Throughput of Wireless Relay Networks with Interference Processing

M. Bama, Srikrishna Bhashyam and Andrew Thangaraj,
Department of Electrical Engineering,
Indian Institute of Technology, Madras, Chennai, India 600036.
bama, skrishna, andrew @ iitm.ac.in

***Abstract*— Consider a wireless relay network, where all nodes except the source and destination act as relays. The problem of evaluating the capacity for a source-destination pair in this network and determining the corresponding optimal transmission strategy have attracted considerable research attention recently. However, even for a single relay, complete solutions are still unknown. A popular method to obtain partial results is to fix the relay strategy. In this paper, we consider a situation where relays adopt the decode-and-forward approach with the possibility of network coding. Also, the nodes can receive information from multiple transmitters simultaneously. This is accomplished by allowing physical layer interference processing at nodes. This processing yields increased sum rate when compared to interference avoidance or treating interference as noise. A linear optimization model is proposed to determine the maximum achievable throughput with interference processing. Numerical evaluation done on some example networks shows significant throughput gains.**

I. Introduction

Consider a multihop wireless network represented by a graph $G = (V, E)$. In this paper, we wish to determine the rate at which a single source node $S \in V$ can communicate to a sink node $D \in V$. Other nodes in $V \setminus \{S, D\}$ act as relays in forwarding the message from S to D.

The capacity of a simple three-node relay network is still an open problem. Several relaying strategies were proposed to simplify this problem. These include strategies such as amplify- and-forward relays or compress-and-forward relays. Most of the strategies use Multiple Input Multiple Output (MIMO) system cooperation and the results are asymptotic in Signal-to-Noise-Ratio (SNR) [1], [2], [3].

In this work, we consider *half-duplex* wireless nodes with the relays performing decode and forward communication along with network coding. However, there is no cooperation assumed either at the transmitter nodes or at the receiver nodes. Further, we operate on the finite SNR regime.

In our network model, each node transmits at the same rate using the *wireless multicast advantage* [4] to reach its neighbours in a single hop. This advantage of wireless nodes is modeled and exploited in [5] using hyperarcs. While hyperarcs characterize one-to-many transmission, they do not allow many-to-one or many-to-many transmission. With advanced physical layer processing, a node can get information from multiple transmitters simultaneously, *i.e.* nodes can operate with interference as well [6]. In this paper, we model many-to-many transmissions using hyperedges. A hyperedge is denoted by (I, J), where I is the set of transmitters and J is the set of receivers. Broadcast Channels (BC), Multiple-Access Channels (MAC) and Interference Channels (IC) are examples of channels that can be modeled as hyperedges. Our hyperedge model is different from the physical model of [7] where interference is treated as noise.

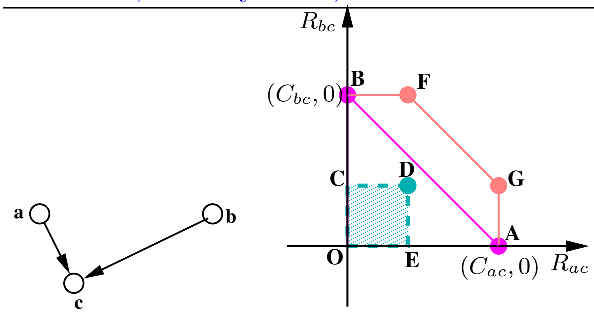
Using the capacity region of a hyperedge (I, J) determined in [8], we formulate an optimization model for determining the network coding unicast throughput from the source to the sink in the network. Numerical evaluation of throughput using the optimization model on a combination network and a diamond network shows significant gains in the network throughput. The throughput gain for combination network is around 30 % whereas it is around 16 % for the diamond network. We bound the throughput by transforming the given network into a layered network.

The paper is organised as follows: Section II describes various schemes to handle interference at the physical layer and the link layer. The wireless network model is described in Section III. The linear programming optimization model for unicast capacity is detailed in Section IV. Numerical evaluations of the optimization model is explained in Section V. Bound on network throughput is derived in Section VI. Advantages of interference processing is dealt in Section VII and Section VIII concludes the paper.

II. Interference in wireless networks

Consider the network shown in Fig. 1(a). Let Nodes a and b want to communicate Node c. Let C_{ac} and C_{bc} be the capacities of the link (a, c) and (b, c) and $C_{ac} = C_{bc}$. If both nodes a and b transmit simultaneously, it causes interference at the sink node c. The interference at the Node c can be handled in many ways. In wireless networks, slotted transmission and link scheduling are mostly commonly done to avoid interference. This time sharing rate region is the region OAB in Fig. 1(b).

