

Optimal Backpressure Routing for Wireless Networks with Multi-Receiver Diversity

Michael J. Neely

University of Southern California

http://www-rcf.usc.edu/~mjneely

Abstract—We consider the problem of optimal scheduling and routing in an ad-hoc wireless network with multiple traffic streams and time varying channel reliability. Each packet transmission can be overheard by a subset of receiver nodes, with a transmission success probability that may vary from receiver to receiver and may also vary with time. We develop a simple backpressure routing algorithm that maximizes network throughput and expends an average power that can be pushed arbitrarily close to the minimum average power required for network stability, with a corresponding tradeoff in network delay. The algorithm can be implemented in a distributed manner using only local link error probability information, and supports a “blind transmission” mode (where error probabilities are not required) in special cases when the power metric is neglected and when there is only a single destination for all traffic streams.

Index Terms—Broadcast advantage, distributed algorithms, dynamic control, mobility, queueing analysis, scheduling

I. INTRODUCTION

In this paper, we consider a multi-node, multi-hop wireless network with “unreliable” channels. Each transmission link has an associated error probability that may vary with time due to external factors such as environment changes or user mobility. Many previous studies assume that accurate channel information is available so that error probabilities are relatively small and can be neglected. However, in this work we consider the opposite case where precise channel information is difficult or impossible to obtain, but where simple *estimates* of channel quality can be made based on limited channel feedback. A motivating example is an underwater sensor network that uses acoustic channels with large propagation delays. This is a particularly challenging environment due to time varying wave ripple, complex signal reflections between surface and ground, and large delay spreads [1] [2]. While it may not be practical to assume that an accurate channel quality can be determined at the time of packet transmission, it is reasonable to estimate the *error probability* based on past signal strength values and/or ACK/NACK history from previous transmissions.

The problem of unreliable channels is also important in other contexts, such as mobile networks where knowledge of which receivers are within transmission range may be uncertain, or in dense ad-hoc networks where unpredictable transmissions of other nodes can act as random inter-channel interference. It is imperative to develop flexible mathematical models of such networks, and to develop robust networking

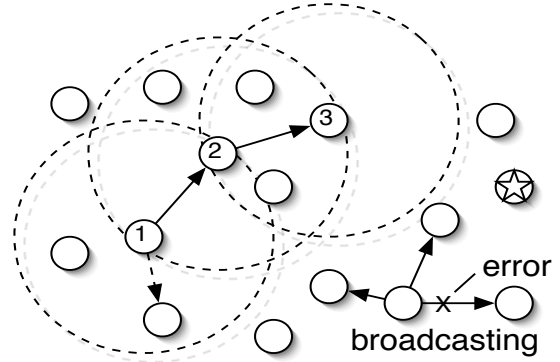


Fig. 1. A multi-hop network with channel errors and multi-receiver diversity. In this example there is a single destination indicated by the star node. Note that a “closest-to-the-destination” heuristic might result in data being routed from node 1 to 2 to 3, resulting in a deadlock.

strategies that exploit all system resources to operate efficiently in these extreme environments.

In this paper, we design robust algorithms by exploiting the *broadcast advantage* of wireless networks. Specifically, our network model includes the fact that a single packet transmission might be overheard by a *subset* of receiver nodes within range of the transmitter. This creates a *multi-receiver diversity gain*, where the probability of successful reception by at least one node within a subset of receivers can be much larger than the corresponding success probability of just one receiver alone. Hence, it is desirable to design flexible routing algorithms that do not require a single “next hop” receiver to be specified in advance. Such algorithms can dynamically adjust routing and scheduling decisions in response to the random outcome of each transmission.

The wireless broadcast advantage has been used in various contexts, for example, in [3] for the design of wireless multi-cast algorithms, and in [4] for the design of minimum energy disjoint paths. Our model and problem formulation is closest to the work by Zorzi and Rao in [5], and more recently by Biswas and Morris in [6], where efficient methods of using multi-receiver diversity for packet forwarding are explored. We note that such formulations inevitably involve situations where the same packet is *redundantly distributed* over different network nodes. A fundamental decision is whether to allow the different versions of the packet to simultaneously propagate throughout the network, or to designate only a single copy that is allowed to proceed. The work in [5] considers the

This material is based on work supported by the National Science Foundation grant OCE 0520324.

simple heuristic that shifts packet forwarding responsibilities to the receiver that is *closest to the destination*. While this scheme has many desirable properties, especially for large ad-hoc networks, it is clear that for a given network of fixed size, the “closest-to-destination” heuristic neither maximizes throughput nor minimizes average power expenditure. Further, this scheme can lead to an undesirable *deadlock mode* if data is consistently forwarded to a particular node for which there are no other next-hop receivers that are closer to the destination (see Fig. 1). Thus, it is often better to route packets along paths that temporarily take them further from the destination, especially if these paths eventually lead to links that are more reliable and/or that are not as heavily utilized by other traffic streams. The work in [6] considers a routing heuristic based on an estimated delivery cost, computed by an estimate of the expected number of hops required to reach the destination along a traditional shortest path. However, this method is not necessarily optimal in terms of energy or throughput.

There are several difficulties associated with developing a *throughput optimal* algorithm in this context. First, individual nodes might only know the error probabilities on their own outgoing links, and may not know the error rates or traffic loads on other portions of the network. Second, even if centralized network knowledge were fully available, an optimal algorithm would need to specify a contingency plan for each possible random transmission outcome. For example, suppose a given node transmits a packet for which there are k potential receivers. There are 2^k possible outcomes of this single transmission (one for each possible subset of successful receivers). An optimal algorithm would require a decision for each possible outcome, perhaps also allowing for redundant packet forwarding. Hence, the design of an optimal algorithm must overcome these geometric complexity issues. This is further complicated if there are multiple simultaneous packet transmissions and multiple traffic streams sharing the same network, and if the network topology and link error probabilities are changing with time.

In this paper, we overcome these challenges with a simple solution that uses the concept of *backpressure routing* and *Lyapunov drift*. We first show that it is possible to restrict attention to algorithms that do not allow redundant forwarding, without loss of optimality. We then show that the optimal packet commodity to transmit at each network node can be determined by a backpressure index that compares the current queue backlog of each commodity to the backlog in the potential receivers. Once a packet from this optimal commodity is transmitted, the responsibility of forwarding the packet to its destination is shifted to the receiver node that maximizes the differential backlog. Responsibility is retained by the original transmitter if no suitable receivers are found on a given transmission attempt.

