Optimal Backpressure Routing for Wireless Networks with Multi-Receiver Diversity

Michael J. Neely , Rahul Urgaonkar

***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**

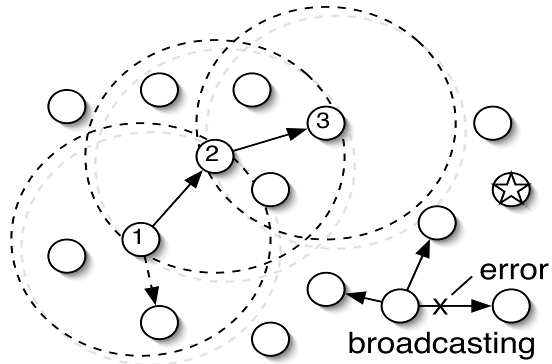


Fig. 1. A multi-hop network with channel errors and multi-receiver diversity. In this example there is a single destination indicated by the star node. Note that a "closest-to-the-destination" heuristic might result in data being routed from node 1 to 2 to 3, resulting in a deadlock.

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 [2] [3]. 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 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 [4] for the design of wireless multicast algorithms, and in [5] for the design of minimum energy disjoint paths. Our model and problem formulation is closest to the work by Zorzi and Rao in [6], and more recently by Biswas and Morris in [7] and Baccelli et. al. in [8], 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

This work was presented in part at the Conference on Information Sciences and Systems (CISS), Princeton University, March 2006.

Michael J. Neely and Rahul Urgaonkar are with the Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089. (website: http://www-rcf.usc.edu/∼mjneely)

This work is supported by one or more of the following: the National Science Foundation grant OCE 0520324, the DARPA IT-MANET program.

work in [6] considers the 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 [7] 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. Work in [8] develops analytical properties of related schemes. However, these algorithms are 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 [9], 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 [9]-[20], including our own work in [17]-[20] 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 [21]-[24]. The work in [9]-[24] 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 [25], and related MIMO results are developed for channels with errors in [26] [27]. Recent work in [28] considers backpressure techniques in combination with network coding, and work in [29] 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 [30]. Heuristic algorithms that combine network coding and multi-receiver diversity are developed in [31]. 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 discuss channel-aware and channel-blind implementations. 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. Simulations are presented in Section VIII.

II. THE BASIC NETWORK MODEL

We consider a timeslotted system with slots normalized to integral units $t \in \{0, 1, 2, \ldots\}$. 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 $n, k \in \{1, \ldots, N\}$). 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 $S(t)$. That is, we have:

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

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 $n \in \{1, \ldots, N\}$. 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 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.

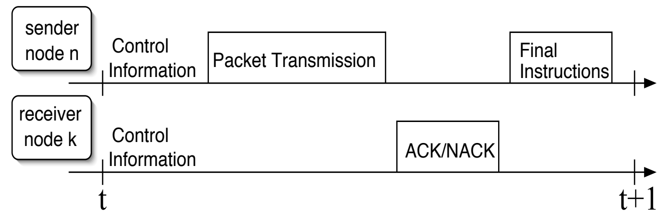


Fig. 2. A timing diagram illustrating the events within a single timeslot.

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 instructions) 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 $\boldsymbol{P} = (P_1, \ldots, P_N)$, we define the separable cost function $h(\boldsymbol{P}) = h_1(P_1) + \ldots + h_N(P_N)$, 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 $\boldsymbol{P}(t) \triangleq P_{tran} \cdot (\mu_1(t), \ldots, \mu_N(t))$, and the time average power cost $\overline{h}$ is defined:

$$\overline{h} \triangleq \lim_{t\rightarrow\infty} \frac{1}{t}\sum_{\tau=0}^{t-1}\mathbb{E}\left\{h(\boldsymbol{P}(\tau))\right\}$$

Note that choosing $h(\boldsymbol{P}) = \sum_{n=1}^N P_n$ 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 queueing 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 that reveals the $H_{nk}(t)$ variables, 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 can be shifted from node n to node k during slot t. The $\beta_{nk}^{(c)}(t)$ variables must satisfy the following constraints:

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

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 queueing 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. We emphasize that the $\beta_{nk}^{(c)}(t)$ values are determined *after* transmission decisions $\mu_n^{(c)}(t)$ have been made and ACK/NACK feedback (in the form of the random $H_{nk}(t)$ variables) have been received.

III. Network Capacity and Minimum Power

Here we characterize the optimal throughput and average power cost operating points. We define a control algorithm to be *rate stable* if the long term average rate of delivering packets to their appropriate destinations is equal to the exogenous input rate.[1] 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 $\{1, \ldots, N\} - \{n\}$. 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 $S(t) = s$.

[1] We shall use the simplified term *stable* throughout this paper when referring to rate stability.

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:

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

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

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

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

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

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

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

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

$$\overline{h}^* = \textstyle\sum_{s \in \mathcal{S}} \pi_s \left[\sum_{n=1}^N \sum_{c=1}^N \alpha_n^{(c)}(s) h_n(P_{tran}) \right] \tag{7}$$

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

Proof: See Appendix A. □

Note that the $\theta_{nk}^{(c)}(\Omega_n)$ probabilities are defined for each link (n, k), each commodity c, and each of the 2^{N-1} subsets Ω_n. In particular, the above theorem describes an optimization problem with geometric complexity. The theorem is similar in spirit to the capacity theorem of [18] [17], 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 notation of the theorem can be simplified according to the following corollary.

For each input rate matrix $\boldsymbol{\lambda} = (\lambda_n^{(c)}) \in \Lambda$, we define $\Phi(\boldsymbol{\lambda})$ as the minimum power cost $\overline{h}^*$ required to stabilize the system. It is not difficult to show that $\Phi(\boldsymbol{\lambda})$ is continuous and convex in the rate vector $\boldsymbol{\lambda}$ [32]. 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 $\boldsymbol{P}^*(t)$, $\mu_n^*(t)$, $\mu_n^{*(c)}(t)$ 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:

$$\sum_a \mathbb{E}\left\{\beta_{an}^{*(c)}(t)\right\} + \lambda_n^{(c)} + \epsilon \leq \sum_b \mathbb{E}\left\{\beta_{nb}^{*(c)}(t)\right\} \ \forall n \neq c \tag{8}$$

$$\mathbb{E}\left\{h(\boldsymbol{P}^*(t))\right\} = \Phi(\boldsymbol{\lambda} + \boldsymbol{\epsilon}) \tag{9}$$

where $\boldsymbol{\epsilon} = (\epsilon 1_n^{(c)})$ and $\boldsymbol{P}(t) = P_{tran} \cdot (\mu_1(t), \ldots, \mu_N(t))$. The expectations in (8) and (9) 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 $\overline{h}^*$. 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 $\overline{h}^*$ (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 [18] [19] [17] (see Appendix A). 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:

$$W_{nk}^{(c)}(t) = \max[U_n^{(c)}(t) - U_k^{(c)}(t), 0] \tag{10}$$

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 bth largest weight $W_{nk}^{(c)}(t)$ for commodity c. Thus, by definition we have:
$$W_{n,k(n,c,t,1)}^{(c)}(t) \geq W_{n,k(n,c,t,2)}^{(c)}(t) \geq W_{n,k(n,c,t,3)}^{(c)}(t) \ldots$$
3) Define $\phi_{nk}^{(c)}(t)$ as the probability that a packet transmission from node n is correctly recieved by node k, but is *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 rank ordering of the previous step.
4) Define the *optimal commodity* $c_n^*(t)$ as the commodity $c \in \{1, \ldots, N\}$ that maximizes (breaking ties arbitrarily):
$$\sum_{b=1}^{|\mathcal{K}_n(t)|} W_{n,k(n,c,t,b)}^{(c)}(t)\phi_{n,k(n,c,t,b)}^{(c)}(t) \qquad (11)$$
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 [17] [18], the algorithm can be implemented without loss of throughput optimality by using *out of date backlog information*, provided that some regularity conditions hold (see also [33]). 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 [18]), 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. Specifically, independent link errors would yield $\phi_{nk}^{(c)}(t)$ values given by:
$$\phi_{nk}^{(c)}(t) = q_{nk}(t)\Pi_{b=1}^{rank(n,k,c,t)-1}[1 - q_{n,k(n,c,t,b)}(t)]$$
where $rank(n,k,c,t)$ is defined as the rank order of receiver $k \in \mathcal{K}_n(t)$ according to the $W_{nk}^{(c)}(t)$ weights.

