mapping $\pi(\cdot)$ from $\{1, 2, \ldots, |\mathcal{S}_d|\}$ to $\mathcal{S}_d$, where $|\mathcal{S}_d|$ is the cardinality of $\mathcal{S}_d$. We use the notation $X_{\mathcal{E}_d} = \{X_{uv} : (u, v) \in \mathcal{E}_d\}$, and similarly for $Y_{\mathcal{E}_d}$ and $Z_{\mathcal{E}_d}$. The following steps describe our bound. Let $X_{\mathcal{E}_d}^N$ be the channel inputs (and outputs) of the edges $\mathcal{E}_d$, $W_{\mathcal{S}_d}$ be the messages with indices in $\mathcal{S}_d$, $\mathcal{S}_d \subseteq \{1, 2, \ldots, K\}$, and $\mathcal{S}_d^C$ be the complement of $\mathcal{S}_d$ in $\{1, 2, \ldots, K\}$.

1) (Initialization) Consider the FDG $\mathcal{G}$ corresponding to the network graph $\mathcal{N}$, i.e., the line graph of $\mathcal{N}$ with the addition of vertices and edges representing the messages and their estimates (see, e.g., Fig. 2 and Fig. 3).
   - Remove all vertices and edges in $\mathcal{G}$ except those encountered when moving backward one or more edges starting from any of the vertices representing: (1) $X_{\mathcal{E}_d}^N$, (2) any choice of non-empty subset of $\{\hat{W}_k^{(i)} : i = 1, 2, \ldots, D_k\}$ for all $k \in \mathcal{S}_d$, and (3) all messages W_k, $k = 1, 2, \ldots, K$.
   - Further remove the edges coming out of the vertices representing $X_{\mathcal{E}_d}^N$ and $W_{\mathcal{S}_d^C}$, and successively remove edges coming out of vertices and on cycles that have no incoming edges, excepting source vertices. Call the resulting graph $\mathcal{G}_{\mathcal{E}_d}$. Set $k = 1$.
2) (Iterations) If $W_{\pi(k)}$ is not disconnected (in an undirected sense) from one of its estimates $\hat{W}_{\pi(k)}^{(i)}$, $i = 1, 2, \ldots, D_k$, then stop (one has no bound). If $W_{\pi(k)}$ is disconnected (in an undirected sense) from all of its estimates then:
   - Remove the edges coming out of the vertex representing $W_{\pi(k)}$.
   - Successively remove edges coming out of vertices and on cycles that have no incoming edges, excepting source vertices. Call the resulting graph $\mathcal{G}_{\mathcal{E}_d W_\pi^k}$.
3) (Termination and Bound) Increment k. If $k \leq K$ go to the previous step. If $k = K + 1$, then we have

$$\sum_{k \in \mathcal{S}_d} R_k \leq \sum_{e \in \mathcal{E}_d} C_e. \tag{5}$$

In [13] we dubbed this bound a *progressive d-separating edge-set* bound (or PdE bound for short) and established the validity of the above procedure. The word "progressive" describes the step-by-step removal of edges from $\mathcal{G}$, and the term "d-separation" describes the use of fd-separation in steps 1 and 2 above. We remark that the PdE bound includes as special cases the previous bounds based on edge cuts that partition $\mathcal{V}$ into two disjoint sets [3, Sec. 14.10], [12].

Example 1: Consider the network of Fig. 2 for which $\mathcal{G}$ is the graph in Fig. 3. Suppose that $C_e = 1$ for all e. We choose $\mathcal{E}_d = \{(2, 3)\}$, $\mathcal{S}_d = \{1, 2\}$, and the resulting graph $\mathcal{G}_{\mathcal{E}_d}$ is shown in Fig. 4. We next choose $\pi(\cdot)$ to be the identity mapping. For $k = 1$, we must check if W_1 is disconnected from $\hat{W}_1$ which is indeed the case. The next graph $\mathcal{G}_{\mathcal{E}_d W_\pi^1}$ has only one edge, and W_2 is disconnected from $\hat{W}_2$. We thus have the desired bound $R_1 + R_2 \leq 1$ (this type of edge-cut bound first appeared in [12]).