Backpressure techniques of this type were first applied to multi-hop wireless networks by Tassiulas and Ephremides in [7], where throughput optimal algorithms were developed using Lyapunov drift theory. Lyapunov theory has since been a powerful mathematical tool for the development of stable scheduling strategies for wireless networks and switching systems [7]-[18], including our own work in [15]-[18] that

applies backpressure concepts to solve problems of optimal power allocation, routing, and fair flow control in wireless networks with mobility. Related work on energy efficient wireless scheduling is developed in [19]-[22]. The work in [7]-[22] does not consider the broadcast advantage of wireless networks, and assumes that all transmissions are fully reliable. Lyapunov scheduling for wireless MIMO downlinks with multiple transmit and receive antennas is considered in [23], and related MIMO results are developed for channels with errors in [24] [25]. Recent work in [26] considers backpressure techniques in combination with network coding, and work in [27] considers backpressure strategies for cooperative transmission (where multiple nodes can transmit redundant information simultaneously for a power enhancement at the receiver). Complexity issues of cooperative communication under the wireless broadcast advantage are discussed in [28]. We do not consider network coding or cooperative transmission in this paper, and restrict attention to the multi-user diversity problem for networks with errors, as described above. It is likely that our formulation can be extended to consider more sophisticated control actions by augmenting the set of decision options available to the network controller, in which case redundant packet forwarding may be required for optimality.

In the next section, we develop a simple network model in terms of (potentially time varying) link error probabilities, and specify the control decision options for this model. In Section III, we specify the *network capacity* and the *minimum average power for stability* associated with this model. In Section IV we develop the dynamic control algorithm, and in Section VI we extend the formulation to include dynamic resource allocation with variable rate and power options, where link error probabilities can depend on transmission decisions.

II. THE BASIC NETWORK MODEL

We consider a timeslotted system with slots normalized to integral units $t \in \{0, 1, 2, \dots\}$. There are N network nodes and L potential transmission links (possibly a single link for each node pair (a, b)). All data arrives randomly to the network in packetized units, and we let $A_n^{(c)}(t)$ represent the number of packets that exogenously arrive to network node n during slot t that are intended for delivery to network node c . All packets destined for a particular node c are defined as *commodity c packets*. Arrivals are assumed to be i.i.d. over timeslots, and we let $\lambda_n^{(c)} = \mathbb{E}\{A_n^{(c)}(t)\}$ represent the arrival rate of commodity c data into source node n (in units of packets/slot). Internal network queues store packets according to their commodities. Each packet is assumed to have an appropriate header field with commodity and packet number identifiers.

We assume that at most one packet can be transmitted from any given node during a single timeslot, and let $\mu_n(t)$ represent the number of packets transmitted by node n during slot t (where $\mu_n(t) \in \{0, 1\}$). Transmission opportunities are determined by an underlying random access or time division multiple access (TDMA) structure, and we let $\chi_n(t)$ represent a 0/1 process which is 1 if and only if node n is allowed to transmit during slot t . Each packet transmission is assumed to expend a constant amount of power P_{tran} , and is successfully

received by the other nodes of the network according to reception probabilities $q_{nk}(t)$ (for $n, k \in \{1, \dots, N\}$). For convenience, we define the *network topology state process* $S(t)$ as the collective process of all node transmission capabilities and link conditions at time t , so that transmission opportunities and link probabilities can be determined as functionals of $S(t)$ as follows:

$$\begin{aligned}\chi_n(t) &= \hat{\chi}_n(S(t)) \\ q_{nk}(t) &= \hat{q}_{nk}(S(t))\end{aligned}$$

Let $\mathcal{K}_n(t)$ represent the set consisting of all potential receivers for node n during slot t (which can potentially change from slot to slot if the network is mobile). The set $\mathcal{K}_n(t)$ can generally contain all $N - 1$ other network nodes, although it typically has a much smaller size and consists only of those nodes within realistic transmission range of node n . Error events for a single packet transmission can be correlated over various links, and hence a more complete characterization of each transmitter n is given by probabilities $q_{n,\Omega_n}(t)$, where Ω_n is a subset of nodes within the receiver set $\mathcal{K}_n(t)$, and $q_{n,\Omega_n}(t)$ represents the probability that the set of all nodes that successfully receive the packet transmitted by node n is exactly given by the subset Ω_n . This probability is also determined as a functional of the topology state process:

$$q_{n,\Omega_n}(t) = \hat{q}_{n,\Omega_n}(S(t))$$

The error events of different packet transmissions from different nodes may also be correlated, and these correlations in principle are also determined by the topology state process $S(t)$. However, we shall find that these additional correlations are irrelevant to network capacity and optimal control.

For analytical purposes, the network topology state $S(t)$ is assumed to take values in a finite (but arbitrarily large) state space \mathcal{S} . We note that the success probabilities of a given link or set of links are completely determined by the network topology state process $S(t)$. That is, given $S(t)$, these probabilities are not affected by the transmission decisions $\mu_n(t)$ for $n \in \{1, \dots, N\}$. This assumption is reasonable if all transmitting nodes use orthogonal signals, or if inter-channel interference can be approximated as randomly and independently influencing the channel probabilities. A more general model where channel probabilities can depend on transmission decisions is considered in Section VI.

A. A Timing Diagram for One Timeslot

The timing diagram of Fig. 2 illustrates our model of information exchange between nodes. The events that take place between a transmitting node n and a potential receiver node k during a single timeslot are outlined in the diagram. At the beginning of the timeslot, channel probability information and any necessary control information is passed between the two nodes. This can possibly take place over a dedicated control channel, or might be implemented by appending header information to packets transmitted on previous slots. Next, the transmitter node n observes the transmission opportunity process $\hat{\chi}_n(S(t))$. If $\hat{\chi}_n(S(t)) = 0$ then node n does not transmit, while if $\hat{\chi}_n(S(t)) = 1$ the node can decide whether

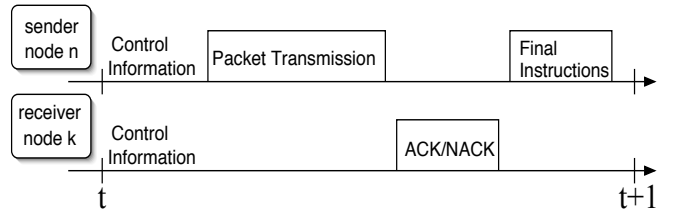


Fig. 2. A timing diagram illustrating the events within a single timeslot.

or not it desires to transmit a packet. If it decides to transmit, it chooses a particular packet and transmits it with power P_{tran} , for a fixed amount of time as indicated in the timing diagram.