A. Intuition on the Backpressure Metric

The above algorithm uses the backpressure concept of [9] [18] to route data in the direction of maximum differential backlog. To understand how the $\phi_{nk}^{(c)}(t)$ probabilities arise in the metric (11), suppose that a particular node n transmits a packet of commodity c and then receives ACK/NACK feedback. Step 6 of DIVBAR implies that node n must find the successful receiver $k \in \mathcal{K}_n(t)$ that maximizes the differential backlog metric $W_{nk}^{(c)}(t)$. Thus, each node n must compute:
$$\max_{k \in \mathcal{K}_n(t)} \left\{H_{nk}(t)W_{nk}^{(c)}(t)\right\}$$
This value is a backpressure index that measures the effectiveness of transmitting a packet of commodity c, and can be written according to the rank ordering:
$$\max_{k \in \mathcal{K}_n(t)} \left\{H_{nk}(t)W_{nk}^{(c)}(t)\right\} = \sum_{b=1}^{|\mathcal{K}_n(t)|} W_{n,k(n,c,t,b)}^{(c)}(t)1_{n,k(n,c,t,b)}^{(c)}(t)$$
where $1_{n,k}^{(c)}(t)$ is an indicator function that is 1 if and only if $H_{nk}(t) = 1$ and $H_{n,m}(t) = 0$ for all receivers $m \in \mathcal{K}_n(t)$ with a rank ordering that is higher than the rank of receiver k. This indicator function can be one for at most one term in the sum on the right hand side, and is 1 only for the term with the largest $H_{nk}(t)W_{nk}^{(c)}(t)$ value. This value cannot be known before transmission, as it depends on the random success/failure events on each of the outgoing links. Its conditional expectation given the current queue backlog is:
$$\mathbb{E}\left\{\max_{k \in \mathcal{K}_n(t)} \left\{H_{nk}(t)W_{nk}^{(c)}(t)\right\} \mid \boldsymbol{U}(t)\right\} = \sum_{b=1}^{|\mathcal{K}_n(t)|} W_{n,k(n,c,t,b)}^{(c)}(t)\phi_{n,k(n,c,t,b)}^{(c)}(t) \leq W_n^*(t) \quad (12)$$
where $\boldsymbol{U}(t) = (U_n^{(c)}(t))$ represents the matrix of current queue backlogs during slot t. Step 4 of the DIVBAR algorithm thus selects the commodity $c_n^*(t)$ with the largest expected backpressure index. The inequality (12) shall also be important in the analysis of DIVBAR presented in Section V.

B. Algorithm Performance

To facilitate mathematical analysis, we assume the network topology state $S(t)$ is i.i.d. over timeslots.[2] Note that this also includes the case when the topology state does not change over time. Define the 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:
$$\max_n \mathbb{E}\left\{\left(\sum_{c=1}^{N} A_n^{(c)}(t)\right)^2\right\} \leq A_{max}^2$$

[2]The 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 [18][33] for such an analysis for a related algorithm.

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

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

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

where $\overline{h}^*$ is the minimum power cost (defined in (7)), $h_{max} \triangleq \sum_n h_n(P_{tran})$, and where B is defined:

$$B \triangleq \frac{(\mu_{max}^{in} + A_{max})^2 + 1}{2} \quad (13)$$

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 $\overline{h}^*$ required for network stability, with a corresponding linear increase in average network congestion (and hence, by Little's Theorem, average delay). We prove Theorem 2 in Section V.

C. 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 2, 3, 4, and 5 of DIVBAR can be avoided. The algorithm thus 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.

Note that this single commodity scenario also applies in cases when the data can be delivered to any one of a set of sink nodes, as these sinks can be viewed collectively as a single virtual destination. This is important, for example, in a sensor network with multiple data recovery points, or in a wireless network with multiple base stations that provide access to a larger wireline system.

If there are K commodities (where $K \geq 1$), the decision of which commodity to transmit can be trivialized by the (sub-optimal) strategy of randomly choosing a commodity every transmission opportunity, independently and uniformly over all commodities $c \in \{1, \ldots, K\}$. When $V = 0$, this random commodity selection can be implemented without knowledge of channel probabilities, and stabilizes the network whenever input rates are within Λ/K (the capacity region that is reduced by a factor of K). This fact is proven in Section VI-A for a generalized version of DIVBAR that supports multiple rate and power options.

V. Performance Analysis

Here we prove Theorem 2. The proof uses the following result from [17] [19] [20] concerning performance optimal Lyapunov scheduling, which is a simple but important extension of classical Lyapunov stability results of [9]-[18]. Let $\boldsymbol{U}(t) = (U_n^{(c)}(t))$ 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 $\boldsymbol{P}(t) = (P_1(t), \ldots, P_N(t))$. Let $h(\boldsymbol{P})$ be any non-negative function of $\boldsymbol{P}$, and let h^* represent a target value for the time average of $h(\boldsymbol{P}(t))$. Let $L(\boldsymbol{U}) = \frac{1}{2}\sum_{n,c}(U_n^{(c)})^2$ represent a quadratic Lyapunov function, and define the *one step Lyapunov drift* $\Delta(\boldsymbol{U}(t))$ as follows:

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

Theorem 3: (Lyapunov Optimization [17] [19][20]) If there exist constants $B > 0$, $\epsilon > 0$, $V \geq 0$ such that for all timeslots t and for all queue backlogs $\boldsymbol{U}(t)$, the Lyapunov drift satisfies:

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

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

$$\overline{\sum_{n,c}U_n^{(c)}} \triangleq \limsup_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1}\sum_{n,c}\mathbb{E}\left\{U_n^{(c)}(\tau)\right\} \leq \frac{B + Vh^*}{\epsilon}$$

$$\overline{h} \triangleq \limsup_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1}\mathbb{E}\left\{h(\boldsymbol{P}(\tau))\right\} \leq h^* + B/V$$

The above theorem suggests the strategy of minimizing the metric $\Delta(\boldsymbol{U}(t)) + V\mathbb{E}\left\{h(\boldsymbol{P}(t)) \mid \boldsymbol{U}(t)\right\}$ 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 [33]), and is given by:

$$\Delta(\boldsymbol{U}(t)) \leq NB - \sum_{n,c}U_n^{(c)}(t)\mathbb{E}\left\{\sum_b \beta_{nb}^{(c)}(t) - \sum_a \beta_{an}^{(c)}(t) - \lambda_n^{(c)} \mid \boldsymbol{U}(t)\right\}$$

where B is defined in (13). Adding the cost metric to both sides (where $\boldsymbol{P}(t) = P_{tran}\cdot(\mu_1(t), \ldots, \mu_N(t))$), we have:

$$\Delta(\boldsymbol{U}(t)) + V\mathbb{E}\left\{h(\boldsymbol{P}(t)) \mid \boldsymbol{U}(t)\right\} \leq NB + V\mathbb{E}\left\{h(\boldsymbol{P}(t)) \mid \boldsymbol{U}(t)\right\} - \sum_{n,c}U_n^{(c)}(t)\mathbb{E}\left\{\sum_b \beta_{nb}^{(c)}(t) - \sum_a \beta_{an}^{(c)}(t) - \lambda_n^{(c)} \mid \boldsymbol{U}(t)\right\} \quad (14)$$

The DIVBAR algorithm is designed to choose control actions that greedily minimize the right hand side of the above inequality. Specifically, we have the following important lemma:

Lemma 1: The algorithm DIVBAR chooses control variables $\boldsymbol{P}(t)$, $\mu_n(t)$, $\mu_n^{(c)}(t)$, and $\beta_{nk}^{(c)}(t)$ that minimize the right hand side of (14) over all feasible control choices that satisfy the constraints (1) and (2).