D. Undirected Graphs

The above procedure extends to undirected graphs with a few extra steps. The main addition is that one replaces every

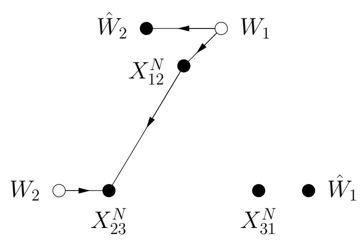

Fig. 4. Modified FDG for the two-commodity problem in Fig. 2.

undirected edge $e = (u, v)$ with capacity C_e by a pair of oppositely directed edges labeled by the entropies $C_{uv} := H(X_{uv}^N)/N$ and $C_{vu} := H(X_{vu}^N)/N$. One then requires that $C_{uv} + C_{vu} \leq C_e$. We remark that it is often more convenient to draw only the bidirected version of the undirected graph without formally converting it into a line graph.

Example 2: Consider the network of Fig. 5 that appeared in a paper by Hu [8]. This network served as an example to show that the vertex-partitioning cut-set bound can be loose for three commodities. We construct the bidirected graph shown in Fig. 6, where the edge from vertex u to vertex v represents X_{uv}^N (we have labeled only some of the edges to avoid cluttering the figure with notation). One can construct the FDG line graph directly from this graph.

Suppose that the undirected edges have capacity two. Hu showed that the vertex-partitioning cut-set bound permits the rate triple $(R_1, R_2, R_3) = (4, 2, 1)$ but routing requires $R_3 = 0$ when $(R_1, R_2) = (4, 2)$. We wish to determine if the same is true with network coding.

We choose $\mathcal{E}_d = \{(3, 6), (5, 6)\}$ and $\mathcal{S}_d = \{1\}$ from which we obtain $R_1 \leq C_{36} + C_{56} \leq 4$, with equality only if $C_{63} = C_{65} = 0$. Similarly, with $\mathcal{E}_d = \{(1, 2), (1, 4)\}$ and $\mathcal{S}_d = \{1\}$ we require $C_{21} = C_{41} = 0$ for $R_1 = 4$. Combining these results, we can restrict attention to the graph in Fig. 7.

For Fig. 7, we choose $\mathcal{E}_d = \{(2, 3), (4, 3), (2, 5), (4, 5)\}$, $\mathcal{S}_d = \{1, 2, 3\}$, $[\pi(1), \pi(2), \pi(3)] = [3, 1, 2]$ and the resulting graph $\mathcal{G}_{\mathcal{E}_d}$ is shown in Fig. 8. We find that

$$R_1 + R_2 + R_3 \leq C_{23} + C_{43} + C_{25} + C_{45}. \quad (6)$$

Next, in Fig. 7 we choose $\mathcal{E}_d = \{(3, 2), (3, 4), (5, 2), (5, 4)\}$, $\mathcal{S}_d = \{2, 3\}$, $[\pi(1), \pi(2)] = [2, 3]$. We find that

$$R_2 + R_3 \leq C_{32} + C_{34} + C_{52} + C_{54}. \quad (7)$$

Combining (6) and (7), for $R_1 = 4$ we have

$$R_1 + 2(R_2 + R_3) \leq 8. \quad (8)$$

Thus, if $R_1 = 4$ and $R_2 = 2$ we require $R_3 = 0$ with or without network coding.

ACKNOWLEDGMENT

Discussions with C. Chekuri are gratefully acknowledged. The work of S. A. Savari was partially supported by NSF Grant No. CCF-0430201. 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.

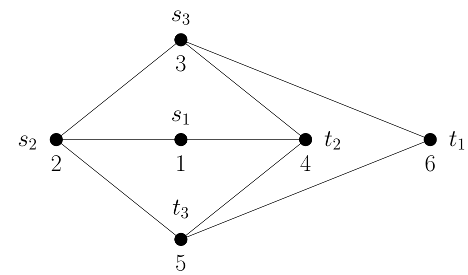

Fig. 5. Hu's three-commodity problem.