Every potential receiver node then provides immediate ACK/NACK feedback to the transmitter, informing the transmitter if the packet was successfully received. The absence of an ACK signal is considered to be equivalent to a NACK (this treats the case when the receiver node did not detect any transmission). The transmitter node accumulates all of the ACK responses, and then transmits a final message that informs the successful receivers of all other successful receivers. This final transmission possibly also provides instructions for future packet forwarding.

The 3-part handshake of the timing diagram (transmission, ACK/NACK, and final message) is designed to cleanly describe a system where transmission outcomes are known to all relevant nodes at the end of a single timeslot. This facilitates mathematical analysis. However, in practice the last two steps of the handshake may take place by appending this information to the packet header of future packet transmissions. This creates a system with *delayed feedback information*, which in principle does not affect throughput optimality (provided some regularity assumptions hold concerning the timeliness of the feedback) but may affect end-to-end network delay, as discussed in more detail in Section VII. Throughout this paper, we make the idealistic assumption of *perfect control information*, so that the control signals themselves are not subject to errors. In particular, for the timing diagram of Fig. 2, it is assumed that if a packet transmitted at node n was successfully received at node k , then the channel from k to n and from n to k is good enough for the remaining parts of the handshake to be successful. This is a reasonable assumption if forward and backward channels are relatively similar for the duration of a timeslot, or if the dedicated control channel is reliable. The possibility of control channel errors can create another situation of delayed feedback information, and this is also briefly discussed in more detail in Section VII.

B. Network Objective and Control Decision Variables

The goal is to design a control algorithm that stabilizes the network whenever possible. Further, the *average power cost* should be as small as possible. Specifically, for a power vector $\mathbf{P} = (P_1, \dots, P_N)$, we define the separable cost function $h(\mathbf{P}) = h_1(P_1) + \dots + h_N(P_N)$, where each component $h_n(P_n)$ is non-negative, continuous, and has the property that $h_n(0) = 0$. The power expended on each timeslot t is given by the vector $\mathbf{P}(t) \triangleq P_{tran} \cdot (\mu_1(t), \dots, \mu_N(t))$, and the time

average power cost \bar{h} is defined:

$$\bar{h} \triangleq \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E} \{h(\mathbf{P}(\tau))\}$$

Note that choosing $h(\mathbf{P}) = \sum_{n=1}^N P_n$ coincides with the objective of minimizing the time average expected power expenditure. Under our simple network model, we have $P_n(t) \in \{0, P_{tran}\}$ for all t , so that $h_n(P_n(t)) \in \{0, h_n(P_{tran})\}$. In this case, the $h_n(\cdot)$ function plays only a limited role in generalizing the minimum average power objective, although it shall be more meaningful in the extended formulation of Section VI that considers a continuum of power options.

In Section III we show that throughput and energy optimality can be achieved without using redundant packet forwarding. This allows the following more detailed network queuing variables and control decision variables to be defined. At each timeslot t , every network node n makes a transmission decision $\mu_n(t)$ subject to $\mu_n(t) \in \{0, 1\}$ and $\mu_n(t) = 0$ whenever $\chi_n(t) = 0$. It then chooses a packet commodity to transmit by selecting control variables $\mu_n^{(c)}(t)$ subject to:

$$\mu_n^{(c)}(t) \in \{0, 1\}, \sum_{c=1}^N \mu_n^{(c)}(t) \leq \mu_n(t), \mu_c^{(c)}(t) = 0 \quad (1)$$

That is, $\mu_n^{(c)}(t)$ represents an opportunity for commodity c packet transmission by node n during slot t . This can be either 0 or 1, but can be 1 for at most one commodity c . We set $\mu_c^{(c)}(t) = 0$ as it does not make sense to retransmit a packet that has already reached its destination. We say that $\mu_n^{(c)}(t)$ is a *transmission opportunity* because it is useful to imagine the possibility of choosing these decision variables independent of queue backlog. In cases when a transmission opportunity arises but there is no commodity c packet available, then no packet is actually transmitted.

We let $H_{nk}(t)$ represent the random variable that is 1 if a packet transmitted from node n was successfully received by receiver k , and zero otherwise. After receiving ACK/NACK feedback, node n selects a new node to take responsibility for the packet (possibly choosing itself), and informs its receivers of the choice. This is done according to control decision variables $\beta_{nk}^{(c)}(t)$, representing the number of commodity c packets whose responsibility is shifted from node n to node k during slot t , where:

$$\begin{aligned} \beta_{nk}^{(c)}(t) \in \{0, 1\} \quad , \quad \beta_{nk}^{(c)}(t) \leq \mu_n^{(c)}(t) H_{nk}(t) \\ \beta_{nn}^{(c)}(t) = 0 \quad , \quad \sum_{k=1}^N \beta_{nk}^{(c)}(t) \leq 1 \end{aligned} \quad (2)$$

That is, the $\beta_{nk}^{(c)}(t)$ variables are either 0 or 1, can be 1 only if a commodity c transmission opportunity occurs on slot t and $H_{nk}(t) = 1$, and can be 1 for at most one receiver node k (where such a node k is necessarily in the set of potential receivers $\mathcal{K}_n(t)$). If $\beta_{nk}^{(c)}(t) = 0$ for all $k \in \mathcal{K}_n(t)$, then node n retains responsibility for the packet. It shall be convenient to also allow these decision variables to be independent of queue backlog, and so both $\beta_{nk}^{(c)}(t)$ and $\mu_n^{(c)}(t)$ can potentially equal 1, regardless of whether or not node n was holding a commodity c packet that it actually transmitted. In this case, the $H_{nk}(t)$ value is viewed as a random variable that is distributed the same as if a packet had actually been transmitted. The actual control decisions $\beta_{nk}^{(c)}(t)$ in the case of no packet transmission

are irrelevant as they do not affect the system. However, it is useful to formally allow choosing non-zero $\beta_{nk}^{(c)}(t)$ values in this case. Specifically, we find it useful for mathematical proofs to imagine the existence of a *stationary randomized* control policy that chooses decision variables independently of queue backlog, but where no packets are actually transferred if these decisions attempt transmission from an empty queue.