Lemma 1 is formally proven in Appendix B. The lemma implies that the right hand side of (14) under the DIVBAR algorithm is less than or equal to the corresponding expression when the control decisions $\boldsymbol{P}(t)$, $\mu_n(t)$, $\mu_n^{(c)}(t)$, and $\beta_{nk}^{(c)}(t)$ are replaced by the decision variables $\boldsymbol{P}^*(t)$, $\mu_n^*(t)$, $\mu_n^{*(c)}(t)$, and $\beta_{nk}^{*(c)}(t)$ associated with the stationary randomized algorithm of Corollary 1. In particular, recall from Corollary 1 that for any value $\epsilon > 0$ that satisfies $(\lambda_n^{(c)} + \epsilon 1_n^{(c)}) \in \Lambda$, there exists a stationary randomized control algorithm that satisfies all control constraints, makes decisions independent of current queue backlog $\boldsymbol{U}(t)$, and yields (compare with (8) and (9)):

$$\mathbb{E}\left\{\sum_b \beta_{nb}^{*(c)}(t) - \sum_a \beta_{an}^{*(c)}(t) - \lambda_n^{(c)} \mid \boldsymbol{U}(t)\right\} \geq \epsilon \quad (15)$$

$$\mathbb{E}\left\{h(\boldsymbol{P}^*(t)) \mid \boldsymbol{U}(t)\right\} = \Phi(\boldsymbol{\lambda} + \boldsymbol{\epsilon}) \quad (16)$$

Plugging (15) and (16) directly into the right hand side of (14) thus preserves the inequality and yields:

$$\Delta(\boldsymbol{U}(t)) + V\mathbb{E}\left\{h(\boldsymbol{P}(t)) \mid \boldsymbol{U}(t)\right\} \leq NB + V\Phi(\boldsymbol{\lambda} + \boldsymbol{\epsilon}) - \epsilon \sum_{n,c} U_n^{(c)}(t)$$

The above inequality is in the exact form for application of the Lyapunov Optimization Theorem (Theorem 3), and we thus have (noting that $\Phi(\boldsymbol{\lambda} + \boldsymbol{\epsilon}) \leq h_{max}$):

$$\overline{\sum_{n,c} U_n^{(c)}} \leq (NB + Vh_{max})/\epsilon \quad (17)$$

$$\overline{h} \leq \Phi(\boldsymbol{\lambda} + \boldsymbol{\epsilon}) + NB/V \quad (18)$$

The above performance bounds hold for any value $\epsilon > 0$ such that $(\lambda_n^{(c)} + \epsilon 1_n^{(c)}) \in \Lambda$, and hence the bounds can be optimized separately over all such ϵ. Letting $\epsilon \to \epsilon_{max}$ in (17) yields the congestion bound of Theorem 2. Letting $\epsilon \to 0$ in (18) and noting that continuity of the $\Phi(\boldsymbol{\lambda})$ function implies $\Phi(\boldsymbol{\lambda}+\boldsymbol{\epsilon}) \to \Phi(\boldsymbol{\lambda}) \triangleq \overline{h}^*$ yields the power cost bound of Theorem 2. □

VI. Variable Rate and Power Control

Consider now a system with variable rate and power control options, so that every timeslot the transmission rates $\boldsymbol{\mu}(t) = (\mu_1(t), \ldots, \mu_N(t))$ can be chosen such that $\mu_n(t) \in \{0, 1, \ldots, \mu_{max}^{out}\}$ for all t (for some pre-specified integer μ_{max}^{out}), and transmission power to support these rates is chosen according a power vector $\boldsymbol{P}(t) = (P_1(t), \ldots, P_N(t))$, 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 $I(t) \triangleq (\boldsymbol{\mu}(t); \boldsymbol{P}(t))$ 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 $\mathcal{I}$ is such that if $(\boldsymbol{\mu}, \boldsymbol{P}) \in \mathcal{I}$, then setting any rate or power entry of $(\boldsymbol{\mu}, \boldsymbol{P})$ to zero yields another vector within $\mathcal{I}$. 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 $\overline{h}$ 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 $\hat{\phi}_{nk}(I(t), S(t))$ 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:[3]

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

Choose a network-collaborative control action $I^*(t) = (\boldsymbol{\mu}^*(t), \boldsymbol{P}^*(t)) \in \mathcal{I}$ and a collection of optimal commodities $c_n^*(t) \in \{1, \ldots, N\}$ (for all nodes n) that jointly maximizes the metric $M(I^*(t), \boldsymbol{c}^*(t), S(t))$, where:

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

3) If $\sum_{b=1}^{|\mathcal{K}_n(t)|}\left[G_{n,c_n^*(t),t,b}(I^*(t), S(t))\right] > Vh_n(P_n^*(t))$, 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 $I(t) = (\boldsymbol{\mu}(t); \boldsymbol{P}(t))$ 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 if the resource allocation

[3]The original paper had a typo here that neglected the $\mu_n(t)$ on the right hand side of the $G_{n,c,t,b}(I(t), S(t))$ definition. That typo is corrected here.

optimization in step 2 is achieved to within a constant factor by some lower complexity scheme. Specifically, for a fixed constant γ such that $0 < \gamma \leq 1$, define $\gamma\Lambda$ as a *γ-scaled version of the capacity region* [17] [34], so that $\boldsymbol{\lambda} \in \gamma\Lambda$ if and only if there exists a vector $\hat{\boldsymbol{\lambda}} \in \Lambda$ such that $\boldsymbol{\lambda} = \gamma\hat{\boldsymbol{\lambda}}$. Suppose now that exogenous arrivals are i.i.d. over timeslots with arrival rate matrix $\boldsymbol{\lambda} = (\lambda_n^{(c)})$ that is interior to $\gamma\Lambda$.