Alternatively, the interfering signal can be treated as noise while decoding the desired user signal at the physical layer. This rate region is the region OCDE as shown in Fig. 1(b). Region OBFGA is obtained by processing the interference using advanced physical layer processing. From the rate regions, we observe that processing the interference has better sum rate than the other methods. This promises substantial gains in the throughput. Hence, in this paper, we study the wireless



(a) An example network (b) Rate region

Fig. 1. An example network and rate regions.

network throughput when the nodes are allowed to perform interference processing.

III. Wireless Network Model

Consider n-node wireless packet networks. All nodes are half-duplex and have uniform transmission range c_n. The network is modeled as a graph, $G = (V, E)$, where $V = \{1, 2, \cdots, n\}$ is the set of vertices and $E = \{(i, j) : d_{ij} \leq c_n, i, j \in V\}$ is the set of edges. Here, d_{ij} is the Euclidean distance between Node i and Node j. A link $(i, j) \in E$ is lossless and has a capacity C_{ij} packets per unit-time. Let $N_i = \{j \in V : d_{ij} \leq c_n\}$ be the set of neighbors of Node i. In such a network G, we consider a single unicast session from a source $S \in V$ to a destination $D \in V$.

Consider a hyperedge $\mathcal{H} = (I, J)$ where $I = \{t_1, \ldots, t_M\}$ is the set of transmitters and $J = \{r_1, r_2, \ldots, r_N\}$ is the set of receivers. Let $V_{\mathcal{H}} = I \cup J$ be the set of vertices in $\mathcal{H}$ and $E_{\mathcal{H}} \subseteq \{(t_i, r_j) : t_i \in I, r_j \in J\}$ be the set of edges in $\mathcal{H}$. For each $v \in V_{\mathcal{H}}$, define the neighbouring sets $\Gamma_+(v) = \{t_i \in V_{\mathcal{H}} : (t_i, v) \in E_{\mathcal{H}}\}$ and $\Gamma_-(v) = \{r_j \in V_{\mathcal{H}} : (v, r_j) \in E_{\mathcal{H}}\}$. Let $d_v^{\pm} = |\Gamma_{\pm}(v)|$ be the incoming/outgoing degrees. Let $\mathcal{A}$ denote the set of all hyperedges in G.

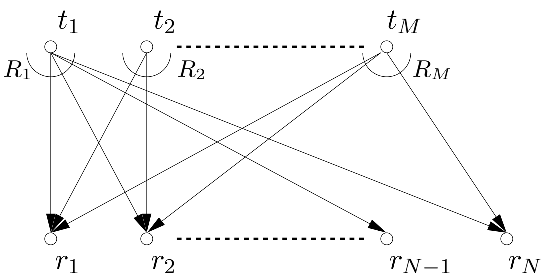


Fig. 2. An example of a hyperedge along with transmission rates

An example of such a hyperedge is shown in Fig. 2. The transmitter $t_k \in I$ sends a common message at rate R_k to the receivers in $\Gamma_-(t_k)$ and not to all receivers in J. This is a single hop communication using the inherent wireless broadcast nature. The power and bandwidth remain the same as that of the point-to-point communication. Receiver $r_j \in J$ needs to decode the codewords from the transmitters in $\Gamma_+(r_j)$. These receivers employ Interference-Aware Physical Layer (I-APL) processing techniques like successive interference cancellation for successful decoding.

IV. Optimization Model

For the single unicast session from the source $S \in V$ to the destination $D \in V$, we now describe the linear programming model to determine the network throughput. Before that, we quickly explain the selection of non-interfering graphs for avoiding the interference between hyperedges and the slotted transmission scheme.

A. Non-interfering subgraphs

To avoid interference between hyperedges, slotted transmission is employed. A set of *non-interfering* hyperedges, called a non-interfering subgraph of G, is active in each slot. Two hyperedges (I_1, J_1) and $(I_2, J_2) \in \mathcal{A}$ are non-interfering, if (i) $(\cup_{i_1 \in I_1} N_{i_1}) \cap J_2 = \emptyset$, and (ii) $(\cup_{i_2 \in I_2} N_{i_2}) \cap J_1 = \emptyset$. Otherwise, the hyperedges are said to be interfering.