Packets are stored at every node according to their commodity, and we define $U_n^{(c)}(t)$ as the current number of commodity c packets in node n at the beginning of slot t . The $U_n^{(c)}(t)$ process takes values in the set of non-negative integers, and evolves according to the following queuing dynamics:

$$\begin{aligned} U_n^{(c)}(t+1) \leq \max \left[U_n^{(c)}(t) - \sum_{k=1}^N \beta_{nk}^{(c)}(t), 0 \right] \\ + \sum_{a=1}^N \beta_{an}^{(c)}(t) + A_n^{(c)}(t) \end{aligned} \quad (3)$$

The expression above is an inequality rather than an equality because the actual endogenous arrivals to node n may be less than $\sum_{a=1}^N \beta_{an}^{(c)}(t)$ if there are little or no actual commodity c packets transmitted from the other nodes $a \neq n$. We formally define $U_n^{(n)}(t)$ to be zero for all n and all t .

III. NETWORK CAPACITY AND MINIMUM POWER

Here we define the optimal throughput and average power cost operating points. The *network layer capacity region* Λ is defined as the closure of all input rate matrices ($\lambda_n^{(c)}$) that can be stabilized by the network according to some control algorithm, perhaps an algorithm that uses redundant packet forwarding. We note that this notion of capacity assumes that network control actions are within the scope of the system model described in Section II, and in particular this model does not include the possibility of cooperative transmission or network coding, which can potentially improve performance.

Suppose that the network topology state process $S(t)$ takes values on a finite state space \mathcal{S} , and has well defined time average probabilities π_s for each $s \in \mathcal{S}$. For each node n , let \mathcal{H}_n denote the set of all subsets Ω_n of $\{1, \dots, N\} - \{n\}$. For each subset Ω_n , recall that $\hat{q}_{n, \Omega_n}(s)$ is the probability that Ω_n is exactly the set of all successful receivers of a packet transmitted by node n , given such a packet is transmitted when the topology state is given by $S(t) = s$.

Theorem 1: (Network Capacity and Minimum Cost) The network capacity region Λ consists of all rate matrices ($\lambda_n^{(c)}$) for which there exist *multi-commodity flow variables* $\{f_{nk}^{(c)}\}$ together with probabilities $\alpha_n^{(c)}(s)$, $\theta_{nk}^{(c)}(\Omega_n)$ for all n, k, c , all topology states $s \in \mathcal{S}$, and all subsets $\Omega_n \in \mathcal{H}_n$, such that:

$$f_{ab}^{(c)} \geq 0, f_{cb}^{(c)} = 0, f_{aa}^{(c)} = 0 \quad (4)$$

$$\sum_a f_{an}^{(c)} + \lambda_n^{(c)} \leq \sum_b f_{nb}^{(c)} \quad \text{for all } n \neq c \quad (5)$$

$$\sum_c f_{nk}^{(c)} \leq \sum_c \sum_{s \in \mathcal{S}} \pi_s \alpha_n^{(c)}(s) \left[\sum_{\Omega_n \in \mathcal{H}_n} \hat{q}_{n, \Omega_n}(s) \theta_{nk}^{(c)}(\Omega_n) \right] \quad (6)$$

where (4) holds for all $a, b, c \in \{1, \dots, N\}$, (6) holds for all links (n, k) , and where the probabilities $\theta_{nk}^{(c)}(\Omega_n)$ satisfy for all $\Omega_n \in \mathcal{H}_n$:

$$\theta_{nk}^{(c)}(\Omega_n) = 0 \text{ if } k \notin \{\Omega_n \cup \{n\}\} \quad , \quad \sum_{k=1}^N \theta_{nk}^{(c)}(\Omega_n) = 1$$

and for all $s \in \mathcal{S}$ the $\alpha_n^{(c)}(s)$ probabilities satisfy:

$$\sum_{c=1}^N \alpha_n^{(c)}(s) \leq 1, \quad \alpha_n^{(c)}(s) = 0 \text{ if } \hat{\chi}_n(s) = 0$$

Furthermore, the minimum average power cost required for network stability is given by the value \bar{h}^* that minimizes the following metric:

$$\bar{h}^* = \sum_{s \in \mathcal{S}} \pi_s \left[\sum_{n=1}^N \sum_{c=1}^N \alpha_n^{(c)}(s) h_n(P_{tran}) \right]$$

over all $\{f_{nk}^{(c)}\}, \alpha_n^{(c)}(s), \theta_{nk}^{(c)}(\Omega_n)$ variables that satisfy (4)-(6).

The above theorem is similar to the capacity theorem of [16] [15], where the constraints (4) represent non-negativity and flow efficiency constraints for the flow variables $\{f_{ab}^{(c)}\}$, the constraints (5) represent flow conservation constraints, and the constraints (6) represent link constraints for each link (n, k) . Each $\alpha_n^{(c)}(s)$ value can be interpreted as the conditional probability that node n transmits a commodity c packet given that $S(t) = s$. Each $\theta_{nk}^{(c)}(\Omega_n)$ value can be interpreted as the conditional probability that node n shifts packet forwarding responsibilities to node k , given that node n transmits a commodity c packet that is heard exactly by the subset Ω_n of receivers. With this interpretation, the theorem can be simplified according to the following corollary.

For each input rate matrix $\lambda = (\lambda_n^{(c)}) \in \Lambda$, we define $\Phi(\lambda)$ as the minimum power cost \bar{h}^* required to stabilize the system. Suppose that the input rate matrix is interior to the capacity region, so that there exists a positive value ϵ such that $(\lambda_n^{(c)} + \epsilon 1_n^{(c)}) \in \Lambda$, where $1_n^{(c)}$ is an indicator function equal to 1 if and only if $n \neq c$, and zero else.

Corollary 1: If the topology state $S(t)$ is i.i.d. over timeslots, then a rate matrix $(\lambda_n^{(c)} + \epsilon 1_n^{(c)})$ is in the capacity region Λ if and only if there exists a stationary randomized algorithm that chooses control decision variables $\mu_n^{(c)}(t)$ and $\beta_{nk}^{(c)}(t)$ (according to the constraints specified in Section II-B) based only on the current topology state $S(t)$ (and hence independent of current queue backlog), to yield:

$$\sum_a \mathbb{E} \left\{ \beta_{an}^{(c)}(t) \right\} + \lambda_n^{(c)} + \epsilon \leq \sum_b \mathbb{E} \left\{ \beta_{nb}^{(c)}(t) \right\} \quad \forall n \neq c \quad (7)$$

$$\mathbb{E} \{ h(\mathbf{P}(t)) \} = \Phi(\lambda + \epsilon) \quad (8)$$

where $\epsilon = (\epsilon 1_n^{(c)})$ and $\mathbf{P}(t) = P_{tran} \cdot (\mu_1(t), \dots, \mu_N(t))$. The expectations in (7) and (8) are taken with respect to the random topology state $S(t)$ and the random control decisions based on this topology state, and do not depend on queue backlog.