Theorem 4: (Generalized DIVBAR) Suppose that the above generalized DIVBAR algorithm is carried out, with the exception that every timeslot a (potentially sub-optimal) control action $\tilde{I}(t) = (\tilde{\boldsymbol{\mu}}(t), \tilde{\boldsymbol{P}}(t)) \in \mathcal{I}$ and commodities $\tilde{\boldsymbol{c}}(t) = (\tilde{c}_1(t), \ldots, \tilde{c}_n(t))$ are used and satisfy for all t:

$$\mathbb{E}\left\{M(\tilde{I}(t), \tilde{\boldsymbol{c}}(t), S(t)) \mid \boldsymbol{U}(t), S(t)\right\} \geq \gamma M(I^*(t), \boldsymbol{c}^*(t), S(t)) - C$$

where γ and C are constants such that $0 < \gamma \leq 1$ and $C \geq 0$, and $I^*(t)$, $\boldsymbol{c}^*(t)$ are the optimal solutions. The above expectation is with respect to possible randomized choices of $\tilde{I}(t)$ and $\tilde{\boldsymbol{c}}(t)$. If there is a positive value ϵ_{max} such that $(\lambda_n^{(c)} + \epsilon_{max} 1_n^{(c)}) \in \gamma\Lambda$, then:

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

$$\limsup_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1}\mathbb{E}\left\{h(\boldsymbol{P}(\tau))\right\} \leq \gamma\Phi(\boldsymbol{\lambda}/\gamma) + \frac{NB + C}{V}$$

where $\Phi(\boldsymbol{\lambda}/\gamma)$ is the minimum average power required to stabilize a rate matrix $\boldsymbol{\lambda}/\gamma$, and B is defined:

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

Proof: See Appendix C. □

The case $\gamma = 1$, $C = 0$ corresponds to a resource and scheduling action that optimizes the metric $M(\cdot)$ in step 2, and yields stability for any rate matrix in the interior to the capacity region Λ. Below we describe two sub-optimal schemes with reduced complexity that yield performance within a constant factor of optimality.

A. Random Commodity Selection

Suppose there are K commodities in the network, and that $V = 0$ (so that average power optimality is neglected). Rather than carrying out the complex computations required to select the optimal commodity to transmit, every timeslot each node *randomly* chooses a commodity, independently and uniformly over all commodities $c \in \{1, \ldots, K\}$. Let $\tilde{\boldsymbol{c}}(t)$ represent the resulting random commodity selections on slot t. The topology state $S(t)$ is then observed, and a control action $\tilde{I}(t)$ is taken, where $\tilde{I}(t)$ maximizes $M(I(t), \tilde{\boldsymbol{c}}(t), S(t))$ over all $I(t) \in \mathcal{I}$.

To analyze this algorithm, define functions $M_n(\cdot)$:

$$M_n(I(t), c_n(t), S(t)) \triangleq \sum_{b=1}^{|\mathcal{K}_n(t)|} G_{n, c_n(t), t, b}(I(t), S(t))$$

and note that, because $V = 0$, for any $I(t), \boldsymbol{c}(t), S(t)$ we have:

$$M(I(t), \boldsymbol{c}(t), S(t)) = \sum_n M_n(I(t), c_n(t), S(t))$$

Let $I^*(t)$ represent the optimal control action corresponding to the optimal commodities $\boldsymbol{c}^*(t)$. Because $I^*(t) \in \mathcal{I}$, by definition of $\tilde{I}(t)$ we have for all t:

$$M(\tilde{I}(t), \tilde{\boldsymbol{c}}(t), S(t)) \geq M(I^*(t), \tilde{\boldsymbol{c}}(t), S(t)) \tag{19}$$

Taking expectations of (19) with respect to the distribution of the random commodities $\tilde{\boldsymbol{c}}(t)$ yields:

$$\begin{aligned}
&\mathbb{E}\left\{M(\tilde{I}(t), \tilde{\boldsymbol{c}}(t), S(t)) \mid \boldsymbol{U}(t), S(t)\right\} \\
&\quad\geq \mathbb{E}\left\{M(I^*(t), \tilde{\boldsymbol{c}}(t), S(t)) \mid \boldsymbol{U}(t), S(t)\right\} \\
&\quad= \textstyle\sum_n \mathbb{E}\left\{M_n(I^*(t), \tilde{c}_n(t), S(t)) \mid \boldsymbol{U}(t), S(t)\right\} \\
&\quad= \textstyle\sum_n \frac{1}{K}\sum_{k=1}^K M_n(I^*(t), k, S(t)) \qquad (20) \\
&\quad\geq \textstyle\sum_n \frac{1}{K} M_n(I^*(t), c_n^*(t), S(t)) \qquad (21) \\
&\quad= \tfrac{1}{K} M(I^*(t), \boldsymbol{c}^*(t), S(t))
\end{aligned}$$

where (20) follows because each commodity $\tilde{c}_n(t)$ is chosen independently and uniformly over $k \in \{1, \ldots, K\}$, and (21) holds because $M_n(\cdot) \geq 0$ and hence the sum over all commodities is greater than or equal to the single term associated with commodity $c_n^*(t)$. It follows that this random commodity selection algorithm satisfies the condition of Theorem 4 with $\gamma = 1/K$, $C = 0$. Thus, the algorithm stabilizes the network whenever input rates are within a K-reduced factor of capacity.

B. Random Transmitter Selection

Consider a network where a node cannot transmit and receive on the same timeslot, and where all transmissions take place with power P_{tran}. Furthermore, assume a simple *collision model*, where every timeslot each node within reception range of a given transmitter has at most J other nodes that can act as *interferers* of this transmission. The reception probability of a node is zero during any timeslot when it is transmitting or when it is attempting to receive while an interferer is also transmitting. Otherwise, the reception probability is either constant or determined by a topology state process $S(t)$. Similar models are used for networks without multi-reciever diversity in [34] [35] [36]. An important special case is when $J = 0$, where the only constraint is that nodes cannot transmit and receive simultaneously (such as when multi-user detection is possible).

For simplicity, again let $V = 0$ so that optimizing average power cost is neglected. Let $I^*(t) = (\boldsymbol{\mu}^*(t), \boldsymbol{P}^*(t))$ represent the control action that optimizes step 2 of the generalized DIVBAR algorithm. This would require extensive coordination to compute. Consider instead the following sub-optimal algorithm: Every timeslot, each node independently enters enters *transmission mode* with probability q, and enters *reception mode* with probability $1 - q$. Define $\tilde{I}_n(t)$ and $\tilde{c}_n(t)$ as the *greedy control action and commodity choice* at node n, which maximizes:

$$M_n(I(t), c, S(t)) \triangleq \sum_{b=1}^{|\mathcal{K}_n(t)|} W_{n,k(n,c,t,b)}^{(c)}(t)\hat{\phi}_{n,k(n,c,t,b)}(I(t), S(t)) \tag{22}$$

over all commodities $c \in \{1, \ldots, K\}$ and all control actions $I(t) = (\boldsymbol{\mu}(t), \boldsymbol{P}(t)) \in \mathcal{I}$ that satisfy the additional restriction that $\mu_m(t) = 0$, $P_m(t) = 0$ for all nodes $m \neq n$. If a node n is in transmission mode during slot t, it sets $\tilde{P}_n(t) = P_{tran}$, and chooses commodity $\tilde{c}_n(t)$ and rate $\tilde{\mu}_n(t)$ that corresponds to this greedy control action $\tilde{I}_n(t)$. Thus, each transmitting node greedily selects a commodity and rate using the local differential backlog and link success probabilities associated with neighboring nodes, under the assumption that these other nodes are all in receive mode.