The conflict graph approach of [9] is used to form the non-interfering subgraphs. Each node in the conflict graph corresponds to a hyperedge from $\mathcal{A}$. Two nodes in the conflict graph are connected if the corresponding hyperedges interfere in G. It is easy to see that every independent set in such a conflict graph will correspond to a non-interfering subgraph of G. In our numerical evaluation, for simplicity, we use an algorithm that generates one random independent set of the conflict graph in every run [9]. Note that we allow interference within a hyperedge, but avoid interference between hyperedges.

B. Transmission slots and packet injection rate

To proceed with the model, we suppose that M maximal non-interfering subgraphs of G have been generated (using M runs of the conflict graph independent set algorithm) and denote them $A_1, A_2, \cdots, A_M$. The multicast session is modeled to be spread over M transmission slots in one time unit. The duration of slot k corresponds to λ_k fraction of time and the non-interfering subgraph A_k is active during slot k (λ_k could be zero).

If $(I, J) \in A_k$, nodes $i \in I$ transmit to nodes in $\Gamma_-(i)$ during slot k. Let a_{IJ} indicate the total fraction of time for which the hyperedge (I, J) is active over M transmission slots. Let z_{iIJ} (packets per unit time) be the average rate at which packets are injected by Node $i \in I$ into the hyperarc (I, J) averaged over all slots. The transmission rates z_{iIJ} are proportional to the ON-time a_{IJ}. A multicast throughput of f is achieved if f packets are sent from the source s and received by all sinks $t \in T$ in one time unit.

C. Linear programming for unicast throughput

We assume that all links in G have equal capacity *i.e.* $C_{ij} = L$ for all $(i, j) \in E$. The constraints in the linear program are described below.

Scheduling constraints: The scheduling constraints on a_{IJ} use the indicator function $g_k(I, J)$ defined as $g_k(I, J) = 1$ if $(I, J) \in A_k$ and 0 otherwise. Intuitively, the scheduling constraints imply that a_{IJ} is upper-bounded by the total time for which the hyperedge (I, J) is active.

Flow constraints: The flow variable x_{iIJj} denotes the average information flow rate from Node i to Node $j \in \Gamma_-(i)$ along a hyperedge (I, J) towards the sink D. The average flows to each sink satisfy the flow constraints at each node.

Capacity constraints: For each hyperedge (I, J), the capacity region is stated in Lemma 1 of [8]. In that capacity region, we fix the time sharing random variable Q as a constant for simplicity. Therefore, for each hyperedge (I, J), the average flows to each sink from Node $i \in I$ along different $j \in \Gamma_-(i)$ lies within the feasible rate region for the broadcast channel (**BC**) from Node i to $\Gamma_-(i)$. Also for each hyperedge (I, J), the transmission rates z_{iIJ} are within the multiple access channel (**MAC**) from $\Gamma_+(j)$ to Node $j \in J$.

The linear program is as follows: Maximize the throughput f subject to

Scheduling Constraints:

$$\sum_k \lambda_k g_k(I,J) - a_{IJ} \geq 0, \quad \forall (I,J) \in \mathcal{A},$$

$$\sum_k \lambda_k \leq 1, \; \lambda_k \geq 0,$$

Flow Conservation Constraints:

$$\sum_{\{(I,J)\in\mathcal{A}:\, i\in I\}} \sum_{j\in\Gamma_-(i)} x_{iIJj} - \sum_{\{(J,I)\in\mathcal{A}\,:\, i\in I\}} \sum_{j\in\Gamma_+(i)} x_{jJIi} = \begin{cases} f & \text{if } i = S \\ -f & \text{if } i = D \\ 0 & \text{otherwise} \end{cases} ,\forall i \in V,$$

Capacity Conservation Constraints:

BC: $$z_{iIJ} - \sum_{j\in\Gamma_-(i)} x_{iIJj} \geq 0, \quad \forall\, i \in I,\ \forall\,(I,J) \in \mathcal{A},$$

$$z_{iIJ} \geq 0, \quad x_{iIJj} \geq 0,$$

MAC: $$\sum_{i\in\mathcal{B}} z_{iIJ} \leq a_{IJ} I(X(\mathcal{B}); Y_j | X(\mathcal{B}^c)), \quad \mathcal{B} \subseteq \Gamma_+(j),$$

$$\forall\, j \in J, \quad (I,J) \in \mathcal{A}.$$

Every feasible solution to the linear programming problem corresponds to a valid network code of throughput of f packets per unit time over M slots with each slot active for λ_k fractional time units [5], [10].