The above theorem and its corollary demonstrate that for any rate matrix $(\lambda_n^{(c)}) \in \Lambda$, there exists a stationary randomized algorithm (with probabilities precisely matched to the network traffic rates and topology state probabilities) that can achieve a multi-commodity flow that supports the input rate matrix by routing all data to its proper destination, and that incurs an average power cost exactly given by \bar{h}^* . However, even if all topology state probabilities π_s were fully known, the geometric complexity of the optimization problem in Theorem 1 demonstrates the extreme difficulty of directly solving for the parameters required to implement such a policy.

Theorem 1 is proven by first showing that the constraints (4)-(6) are *necessary* for network stability. The *sufficiency* part of the theorem is proven by constructing a stabilizing algorithm for any rate matrix $(\lambda_n^{(c)})$ that is *interior* to the capacity region (so that $(\lambda_n^{(c)} + \epsilon 1_n^{(c)}) \in \Lambda$, for some positive value ϵ). Such stabilizing policies can be constructed with resulting average power costs that are arbitrarily close to \bar{h}^* (by choosing ϵ arbitrarily small), with a corresponding tradeoff in end-to-end network delay. The proof of necessity uses the finite state space assumption for the topology state variable $S(t)$, and is related to similar proofs of capacity and minimum energy in [16] [17] [15] (proof omitted for brevity). Sufficiency does not require the finite state space property, and is proven in the next section, where a simple dynamic control algorithm is constructed that can be implemented in real time.

IV. THE DYNAMIC CONTROL ALGORITHM

We have the following dynamic control algorithm, defined in terms of a non-negative control parameter V that determines the degree to which we emphasize power cost minimization.

Diversity Backpressure Routing (DIVBAR): Every timeslot t , each network node n observes the queue backlogs in each of its potential receiver nodes $k \in \mathcal{K}_n(t)$, and observes the current link channel probabilities associated with its receivers. Each node n determines if $\chi_n(t) = 1$ (i.e., it determines if a transmission opportunity is available on the current slot). If so, it performs the following operations:

- 1) For each commodity c and each receiver $k \in \mathcal{K}_n(t)$, the *differential backlog weights* $W_{nk}^{(c)}(t)$ are computed as follows:

$$W_{nk}^{(c)}(t) = \max[U_n^{(c)}(t) - U_k^{(c)}(t), 0] \quad (9)$$

That is, the weight $W_{nk}^{(c)}(t)$ is equal to the difference between the commodity c backlog in node n and the commodity c backlog in node k (maxed with zero).

- 2) The receivers $k \in \mathcal{K}_n(t)$ are *priority ranked* according to the $W_{nk}^{(c)}(t)$ weights, so that receivers with larger weights are ordered with higher priority (breaking ties arbitrarily). We define $k(n, c, t, b)$ as the node $k \in \mathcal{K}_n(t)$ with the b th largest weight $W_{nk}^{(c)}(t)$ for commodity c . Thus, by definition we have:

$$W_{n, k(n, c, t, 1)}^{(c)}(t) \geq W_{n, k(n, c, t, 2)}^{(c)}(t) \geq W_{n, k(n, c, t, 3)}^{(c)}(t) \dots$$

- 3) Define $\phi_{nk}^{(c)}(t)$ as the probability that a packet transmission from node n is correctly received by node k , but is *not* received by any other nodes $\hat{k} \in \mathcal{K}_n(t)$ that are ranked with higher priority than node k according to the commodity c rank ordering of the previous step.
- 4) Define the *optimal commodity* $c_n^*(t)$ as the commodity $c \in \{1, \dots, N\}$ that maximizes (breaking ties arbitrarily):

$$\sum_{b=1}^{|\mathcal{K}_n(t)|} W_{n, k(n, c, t, b)}^{(c)}(t) \phi_{n, k(n, c, t, b)}^{(c)}(t)$$

where $|\mathcal{K}_n(t)|$ denotes the number of nodes in the set $\mathcal{K}_n(t)$. Define $W_n^*(t)$ as the resulting maximum value:

$$W_n^*(t) = \sum_{b=1}^{|\mathcal{K}_n(t)|} W_{n,k(n,c_n^*,t,b)}^{(c_n^*)}(t) \phi_{n,k(n,c_n^*,t,b)}^{(c_n^*)}(t)$$

- 5) If $W_n^*(t) - Vh_n(P_{tran}) > 0$, node n transmits a packet of commodity $c_n^*(t)$. Else, node n remains idle for slot t .
- 6) After receiving ACK/NACK feedback about the successful recipients of the transmission, node n shifts responsibility of packet forwarding to the successful receiver k with the largest positive differential backlog $W_{nk}^{(c_n^*(t))}(t)$. If no successful receivers have positive differential backlog, node n retains responsibility of the packet.

The above algorithm is fully distributed, in that each node only requires queue backlog and link probability values for each of its neighboring nodes (i.e., each node within $\mathcal{K}_n(t)$). The queue backlogs can be passed during the control information phase of the timeslot, or can be based on backlog updates received in the headers of previous packets. We note that, as in the Dynamic Routing and Power Control (DRPC) policy of [15] [16], the algorithm can be implemented without loss of throughput optimality by using *out of date backlog information*, provided that some regularity conditions hold (see Chapter 4.3.6 of [15]). The link error probabilities can be obtained based on control information exchange at the beginning of the timeslot (such as a pilot signal and a corresponding SINR measurement, as in [16]), or can be estimated based on previous ACK/NACK history. The above algorithm considers the general case where link error events can be correlated. However, computation of the $\phi_{nk}^{(c)}(t)$ probabilities can be greatly simplified under the assumption that error events are independent over each link. In this case, $\phi_{nk}^{(c)}(t)$ is obtained from a simple multiplication of the appropriate success or error probabilities of the corresponding links.

A. Algorithm Performance

To facilitate mathematical analysis, we assume the network topology state $S(t)$ is i.i.d. over timeslots.¹ Note that this also includes the case when the topology state does not change over time. Define the following constant μ_{max}^{in} to be the largest number of endogenous packet arrivals that any single node can receive during a timeslot. Further, define A_{max}^2 as an upper bound on the second moment of the total exogenous arrivals to any node during a timeslot, so that:

$$\max_n \mathbb{E} \left\{ \left(\sum_{c=1}^N A_n^{(c)}(t) \right)^2 \right\} \leq A_{max}^2$$

We assume the input rate matrix is interior to the capacity region Λ (so that stability is possible), and define ϵ_{max} as the largest scalar such that $(\lambda_n^{(c)} + \epsilon_{max} 1_n^{(c)}) \in \Lambda$.