To analyze this algorithm, note that by definition of $\tilde{I}_n(t)$, we have for all t:

$$M_n(\tilde{I}_n(t), \tilde{c}_n(t), S(t)) \geq M_n(I^*(t), c_n^*(t), S(t)) \quad (23)$$

Define $\tilde{\boldsymbol{c}}(t) \triangleq (\tilde{c}_1(t), \ldots, \tilde{c}_N(t))$, and define $\tilde{I}(t)$ as the collective greedy control actions of all nodes:

$$\tilde{I}(t) \triangleq [(\tilde{\mu}_1(t), \ldots, \tilde{\mu}_N(t)); (\tilde{P}_1(t), \ldots, \tilde{P}_n(t))]$$

Furthermore, the random transmitter selection ensures:

$$\mathbb{E}\left\{M_n(\tilde{I}(t), \tilde{c}_n(t), S(t)) \mid \boldsymbol{U}(t), S(t)\right\} \geq q(1-q)^{J+1} M_n(\tilde{I}_n(t), \tilde{c}_n(t), S(t)) \quad (24)$$

This inequality can be understood as follows: The value $M_n(\tilde{I}_n(t), \tilde{c}_n(t), S(t))$ is achieved exactly in the case when only node n transmits, all other nodes are in receive mode, and the reception events and corresponding ACK/NACK feedback takes place according to the reception probabilities associated with the network channels (not including collision effects). Let k be the node that would be selected in this no-collision scenario (possibly being node n itself). Then conditional on these same channel events in the actual experiment, this node k would be chosen with probability at least $q(1-q)^{J+1}$, where q is the probability that node n indeed enters transmission mode, and $(1-q)^{J+1}$ bounds the probability that node k and all of the (at most J) nodes that can interfere with the n-to-k channel are in receiver mode. Therefore, the lower bound in (24) holds.

Note by definition that for any $I(t), \boldsymbol{c}(t), S(t)$:

$$M(I(t), \boldsymbol{c}(t), S(t)) = \sum_n M_n(I(t), c_n(t), S(t))$$

Therefore:

$$\begin{aligned} &\mathbb{E}\left\{M(\tilde{I}(t), \tilde{\boldsymbol{c}}(t), S(t)) \mid \boldsymbol{U}(t), S(t)\right\} \\ &\quad = \sum_n \mathbb{E}\left\{M_n(\tilde{I}(t), \tilde{c}_n(t), S(t)) \mid \boldsymbol{U}(t), S(t)\right\} \\ &\quad \geq \sum_n q(1-q)^{J+1} M_n(\tilde{I}_n(t), \tilde{c}_n(t), S(t)) \quad (25) \\ &\quad \geq q(1-q)^{J+1} \sum_n M_n(I^*(t), c_n^*(t), S(t)) \quad (26) \\ &\quad = q(1-q)^{J+1} M(I^*(t), \boldsymbol{c}^*(t), S(t)) \end{aligned}$$

where (25) follows by (24) and (26) follows by (23).

Therefore, this random transmitter selection algorithm satisfies the condition of Theorem 4 with $\gamma = q(1-q)^{J+1}$, $C = 0$. The value of q that yields the largest γ is given by $q = 1/(J+2)$, resulting in:

$$\gamma = \frac{(1 - 1/(J+2))^{J+1}}{J+2}$$

If $J = 0$, then we have $q = 1/2$, $\gamma = 1/4$. If J is large then we have $\gamma \approx \frac{1}{(J+2)e}$. We further note that combinations of random transmitter and random commodity selection can be used to further reduce complexity (with corresponding performance tradeoffs).

VII. Delayed Feedback and Other Extensions

Suppose that instead of using the actual differential backlog values $\{W_{nk}^{(c)}(t)\}$ in the generalized DIVBAR algorithm, alternative weights $\{\tilde{W}_{nk}^{(c)}(t)\}$ are used, where $\tilde{W}_{nk}^{(c)}(t)$ satisfies for all n, k, c and all t:

$$\left|\tilde{W}_{nk}^{(c)}(t) - W_{nk}^{(c)}(t)\right| \leq D \quad (27)$$

for some finite constant D. This would occur, for example, when the DIVBAR algorithm uses *out-of-date* queue backlog information due to delayed feedback (described in more detail below). Let $\tilde{I}(t), \tilde{\boldsymbol{c}}(t)$ represent the control action and commodity choices of a modified DIVBAR that uses these alternative weights, and let $I^*(t)$ and $\boldsymbol{c}^*(t)$ represent the choices that minimize $M(I(t), \boldsymbol{c}(t), S(t))$ (where $M(\cdot)$ is defined according to the *true* weights $\{W_{nk}^{(c)}(t)\}$). It is not difficult to see that:

$$M(\tilde{I}, \tilde{\boldsymbol{c}}(t), S(t)) \geq M(I^*(t), \boldsymbol{c}^*(t), S(t)) - DN$$

and hence using the alternative weights ensures that the conditions of Theorem 4 are satisfied with $\gamma = 1$, $C = DN$. Therefore, these modified weights create only a (potential) increase in network congestion and delay while maintaining full throughput optimality and energy efficiency.

A. Delay Improvement via Enhanced DIVBAR (E-DIVBAR)

The DIVBAR algorithm uses back-pressure to learn efficient routes, where incoming data "pushes" old data in directions of least resistance. However, when the network is lightly loaded, many packets may be routed in inappropriate directions before enough backlog builds up to suggest alternative routes. An extreme example is the case when a single packet arrives to an empty network. This packet could be routed randomly back and forth and might never reach its destination. One approach that potentially reduces delay in these situations is to impose an additional constraint that restricts routing options to directions that make progress toward the destination. However, such additional constraints might reduce network capacity, and can restrict adaptation in cases of link failures.

An alternative is to apply the Enhanced Dynamic Routing and Power Control (EDRPC) approach developed for link-based networks in [17] [18] to this multi-receiver diversity context. Specifically, for each actual queue backlog $U_n^{(c)}(t)$, define a *modified* backlog metric $\tilde{U}_n^{(c)}(t)$ as follows:

$$\tilde{U}_n^{(c)}(t) \triangleq U_n^{(c)}(t) + X_n^{(c)}(t) \quad (28)$$

where $X_n^{(c)}(t)$ are non-negative weights that satisfy $X_n^{(c)}(t) \leq D$ for all t (for some finite constant D). Modified differential backlogs $\tilde{W}_{nk}^{(c)}(t)$ are then used, where:

$$\tilde{W}_{nk}^{(c)}(t) = \max[\tilde{U}_n^{(c)}(t) - \tilde{U}_k^{(c)}(t), 0]$$

Clearly these modified weights satisfy (27), and hence throughput optimality and energy efficiency is unaffected. While the analytical congestion *bound* increases under these modified weights, in practice it is possible to choose $X_n^{(c)}(t)$ to *improve* delay, particularly in lightly loaded situations. For example, $X_n^{(c)}(t)$ can be chosen to be proportional to the estimated number of hops from node n to destination c along a shortest path. The $\tilde{W}_{nk}^{(c)}(t)$ values then include the estimated hop count differential associated with sending from node n to node k, so that data tends to be routed in the direction of the shortest path, and only deviates from this when backlog starts to build up along the path. Alternatively, one can define $X_n^{(c)}(t)$ as an estimated *geographic distance* between node n and destination c, so that routing decisions tend to move data closer to the destination.

B. Delayed Feedback

Suppose now that ACK/NACK feedback is delayed for a maximum of T timeslots. This is the case when the physical signal propagation is long (as in an underwater acoustic network) and/or when previous feedback signals had errors and are thus retransmitted with current feedback. The absence of feedback within T slots is considered equivalent to a NACK. We modify the DIVBAR algorithm in this context as follows: For each node n, let $U_n^{(c)}(t)$ represent the backlog of commodity c packets in node n as before, but define two new queues $X_n^{(c)}(t)$ and $Y_n^{(c)}(t)$ which can be viewed as waiting areas for packets requiring feedback and final instructions. Any packet within the $U_n^{(c)}(t)$ backlog that is transmitted by node n at time t is placed into the $X_n^{(c)}(t)$ queue to await feedback from the potential receivers. Any receiver k that successfully receives the packet places it into its queue $Y_k^{(c)}(t)$ to await final instructions.