V. Numerical Evaluation

We evaluate the throughput using the proposed optimization model on (i) a diamond network and (ii) a $(5, 4)$ combination network . We assume all the links are AWGN channels with capacity one. The number of hyperedges in wireless networks is typically very large. Hence, we consider only the hyperedges shown in Fig. 3 along with their subgraphs for ease of numerical evaluation. We run the conflict graph scheduling algorithm 3,000 times to generate sufficient number (say M) of non-interfering subgraphs which obtain good estimate of network throughput.

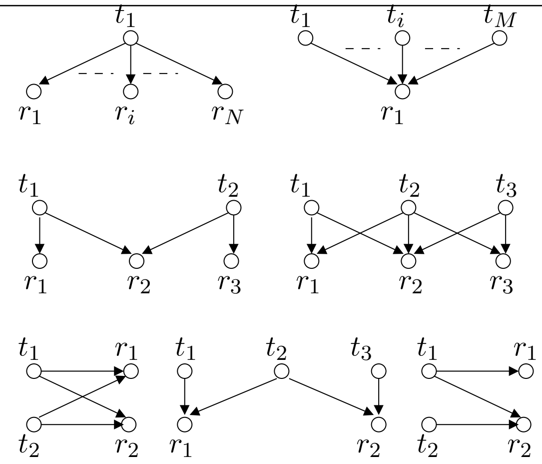
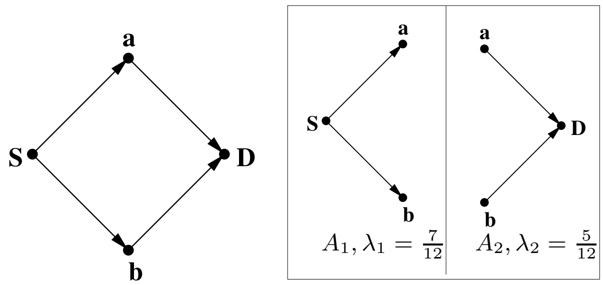


Fig. 3. Hyperedge channels used for numerical evaluation.

A. Diamond network

Consider the diamond network shown in Fig. 4(a). The optimization model is evaluated with the seven hyperedge channels shown in Fig. 3 to determine the throughput from the source S to the destination D. The non-interfering subgraphs chosen are shown in Fig. 4(b). The throughput is bounded by the broadcast cut at the source and the multiple-access cut at the Sink. Therefore, $f \leq \min(1, 1.4037)$. The unicast throughput with these seven hyperedges is $f = \frac{7}{12}$ whereas it is $f = \frac{1}{2}$ with interference avoidance. Notice that, the MAC channel in A_2 is very useful in obtaining higher throughput.



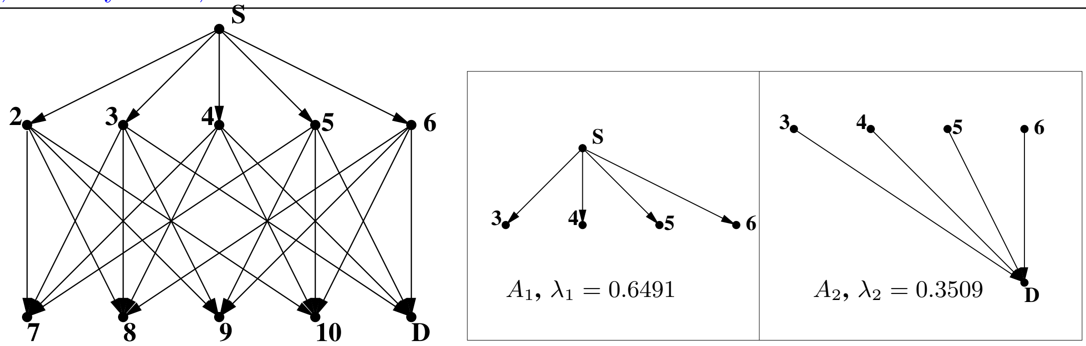
(a) Diamond network. (b) Non-interfering subgraphs.

Fig. 4. Unicast throughput of a diamond network.