Theorem 2: (Algorithm Performance) If topology state variations $S(t)$ are i.i.d. over timeslots, and if the input rate matrix is strictly interior to the capacity region Λ , then the

¹The same algorithm can be shown to be throughput optimal for non-i.i.d. topology state variations using a similar T -slot Lyapunov drift argument, see [16][15] for such an analysis for a related algorithm.

DIVBAR algorithm stabilizes all queues of the system (and hence provides maximum throughput). Furthermore, average network congestion and average power cost satisfies:

$$\limsup_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \sum_{n,c} \mathbb{E} \left\{ U_n^{(c)}(\tau) \right\} \leq \frac{NB + Vh_{max}}{\epsilon_{max}}$$

$$\limsup_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \sum_n \mathbb{E} \left\{ h_n(\mu_n(\tau) P_{tran}) \right\} \leq \bar{h}^* + NB/V$$

where $h_{max} \triangleq \sum_n h_n(P_{tran})$, and where B is defined:

$$B \triangleq \frac{(\mu_{max}^{in} + A_{max})^2 + 1}{2} \quad (10)$$

Note that choosing the control parameter V to be zero leads to the best congestion bound but does not lead to any power efficiency guarantees. The parameter V can be increased to drive average power cost arbitrarily close to the minimum cost \bar{h}^* required for network stability, with a corresponding linear increase in average network congestion (and hence, by Little's Theorem, average delay).

B. Channel Blind Packet Transmission

In the special case when power optimization is neglected (so that $V = 0$) and there is a single destination for all packets, the DIVBAR algorithm can be significantly simplified to allow for *blind packet transmissions*. Specifically, because there is just a single commodity, the steps (1)-(5) can be avoided and the algorithm reduces to having node n transmit a packet whenever possible (i.e., whenever $\chi_n(t) = 1$). It then receives ACK/NACK feedback from the various receivers, and chooses the receiver k with the largest positive differential backlog $U_n(t) - U_k(t)$, breaking ties arbitrarily and retaining the packet if no receiver has a positive differential backlog. Note that the backlog of each receiver can simply be included in the ACK/NACK signal. The algorithm thus achieves throughput optimality *without requiring channel probability information*. This is a remarkable property, and enables perfect throughput optimality to be achieved even when channel probabilities are rapidly changing due to dramatic node mobility. No effort is needed to estimate error rates, or to track them if they vary with time.

V. PERFORMANCE ANALYSIS

Here we prove Theorem 2. The proof uses the following result from [15] [17] [18] concerning performance optimal Lyapunov scheduling, which is a simple but important extension of classical Lyapunov stability results of [7]-[16]. Let $\mathbf{U}(t) = (U_n^{(c)}(t))$ represent the matrix of queue backlog values, and assume these backlogs evolve according to a given probability law and are affected by a control process $\mathbf{P}(t) = (P_1(t), \dots, P_N(t))$. Let $h(\mathbf{P})$ be any non-negative function of \mathbf{P} , and let h^* represent a target value for the time average of $h(\mathbf{P}(t))$. Let $L(\mathbf{U}) = \frac{1}{2} \sum_{n,c} (U_n^{(c)})^2$ represent a quadratic Lyapunov function, and define the *one step Lyapunov drift* $\Delta(\mathbf{U}(t))$ as follows:

$$\Delta(\mathbf{U}(t)) \triangleq \mathbb{E} \{ L(\mathbf{U}(t+1)) - L(\mathbf{U}(t)) \mid \mathbf{U}(t) \}$$

Theorem 3: (Lyapunov Optimization [15] [17][18]) If there exist positive constants B, V, ϵ such that for all timeslots t and for all queue backlogs $\mathbf{U}(t)$, the Lyapunov drift satisfies:

$$\Delta(\mathbf{U}(t)) + V\mathbb{E}\{h(\mathbf{P}(t)) | \mathbf{U}(t)\} \leq B - \epsilon \sum_{n,c} U_n^{(c)}(t) + Vh^*$$

then all queues are stable, and time average congestion and network cost satisfies:

$$\begin{aligned} \overline{\sum_{n,c} U_n^{(c)}} &\triangleq \limsup_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \sum_{n,c} \mathbb{E}\{U_n^{(c)}(\tau)\} \leq \frac{B + Vh^*}{\epsilon} \\ \bar{h} &\triangleq \limsup_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\{h(\mathbf{P}(\tau))\} \leq h^* + B/V \end{aligned}$$

The above theorem suggests the strategy of minimizing the metric $\Delta(\mathbf{U}(t)) + V\mathbb{E}\{h(\mathbf{P}(t)) | \mathbf{U}(t)\}$ every timeslot t , which is the motivation behind DIVBAR.

A. Proof of the DIVBAR Performance Theorem (Theorem 2)

The conditional Lyapunov drift can be computed from the queue backlog expression (3) according to standard drift techniques (see [7][15][16]), and is given by:

$$\begin{aligned} \Delta(\mathbf{U}(t)) &\leq NB \\ &- \sum_{n,c} U_n^{(c)}(t) \mathbb{E}\left\{ \sum_k \beta_{nk}^{(c)}(t) - \sum_a \beta_{an}^{(c)}(t) - \lambda_n^{(c)} \mid \mathbf{U}(t) \right\} \end{aligned}$$

where B is defined in (10). Adding the cost metric to both sides (where $\mathbf{P}(t) = P_{tran} \cdot (\mu_1(t), \dots, \mu_N(t))$), we have:

$$\begin{aligned} \Delta(\mathbf{U}(t)) + V\mathbb{E}\{h(\mathbf{P}(t)) | \mathbf{U}(t)\} &\leq \\ NB + V\mathbb{E}\{h(\mathbf{P}(t)) | \mathbf{U}(t)\} &- \sum_{n,c} U_n^{(c)}(t) \mathbb{E}\left\{ \sum_k \beta_{nk}^{(c)}(t) - \sum_a \beta_{an}^{(c)}(t) - \lambda_n^{(c)} \mid \mathbf{U}(t) \right\} \end{aligned} \quad (11)$$