Each feedback ACK from a receiver k is assumed to also contain the queue state $U_k^{(c)}(\tau) + X_k^{(c)}(\tau)$ at the time of reception τ. Once all feedback messages are received or the feedback delay time T has expired, node n makes a decision about which successful receiver should forward the packet, using modified differential backlog values $\hat{W}_{nk}^{(c)}(t)$. These differential backlog values are computed using $\hat{U}_n^{(c)}(t) = U_n^{(c)}(t) + X_n^{(c)}(t)$ and using $\hat{U}_k^{(c)}(t)$ values that are estimates of the true $U_k^{(c)}(t) + X_k^{(c)}(t)$ values based on the latest received queue update. Once a decision has been made, the corresponding $X_n^{(c)}(t)$ packet is removed. If the decision is for node n to keep responsibility of the packet, the packet is placed back into the $U_n^{(c)}(t)$ queue. Otherwise, a final instruction is delivered to the successful receiver k that should take responsibility for the packet. Once such an instruction is received, the packet at receiver k is removed from the $Y_k^{(c)}(t)$ queue and placed into the $U_k^{(c)}(t)$ queue. If no final instruction is received within $2T$ slots, the packet is deleted from the $Y_k^{(c)}(t)$ queue.

These modified differential backlogs can be shown to differ by a constant from the differential backlogs that would be used if all feedback that is *going* to be received were to be received immediately on each slot. It follows that (27) holds for a constant D that is proportional to T. Therefore, if all feedback is guaranteed to be received within the time limits, then this delayed feedback scheme yields full throughput optimality and energy efficiency as described in Theorem 4, with a potential increase in average congestion and delay. Otherwise, throughput and energy expenditure is degraded by an amount that depends on the fraction of packets that are successfully transmitted but whose feedback is not received within the specified time limits. The idea is to set the feedback timer T large enough so that such delay violations are rare and do not significantly effect network performance.

The reader should notice a potential difficulty with this approach in the case when a final instruction message is delivered by node n indicating that a particular receiver k should take charge of a given packet, but this final instruction message itself is not received by node k within the required time. In that case, both node n and node k (and all other potential receivers) delete the packet and hence this packet disappears from the network. Such events can be handled separately with a variety of techniques. For example, one might relay on an end-to-end transport layer to detect such errors and re-inject lost packets back into the network. Alternatively, a more sophisticated link-by-link feedback ARQ mechanism, similar to standard link layer Go-Back-n schemes [37], could be envisioned.

C. Utility Optimization and Average Power Constraints

The DIVBAR algorithm uses the Lyapunov optimization framework of [17] [33], and hence it can easily be adapted to optimize general utility metrics as well as to satisfy general constraints. In particular, optimization of general utility and fairness metrics can be achieved in cases when the input rate matrix is *either inside or outside of the capacity region* Λ by using the optimal flow control techniques of [17] [20] [33] together with DIVBAR. Similarly, the *virtual queue* technique developed in [32] can easily be incorporated to maximize utility subject to average power constraints at every node.

VIII. SIMULATIONS

We now present simulation results on the performance of DIVBAR on two example networks.

Example 1: A Static Network with Independent Links

We first consider a static network with independent non-interfering links and success probabilities as shown in Fig.3. There are two sessions in the network: Source node A desires to send data to destination A', and source node B desires to send data to destination B'. We consider a time-slotted system in which new packets arrive at the two sources every slot according to a Bernoulli process of rate λ. These need to be routed to their respective destinations. Each node can broadcast at most one packet per slot to its neighbors. Packet receptions are independent over each link, with link success probabilities as shown in the figure. We simulate DIVBAR and E-DIVBAR on this network, and compare to the ExOR strategy of [7]. The ExOR algorithm labels each node n with a "shortest path" estimate $X_n^{(c)}$, representing a

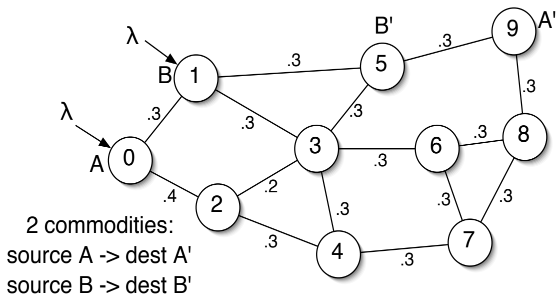


Fig. 3. Example static network used to compare ExOR, DIVBAR and E-DIVBAR. The number on a link is the probability of successful packet reception on that link.

"distance" estimate between node n and destination c based on link distances given by the inverse of the link success probability. The E-DIVBAR algorithm adds this $X_n^{(c)}$ value to the differential backlog metric according to (28). For DIVBAR and E-DIVBAR, the control parameter V is set to 0.

Simulations were conducted for all algorithms for different λ values ranging from 0 to 0.5, and each simulation was run for 1 million timeslots. The resulting average congestion is shown for each experiment in Fig. 4. Under ExOR, it can be shown that packets only traverse routes involving nodes $0, 1, 3, 5, 9$. The maximum rate λ that can be stably supported by both sessions using the ExOR policy can be calculated to be 0.255 packets/slot (see vertical asymptote in Fig. 4). However, DIVBAR and E-DIVBAR can support a higher rate (in fact, they achieve the maximum possible rate over all diversity algorithms that satisfy the structural properties described in Section II). This maximum throughput can be calculated to be 0.455 packets/slot. It should also be noted that while the average total occupancy of DIVBAR exceeds that of ExOR under light loadings, E-DIVBAR has the best performance across all input rates.

We next simulate DIVBAR on this network for increasing values of the control parameter V after fixing the input rate at $\lambda = 0.3$ packets/slot. Note that this rate cannot be supported

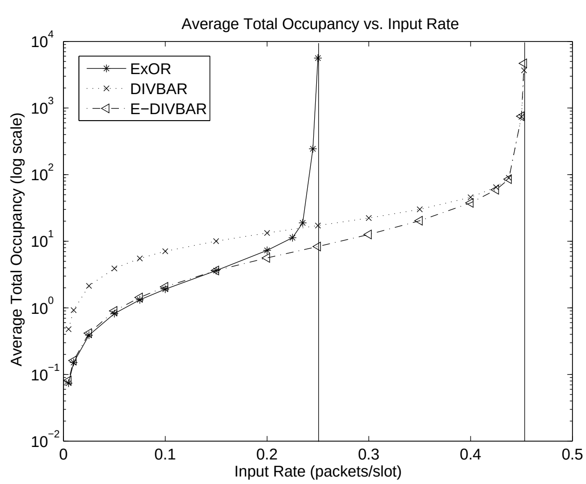
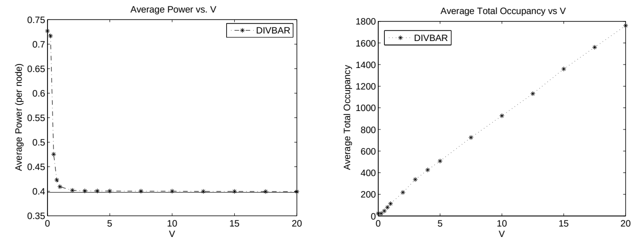


Fig. 4. Comparison of Average Total Occupancy under ExOR, DIVBAR and E-DIVBAR on the example static network.