Transmission scheme: Consider twelve time slots and seven information bits. In the first seven slots source broadcasts all the seven information bits to Nodes a and b. The MAC channels in A_2 operate at the point $(R_{aD}, R_{bD}) = (1, \frac{2}{5})$ to enable interference processing. The nodes a and b select the codebooks accordingly. In slots $8 - 12$, Nodes a and b use their MAC codebooks to transmit all seven information bits to sink D *i.e.*, $5(R_{aD} + R_{bD}) = 7$.

B. $(5, 4)$ combination network

Consider the $(5, 4)$ combination network shown in Fig. 5(a). The optimization model is evaluated with the seven hyperedge channels shown in Fig. 3 to determine the throughput from the source S to the destination D. The non-interfering subgraphs



(a) $(5, 4)$ combination network. (b) Non-interfering subgraphs.

Fig. 5. Unicast throughput of a $(5, 4)$ combination network.

chosen are shown in Fig. 5(b). The unicast throughput is bounded by the broadcast cut at the source and the multiple-access cut at the sink. Therefore, $f \leq \min(1, 1.8502)$. The unicast throughput with these seven hyperedges is $f = 0.6491 \approx \frac{9}{14}$ whereas it is $f = \frac{1}{2}$ with interference avoidance. Notice that, the MAC channel in A_2 is very useful in obtaining higher throughput. This implies that interference processing at the receivers helps in obtaining higher throughput

Transmission scheme: Consider fourteen time slots and nine information bits. In the first nine slots source broadcasts all the nine information bits to Nodes 3, 4, 5 and 6. The MAC channels in A_2 operate at the point $(R_{3D}, R_{4D}, R_{5D}, R_{6D}) = (1, 0.40, 0.25, 0.15)$ to enable interference processing. The Nodes 3, 4, 5 and 6 select the codebooks accordingly. In slots $10 - 14$, Nodes 3, 4, 5 and 6 use their MAC codebooks to transmit all nine information bits to sink D *i.e.*, $5(R_{3D} + R_{4D} + R_{5D} + R_{6D}) = 9$.

VI. Bound on Network Throughput

In this section, we suggest approaches that simplify the computations and provide approximate solutions for large networks. For networks with large number of nodes, the linear programming problem becomes unsolvable because of the exponential number of variables and constraints. To do this, the given network G is transformed into a layered network G_L in a manner explained below. Each layer in G_L is a hyperedge. Let m be the number of layers in G_L. The construction of G_L is as follows:

(i) Start with source node S as the root node on G_L. Therefore, the hyperedge associated with layer 1 is $(I_1, J_1) = (S, N_S)$.

(ii) Any k^{th} intermediate layer in G_L is constructed as:

$$\begin{aligned} I_k &= J_{k-1}, \\ J_k &= N(I_k) \setminus I_k, \end{aligned}$$

for $k = 2, 3, \ldots$. Here $N(I_k) = \cup_{i \in I_k} N_i$. This procedure continues until the sink node is contained in the m^{th} hyperedge *i.e.*, $D \in J_m$. The layered network is also useful to derive a bound on the network throughput.

A. Upper bound on the layered network throughput

The information flow f' in G_L from source node S to sink node D is through the layers in it. Therefore, the flow f' is restricted by the minimum of the maximum amount of information flow in each layer *i.e.*,

$$f' \quad \leq \quad \min_{1 \leq k \leq m} \mathcal{R}_s^k,$$

where $\mathcal{R}_s^k$ is the maximum amount of information flow in the k^{th} layer (hyperedge). Notice that, this bound does not account for interference between hyperedges and *half duplex* nature of wireless nodes.

Now, we investigate the sum rate of the k^{th} hyperedge (layer) (I_k, J_k). This is the maximum amount of information flow from the transmitters in $t_i \in I_k$ to the receivers in $r_j \in J_k$. Let X_i be the codeword sent by transmitter t_i and Y_j be the codeword received by receiver r_j.

Each transmitter $t_i \in I$ sends a common information to all its receivers at rate R_i. Therefore, this rate R_i is bounded by:

$$R_i \leq C_i, \tag{1}$$

where $C_i = \min_{j \in \Gamma_-(t_i)} I(X_i; Y_j)$. Since mutual information is a non-negative quantity, the *simplest upper bound* on the sum rate of the hyperedge with interference processing $\mathcal{R}_s^k$ is the following:

$$\mathcal{R}_s^k = \sum_{t_i \in I_k} R_i \leq \sum_{t_i \in I_k} C_i. \tag{2}$$

However, *tighter bounds* on the sum rate of the hyperedge (I_k, J_k) with interference processing can be obtained by considering additional bounds on the MAC reception at the receivers from [8].