The DIVBAR algorithm is designed to choose control actions that greedily minimize the right hand side of the above inequality over all possible choices of the control variables $\mu_n(t)$, $\mu_n^{(c)}(t)$, and $\beta_{nk}^{(c)}(t)$ that satisfy the constraints (1) and (2). This can be seen by switching the sums to note that:

$$\begin{aligned} \sum_{n,c} U_n^{(c)}(t) \mathbb{E}\left\{ \sum_k \beta_{nk}^{(c)}(t) - \sum_a \beta_{an}^{(c)}(t) \mid \mathbf{U}(t) \right\} &= \\ \sum_{ab} \sum_c \mathbb{E}\left\{ \beta_{ab}^{(c)}(t) \mid \mathbf{U}(t) \right\} \left[U_a^{(c)}(t) - U_b^{(c)}(t) \right] \end{aligned}$$

which reveals the differential backlog metric (details omitted for brevity). It follows that the right hand side of (11) under the DIVBAR algorithm is less than or equal to the corresponding expression when the control variables are replaced with any others, and in particular those of the stationary randomized algorithm from (7) and (8) of Corollary 1, and so:

$$\begin{aligned} \Delta(\mathbf{U}(t)) + V\mathbb{E}\{h(\mathbf{P}(t)) | \mathbf{U}(t)\} &\leq \\ NB + V\Phi(\boldsymbol{\lambda} + \boldsymbol{\epsilon}) - \sum_{n,c} U_n^{(c)}(t)\epsilon \end{aligned}$$

The above inequality is in the exact form for application of the Lyapunov Optimization Theorem (Theorem 3), and we

thus have (noting that $\Phi(\boldsymbol{\lambda} + \boldsymbol{\epsilon}) \leq h_{max}$):

$$\sum_{n,c} \overline{U_n^{(c)}} \leq (NB + Vh_{max})/\epsilon \quad (12)$$

$$\bar{h} \leq \Phi(\boldsymbol{\lambda} + \boldsymbol{\epsilon}) + NB/V \quad (13)$$

The above performance bounds hold for any value $\epsilon > 0$ such that $(\lambda_n^{(c)} + \epsilon \mathbf{1}_n^{(c)}) \in \Lambda$, and hence the bounds can be optimized separately over all such ϵ . Letting $\epsilon \rightarrow \epsilon_{max}$ in (12) yields the congestion bound of Theorem 2, and letting $\epsilon \rightarrow 0$ in (13) yields the power cost bound of Theorem 2. \square

VI. VARIABLE RATE AND POWER CONTROL

Consider now a system with variable rate and power control options, so that every timeslot the transmission rates $\boldsymbol{\mu}(t) = (\mu_1(t), \dots, \mu_N(t))$ can be chosen such that $\mu_n(t) \in \{0, 1, \dots, \mu_{max}^{out}\}$ for all t (for some pre-specified integer μ_{max}^{out}), and transmission power to support these rates is chosen according a power vector $\mathbf{P}(t) = (P_1(t), \dots, P_N(t))$, where $0 \leq P_n(t) \leq P_{peak}$ for all t and all n (for some peak transmission power P_{peak}). Note that the $\mu_n(t)$ variable is still integer valued, but there is no longer any multiple access process $\chi_n(t)$ that places further restrictions on $\mu_n(t)$. Define $I(t) \triangleq (\boldsymbol{\mu}(t); \mathbf{P}(t))$ as the collective transmission control decisions of all network nodes during slot t , and define \mathcal{I} as the set of all possible options for $I(t)$. We assume that error probabilities are functions of $I(t)$ and the current topology state $S(t)$, so that:

$$q_{n,\Omega_n}(t) = \hat{q}_{n,\Omega_n}(I(t), S(t))$$

If m packets are transmitted by node n , then each of them is assumed to have the same $q_{n,\Omega_n}(t)$ probability. Correlations in the error events of different packets within the batch of m are arbitrary and do not affect capacity or optimal control decisions.

The control objective of stabilizing the network and minimizing \bar{h} is the same as before. Using similar reasoning, it can again be shown that it is possible to restrict to algorithms that do not allow redundant forwarding, without loss of optimality. A similar Lyapunov argument then leads to the following optimal policy:

- 1) Compute $W_{nk}^{(c)}(t) = \max[U_n^{(c)}(t) - U_k^{(c)}(t), 0]$ as before. For each node n and each commodity c , we again rank order the receivers $k \in \mathcal{K}_n(t)$ with priority given by the largest values of $W_{nk}^{(c)}(t)$, and define $k(n, c, t, b)$ as before. We define $\hat{\phi}_{nk}^{(c)}(I(t), S(t))$ as the probability that a packet transmission from node n during slot t is correctly received by node k , but not received by any other nodes $\tilde{k} \in \mathcal{K}_n(t)$ that are ranked with higher priority than node k according to the commodity c ordering.
- 2) Define:

$$\begin{aligned} G_{n,c,t,b}(I(t), S(t)) &\triangleq \\ W_{n,k(n,c,t,b)}^{(c)}(t) &\hat{\phi}_{n,k(n,c,t,b)}(I(t), S(t)) \end{aligned}$$

Choose a network-collaborative control action $I^*(t) = (\boldsymbol{\mu}^*(t), \mathbf{P}^*(t)) \in \mathcal{I}$ and a collection of optimal commodities $c_n^*(t) \in \{1, \dots, N\}$ (for all nodes n) that jointly

maximizes the metric:

$$\sum_n \left[\left(\sum_{b=1}^{|\mathcal{K}_n(t)|} G_{n,c_n^*(t),t,b}(I^*(t), S(t)) \right) - Vh_n(P_n^*(t)) \right]$$

- 3) If $\sum_{b=1}^{|\mathcal{K}_n(t)|} [G_{n,c_n^*(t),t,b}(I^*(t), S(t))] > Vh_n(P_n^*(t))$, node n transmits $\mu_n^*(t)$ commodity $c_n^*(t)$ packets (using idle fill if there are not enough such packets).
- 4) After receiving ACK/NACK feedback from each receiver about each of the $\mu_n^*(t)$ transmitted packets, node n shifts responsibility of each packet to the successful receiver with the largest positive differential backlog $W_{nk}^{(c_n^*(t))}(t)$. If no receivers of a given packet have positive differential backlog, node n retains responsibility of the packet.

Choosing the appropriate control action $I(t) = (\mu(t); P(t))$ effectively optimizes over all multiple access decisions, but yields an optimization problem in step 2 that can be quite difficult to solve and may require full centralized coordination. However, distributed implementation is possible if all nodes transmit with orthogonal signals, and constant factor throughput optimality results can be achieved in a decentralized manner for some networks models (such as the node exclusive spectrum sharing model of [29]), if power optimization is neglected and simple random access methods are employed. We further note that if simple random access methods are used as in [16][15], and if transmissions are independent of queue backlog, the random transmissions themselves can be viewed as part of the channel state, and the algorithm thus achieves efficient performance with respect to this (sub-optimal) random access.