(a) (b)

Fig. 5. Average Power and Average Total Occupancy versus V under DIVBAR on the example static network.

by ExOR. Fig. 5(a) shows that the average power converges to about 0.399 as V increases. Fig. 5(b) shows a linear increase in average total occupancy as V is increased. These results clearly exhibit the $[O(1/V); O(V)]$ energy-delay tradeoff as suggested by the performance bounds of Theorem 2.

Example 2: A Mobile Network with Heterogeneous Mobility

We next consider a network with cell-partitioned structure as shown in Fig. 6. There are 9 source nodes: 3 stationary, 3 locally mobile and 3 fully mobile. The locally mobile nodes are restricted to move in the shaded cells while the fully mobile nodes can move anywhere in the network. There are 2 stationary sinks and packets can be delivered to either of them (thus, this is a single commodity scenario). Time is slotted and new packets arrive at the source nodes every slot according to a Bernoulli process of rate λ. The mobile nodes perform a Markovian random walk over their respective regions, with equal probability of moving either North, South, East, or West (if a node decides to move in an infeasible region, it stays in its same cell). The steady state location distribution of each mobile node is thus uniform over its feasible cell locations.

Similar to the previous example, each node can broadcast at most one packet per slot on its outgoing links, and a packet transmitted on a link is successfully received by a node with probability equal to the success probability of that link. This value is taken to be 0.9 for links in the same cell and 0.5 for links between adjacent cells (defined as cells that are either horizontal, vertical, or diagonal neighbors). We assume multi-user reception is possible. However, we impose the additional constraint that nodes cannot simultaneously transmit and receive (as in the case $J = 0$ of Section VI-B). We consider a randomized algorithm where each node

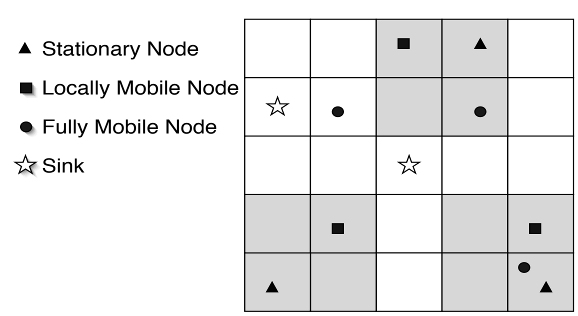


Fig. 6. A mobile network with two sinks and heterogeneous mobility.

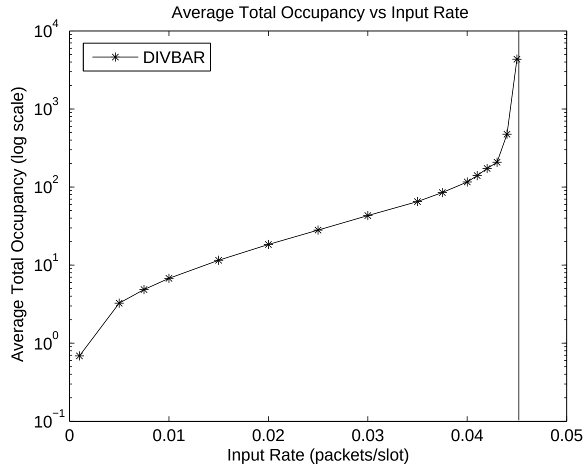
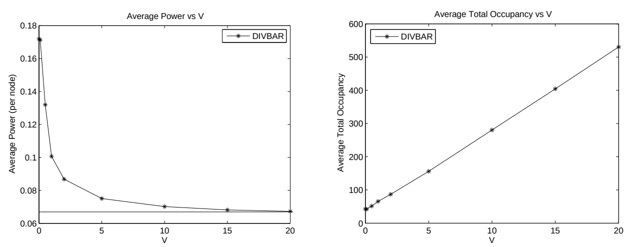


Fig. 7. Average Total Occupancy with increasing load under DIVBAR on the example mobile network.



(a) (b)

Fig. 8. Average Power and Average Total Occupancy versus V under DIVBAR on the example mobile network.

with packets to send decides to enter transmission mode with probability $q = 1/2$, which ensures capacity is achieved to within a factor of 4 (see Section VI-B).

We simulate DIVBAR on this network. Fig. 7 shows the average total occupancy with increasing input rate. The vertical asymptote occurs around $\lambda = 0.045$ packets/slot. Figs. 8(a) and 8(b) show results when we fix $\lambda = 0.03$ packets/slot. The average power and total average occupancy are plotted versus V, demonstrating the $[O(1/V); O(V)]$ energy-delay tradeoff of Theorem 2.

APPENDIX A — PROOF OF THEOREM 1

Here we prove the Network Capacity and Minimum Cost Theorem (Theorem 1). Consider a network with input rate matrix $(\lambda_n^{(c)})$, and suppose there exists a stabilizing control strategy (possibly one that uses redundant packet transfers). Let $\overline{h}$ represent the lim inf of the average power cost of this strategy:

$$\overline{h} \triangleq \liminf_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1}\sum_{n=1}^{N} h_n(P_n(\tau)) \tag{29}$$

We show that there exist multi-commodity flow variables $\{f_{ab}^{(c)}\}$ and probabilities $\alpha_n^{(c)}(s), \theta_{nk}^{(c)}(\Omega_n)$ that satisfy (4)-(6), and further that $\overline{h}$ is greater than or equal to the value $\overline{h}^*$ defined in Theorem 1. This proves that the constraints (4)-(6) are *necessary* for network stability, and that no stabilizing algorithm can achieve an average power cost less than $\overline{h}^*$.

Let $X_n^{(c)}(t)$ represent the number of commodity c packets that exogenously arrive to network node n during the first t timeslots. Define a *unit* as a packet or a replicated copy of a packet (replicated units are also considered to be units). When a unit is successfully transmitted from one node to another, we say that the original unit is retained in the transmitting node while a copy of the unit is created in the new node. In this case, we say that the unit in the transmitter is the *parent* of the new unit in the receiver. The *commodity* of a unit is defined as the commodity of its original packet and is the same as the *destination node* of the unit. The *source node* of a unit is the source node of its original packet. Two units are said to be *distinct* if they are copies of distinct original packets.

Let $Y_n^{(c)}(t)$ represent the number of distinct units with source node n and commodity c that have been successfully delivered to their destination during the first t timeslots. Because the algorithm is assumed to be *rate stable*, with probability 1 and for all (n, c), the delivery rate is equal to the input rate:

$$\lim_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1}Y_n^{(c)}(\tau) = \lim_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1}X_n^{(c)}(\tau) = \lambda_n^{(c)} \tag{30}$$

Let $\mathcal{U}_n^{(c)}(t)$ be the set of distinct units that were the first to reach their destination by time t (if two non-distinct units reach their destination at the same time, we arbitrarily assign only one of them to $\mathcal{U}_n^{(c)}(t)$). Thus, there are exactly $Y_n^{(c)}(t)$ units within the set $\mathcal{U}_n^{(c)}(t)$. Define the *ancestors* of a given unit u to be the set consisting of the parent of u, the parent of the parent, etc., all the way up to the original packet. For each successfully delivered unit u, define $path(u)$ as the sequence of nodes $\{n_1, n_2, \ldots, n_k\}$ associated with transmission of its ancestors, where n_1 is the source and n_k is the destination (for some finite integer k that depends on the number of transmissions). The path might repeat one or more nodes several times. However, if a unit u was successfully delivered to its destination, then for any node n along its path, the number of times an ancestor of u is endogenously transmitted *into* node n must be exactly the same as the number of times an ancestor of u is transmitted *out* of node n (provided that node n is not the destination). Defining $F_{ab}^{(c)}(t)$ to be the total number of times that ancestors of distinct commodity c units within the set $\cup_m \mathcal{U}_m^{(c)}(t)$ have been transmitted from node a to node b, we thus have for each node n and commodity c:

$$Y_n^{(c)}(t) + \sum_a F_{an}^{(c)}(t) = \sum_b F_{nb}^{(c)}(t) \quad \text{whenever } n \neq c \tag{31}$$