VII. Advantages of Interference Processing

We consider some example hyperedge channels to illustrate the benefits of physical layer interference processing.

A. Orthogonal channels

In the hyperedge $\mathcal{H}$, let $M = N$ and t_i be connected only to the receiver r_i, for all $i = 1, 2, \ldots, M$. This forms a set of M parallel independent channels in $\mathcal{H}$. Therefore, the sum rate in $\mathcal{H}$ with interference processing is:

$$\mathcal{R}_s = \sum_{i=1}^{M} C_i.$$

This is also the sum-rate achievable with interference avoidance. It implies that interference processing is not useful in orthogonal channels when compared to interference avoidance.

B. Complete bipartite graph

Suppose the hyperedge channel $\mathcal{H}$ is complete *i.e.*, each transmitter $t_i \in I$ is connected to all receivers $r_j \in J$. The sum rate possible with interference processing is bounded as

$$\begin{aligned} \mathcal{R}_s &= \sum_{i=1}^{M} R_i, \\ &\leq \min_{r_j \in J} I(X^{(\Gamma_+(r_j))}; Y_j). \end{aligned} \tag{3}$$

In interference avoidance, a receiver is allowed to receive from only one transmitter.Therefore, the sum-rate with interference avoidance is the maximum of the transmission rates $\max_{t_i \in I} R_i$.

In (3), if the links are AWGN channels with equal transmission power P, the bound becomes

$$\mathcal{R}_s \leq \min_{r_j \in J} \frac{1}{2} \log_2 \left(1 + \frac{MP}{\sigma_j^2}\right),$$

where σ_j^2 is the noise variance at the receiver r_j. With interference avoidance, the corresponding bound is

$$\mathcal{R}_s \leq \max_{r_j \in J} \frac{1}{2} \log_2 \left(1 + \frac{P}{\sigma_j^2}\right).$$

For the complete bipartite graph example,we see that the sum rate can potentially improve because of interference processing. In section V, we notice significant gains in the network throughput because of the increased sum-rate in each hyperedge. In general, when the receivers in J receives signals from multiple transmitters, it is very useful to use interference processing.

VIII. Conclusion

We considered wireless relay networks with the relays adopting decode-and-forward strategy and performing network coding. We allowed the transmitters to do common broadcast transmission and the receivers to do interference processing but did not require either MIMO cooperation or asymptotically large SNR. An optimization model was formulated to determine the network throughput. Numerical evaluation on some example networks promises substantial gain in network throughput because of the interference processing at the receivers.

References

[1] A. S. Avestimher, S. N. Diggavi, and D. N. C. Tse, "Approximate capacity of gaussian relay networks," in *Proc. of IEEE symposium on Information theory*, July 2008, pp. 474–478.

[2] K. Sreeram, S. Brienjith, and P. V. Kumar, "DMT of multi-hop cooperative networks-part ii: Layered and multi-antenna networks," in *Proc. of IEEE symposium on Information theory*, July 2008, pp. 2076–2080.

[3] ——, "DMT of multi-hop cooperative networks-part i: K-parallel-path networks," in *Proc. of IEEE symposium on Information theory*, July 2008, pp. 2081–2085.

[4] J. Wieselthier, G. Nguyen, and A. Ephermides, "On the construction of energy - efficient broadcast and multicast trees in wireless networks," in *Proc. of INFOCOM*, 2000, pp. 585–594.

[5] J.-S. Park, D. S. Lun, F. Soldo, M. Gerla, and M. Medard, "Performance of network coding in ad hoc networks," in *Proc. of IEEE MILCOM 2006*, October 2006.

[6] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons (Asia), 2004.

[7] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inform Theory*, vol. 46, no. 2, pp. 388–404, March 2000.

[8] M. Bama, S. Bhashyam, and A. Thangaraj, "Enhancing network coding throughput using interference-aware physical layers," *submitted to IEEE Trans. on Wireless Communications*, July 2008. [Online]. Available: www.ee.iitm.ac.in/∼ skrishna

[9] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," in *Proc. of Mobicom*, 2003.

[10] R. Ahlswede, N. Cai, S. Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inform. Theory*, vol. 46, no. 4, pp. 1204–1216, July 2000.