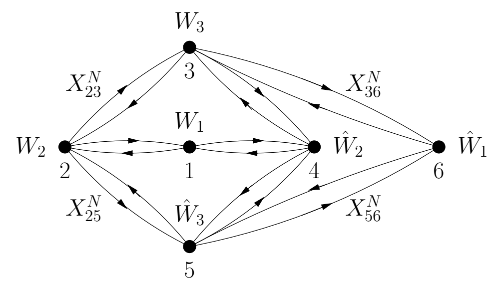

Fig. 6. Bidirected graph for the three-commodity problem in Fig. 5.

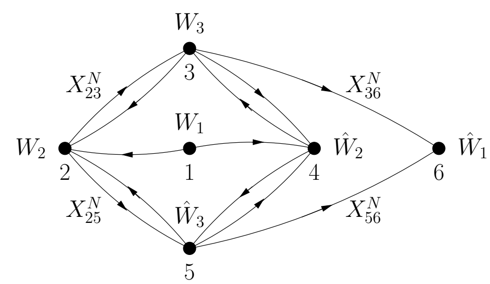

Fig. 7. Modified graph for the three-commodity problem in Fig. 5.

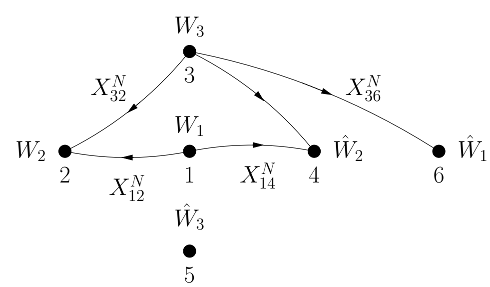

Fig. 8. Modified graph for the three-commodity problem in Fig. 5.

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] S. P. Borade, "Network information flow: limits and achievability," *Proc. IEEE Int. Symp. Inform. Theory*, Lausanne, Switzlerland, p. 139, June 30–July 5, 2002.

[3] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, New York: Wiley, 1991.

[4] G. B. Dantzig and D. R. Fulkerson, "On the max-flow, min-cut theorem of networks," in "Linear Inequalities," *Ann. Math. Studies*, no. 38, pp. 215-221, Princeton, New Jersey, 1956.

[5] P. Elias, A. Feinstein, and C. E. Shannon, "A note on the maximum flow through a network," *IRE Trans. Inform. Theory*, vol. 2, pp. 117–119, Dec. 1956.
[6] L. R. Ford and D. R. Fulkerson, "Maximal flow through a network," *Can. J. Math.*, vol. 8, pp. 399-404, 1956.
[7] L. R. Ford and D. R. Fulkerson, *Flows in Networks*, Princeton: Princeton University Press, 1962.
[8] T. C. Hu, "Multi-commodity network flows," *Oper. Res.*, vol. 11, pp. 344-360, 1963.
[9] R. Koetter and M. Médard, "An algebraic approach to network coding," *IEEE/ACM Trans. Networking*, 2003.
[10] G. Kramer, *Directed Information for Channels with Feedback*, Hartung-Gorre Verlag, Konstanz, 1998, ETH Series in Information Processing, Vol. 11.
[11] G. Kramer, "Capacity results for the discrete memoryless network," *IEEE Trans. Inform. Theory*, vol. 49, no. 1 , pp. 4–21, Jan. 2003.
[12] G. Kramer and S. A. Savari, "On networks of two-way channels," to appear in the *Proc. DIMACS Workshop on Algebraic Coding Theory and Information Theory*, DIMACS Center, Rutgers University, Dec. 15-18, 2003.
[13] G. Kramer and S. A. Savari, "Edge-cut bounds on network coding rates," to appear in *J. Network and Systems Management*, March 2006.
[14] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann, San Mateo, Calif., 1988.
[15] C.E. Shannon, "Two-way communication channels," in *Proc. 4th Berkeley Symp. on Mathematical Statistics and Probability*, J. Neyman, Ed., Berkeley, CA, 1961, vol. 1, pp. 611–644, Univ. Calif. Press, Reprinted in *Claude Elwood Shannon: Collected Papers*, pp. 351-384, (N.J.A. Sloane and A.D. Wyner, eds.) Piscataway: IEEE Press, 1993.
[16] J. T. Robacker, "On network theory," The RAND Corporation, Research Memorandum RM-1498, May 26, 1955.
[17] A. Schrijver, *Combinatorial Optimization*, New York: Springer, 2003.