VII. DISCUSSION OF EXTENSIONS

We note that delay can be improved by modifying the differential backlog $W_{nk}^{(c)}(t)$ by adding an estimated hop count differential to the destination, as in the EDRPC policy of [16] [15]. Optimization of general utility and fairness metrics can also be achieved in cases when the input rate matrix is either inside or outside of the capacity region Λ by using the simple and optimal flow control techniques of [15] [18] together with DIVBAR. Finally, we note that Lyapunov drift theorems hold if drift expressions are off by an additive constant, so that queue backlog estimates can be used in replacement of actual queue backlogs [15]. This does not effect throughput optimality, but may increase delay by a constant proportional to the difference between the true and estimated backlog values. This enables DIVBAR to be implemented with delayed feedback, provided the feedback delay is bounded. A queueing implementation of this would provide separate storage for data that has been correctly received but has not yet been acknowledged by a final instruction message from the transmitter.

REFERENCES

- [1] M. Stojanovic, J. A. Catipovic, and J. G. Proakis. Phase-coherent digital communications for underwater acoustic channels. *IEEE Journal of Oceanic Engineering*, vol. 19, no. 1, Jan. 1994.
- [2] D. B. Kilfoyle, J. C. Preisig, and A. B. Baggeroer. Spatial modulation experiments in the underwater acoustic channel. *IEEE Journal of Oceanic Engineering*, vol. 30, no. 2, April 2005.
- [3] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides. Algorithms for energy-efficient multicasting in ad hoc wireless networks. *Proc. IEEE Military Communications Conference*, pp. 1414-1418, 1999.
- [4] A. Srinivas and E. Modiano. Minimum energy disjoint path routing in wireless ad-hoc networks. *IEEE Proc. of Mobicom*, September 2003.
- [5] M. Zorzi and R. Rao. Geographic random forwarding (geraf) for ad hoc and sensor networks: Multihop performance. *IEEE Trans. on Mobile Computing*, Vol. 2, no. 4, Oct.-Dec. 2003.
- [6] S. Biswas and R. Morris. Exor: Opportunistic multi-hop routing for wireless networks. *Proc. of Sigcomm*, 2005.
- [7] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control*, Vol. 37, no. 12, Dec. 1992.
- [8] L. Tassiulas and A. Ephremides. Dynamic server allocation to parallel queues with randomly varying connectivity. *IEEE Trans. on Inform. Theory*, vol. 39, pp. 466-478, March 1993.
- [9] P.R. Kumar and S.P. Meyn. Stability of queueing networks and scheduling policies. *IEEE Trans. on Automatic Control*, Feb. 1995.
- [10] N. McKeown, V. Anantharam, and J. Walrand. Achieving 100% throughput in an input-queued switch. *Proc. INFOCOM*, 1996.
- [11] N. Kahale and P. E. Wright. Dynamic global packet routing in wireless networks. *IEEE Proceedings of INFOCOM*, 1997.
- [12] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, and P. Whiting. Providing quality of service over a shared wireless link. *IEEE Communications Magazine*, 2001.
- [13] E. Leonardi, M. Melia, F. Neri, and M. Ajmone Marson. Bounds on average delays and queue size averages and variances in input-queued cell-based switches. *Proc. INFOCOM*, 2001.
- [14] M. J. Neely, E. Modiano, and C. E. Rohrs. Power allocation and routing in multi-beam satellites with time varying channels. *IEEE Transactions on Networking*, Feb. 2003.
- [15] M. J. Neely. *Dynamic Power Allocation and Routing for Satellite and Wireless Networks with Time Varying Channels*. PhD thesis, Massachusetts Institute of Technology, LIDS, 2003.
- [16] M. J. Neely, E. Modiano, and C. E Rohrs. Dynamic power allocation and routing for time varying wireless networks. *IEEE Journal on Selected Areas in Communications*, January 2005.
- [17] M. J. Neely. Energy optimal control for time varying wireless networks. *Proceedings of IEEE INFOCOM*, March 2005.
- [18] M. J. Neely, E. Modiano, and C. Li. Fairness and optimal stochastic control for heterogeneous networks. *Proceedings of IEEE INFOCOM*, March 2005.
- [19] R. Cruz and A. Santhanam. Optimal routing, link scheduling, and power control in multi-hop wireless networks. *IEEE Proceedings of INFOCOM*, April 2003.
- [20] X. Liu, E. K. P. Chong, and N. B. Shroff. A framework for opportunistic scheduling in wireless networks. *Computer Networks*, vol. 41, no. 4, pp. 451-474, March 2003.
- [21] A. Stolyar. Maximizing queueing network utility subject to stability: Greedy primal-dual algorithm. *Queueing Systems*, 2005.
- [22] J. W. Lee, R. R. Mazumdar, and N. B. Shroff. Opportunistic power scheduling for dynamic multiserver wireless systems. *to appear in IEEE Trans. on Wireless Systems*, 2005.
- [23] C. Swannack, E. Uysal-Biyikoglu, and G. Wornell. Low complexity multiuser scheduling for maximizing throughput in the mimo broadcast channel. *Proc. of 42nd Allerton Conference on Communication, Control, and Computing*, September 2004.
- [24] M. Kobayashi. *On the Use of Multiple Antennas for the Downlink of Wireless Systems*. PhD thesis, Ecole Nationale Supérieure des Telecommunications, Paris, 2005.
- [25] M. Kobayashi, G. Caire, and D. Gesbert. Impact of multiple transmit antennas in a queued SDMA/TDMA downlink. In *Proceedings of 6th IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, June 2005.
- [26] T. Ho and H. Viswanathan. Dynamic algorithms for multicast with intra-session network coding. In *Proceedings of 43rd Allerton Conference on Communication, Control and Computing*, September 2005.
- [27] E. Yeh and R. Berry. Throughput optimal control of cooperative relay networks. *Proc. of International Symposium on Information Theory, Adelaide, Australia*, pp. 1206-1210, September 2005.
- [28] A. Khandani, J. Abounadi, E. Modiano, and L. Zheng. Cooperative routing in wireless networks. *Allerton Conference on Communications, Control, and Computing*, pp. 1270-1279, Oct. 2003.
- [29] X. Lin and N. B. Shroff. The impact of imperfect scheduling on cross-layer rate control in wireless networks. *IEEE Proc. of INFOCOM*, 2005.