Note that $\sum_c F_{ab}^{(c)}(t)$ is the total number of ancestors of delivered units within $\cup_{m,c}\mathcal{U}_m^{(c)}(t)$ that successfully traverse link (a, b). Now define for all channel states s, all nodes a, b and commodities c, and all subsets Ω_a:

- $T_s(t)$: the number of times the channel state is equal to s during the first t slots.
- $\alpha_a^{(c)}(s, t)$: the number of times a commodity c packet attempts transmission at node a while the channel state is s (during the first t slots).
- $q_{a,\Omega_a}^{(c)}(s, t)$: the number of times a transmission of a commodity c packet in node a is correctly received

exactly by the subset Ω_a when the channel state is s (during the first t slots).

- $\theta_{ab}^{(c)}(\Omega_a, s, t)$: the number of times a commodity c unit within the set $\cup_m \mathcal{U}_m^{(c)}(t)$ is created at node b by a parent at node a when the topology state is s and the set of correctly received packets is equal to Ω_a (during the first t slots).

It follows that:

$$\frac{\sum_c F_{ab}^{(c)}(t)}{t} = \sum_{c,s,\Omega_a} \frac{T_s(t)}{t} \frac{\alpha_a^{(c)}(s,t)}{T_s(t)} \frac{q_{a,\Omega_a}^{(c)}(s,t)}{\alpha_a^{(c)}(s,t)} \frac{\theta_{ab}^{(c)}(\Omega_a,s,t)}{q_{a,\Omega_a}^{(c)}(s,t)} \tag{32}$$

where we formally define $0/0 \triangleq 0$ for terms on the right hand side of (32).[4] Note that for all t we have:

$$0 \leq \frac{\alpha_a^{(c)}(s,t)}{T_s(t)} \leq 1 \ , \ 0 \leq \frac{\theta_{ab}^{(c)}(\Omega_a,s,t)}{q_{a,\Omega_a}^{(c)}(s,t)} \leq 1 \tag{33}$$

Further, because the control strategy conforms to the system constraints, we have:

$$\textstyle\sum_c \frac{\alpha_a^{(c)}(s,t)}{T_s(t)} \leq 1 \ , \ \frac{\alpha_a^{(c)}(s,t)}{T_s(t)} = 0 \text{ if } \hat{\chi}_a(s) = 0 \tag{34}$$

$$\frac{\theta_{ab}^{(c)}(\Omega_a,s,t)}{q_{a,\Omega_a}^{(c)}(s,t)} = 0 \text{ if } b \notin \{\Omega_a \cup \{a\}\} \tag{35}$$

$$\textstyle\sum_{k=1}^N \frac{\theta_{ak}^{(c)}(\Omega_a,t)}{q_{a,\Omega_a}^{(c)}(s,t)} \leq 1 \tag{36}$$

Because channel states are i.i.d. over slots, we have by the law of large numbers:

$$\lim_{t\rightarrow\infty} \frac{T_s(t)}{t} = \pi_s \text{ with prob. 1}$$

Likewise, because transmission probabilities $\hat{q}_{a,\Omega_a}(s)$ do not depend on the commodity c transmitted, we have by the law of large numbers:

$$\lim_{t\rightarrow\infty} \frac{q_{a,\Omega_a}^{(c)}(s,t)}{\alpha_a^{(c)}(s,t)} = \hat{q}_{a,\Omega_a}(s) \text{ with prob. 1}$$

whenever $\alpha_a^{(c)}(s,t) \rightarrow \infty$ as $t \rightarrow \infty$.

Now define $f_{ab}^{(c)}(t) \triangleq F_{ab}^{(c)}(t)/t$, and note that:

$$0 \leq f_{ab}^{(c)}(t) \leq 1 \ , \ f_{cb}^{(c)} = f_{aa}^{(c)} = 0 \tag{37}$$

Let t_i represent a sequence of time slots over which the time average power cost achieves its liminf $\overline{h}$. Because the constraints in (33)-(37) define a closed and bounded region with finite dimension, there must exist an infinite subsequence $\tilde{t}_i$ over which the time average power cost also achieves its liminf, and the individual terms converge to points $\alpha_a^{(c)}(s)$, $\theta_{ab}^{(c)}(\Omega_a, s)$, $f_{ab}^{(c)}$ that also satisfy the inequalities (33)-(37):

$$\begin{aligned} \lim_{\tilde{t}_i\rightarrow\infty} \frac{1}{\tilde{t}_i} \sum_{\tau=0}^{\tilde{t}_i-1} \sum_{n=1}^N h_n(P_n(\tau)) &= \overline{h} \\ \lim_{\tilde{t}_i\rightarrow\infty} \frac{\alpha_a^{(c)}(s,\tilde{t}_i)}{T_s(\tilde{t}_i)} &= \alpha_a^{(c)}(s) \\ \lim_{\tilde{t}_i\rightarrow\infty} \frac{\theta_{ab}^{(c)}(\Omega_a,s,\tilde{t}_i)}{q_{a,\Omega_a}^{(c)}(s,\tilde{t}_i)} &= \theta_{ab}^{(c)}(\Omega_a,s) \\ \lim_{\tilde{t}_i\rightarrow\infty} f_{ab}^{(c)}(\tilde{t}_i) &= f_{ab}^{(c)} \end{aligned} \tag{38}$$

Furthermore, using (30) in (31) and taking $\tilde{t}_i \rightarrow \infty$ yields:

$$\lambda_n^{(c)} + \sum_a f_{an}^{(c)} = f_{nb}^{(c)} \text{ whenever } n \neq c$$

Likewise, using the above limits in (32) as $\tilde{t}_i \rightarrow \infty$ yields:

$$\sum_c f_{ab}^{(c)} = \sum_{c,s,\Omega_a} \pi_s \alpha_a^{(c)}(s) \hat{q}_{a,\Omega_a}(s) \theta_{ab}^{(c)}(\Omega_a,s) \text{ for all } (a,b)$$

Now define:

$$\theta_{ab}^{(c)}(\Omega_a) \triangleq \frac{\sum_s \pi_s \alpha_a^{(c)}(s) \hat{q}_{a,\Omega_a}(s) \theta_{ab}^{(c)}(\Omega_a,s)}{\sum_s \pi_s \alpha_a^{(c)}(s) \hat{q}_{a,\Omega_a}(s)}$$

assuming the denominator is non-zero (define $\theta_{ab}^{(c)}(\Omega_a, s) \triangleq 0$ otherwise). Thus:

$$\sum_c f_{ab}^{(c)} = \sum_{c,s,\Omega_a} \pi_s \alpha_a^{(c)}(s) \hat{q}_{a,\Omega_a}(s) \theta_{ab}^{(c)}(\Omega_a) \text{ for all } (a,b)$$

This proves that there exist suitable multi-commodity flow variables and probability variables for which the rate matrix $(\lambda_n^{(c)})$ satisfies the constraints of Theorem 1.

Finally, note that the time average power cost satisfies:

$$\frac{1}{\tilde{t}_i} \sum_{\tau=0}^{\tilde{t}_i-1} \sum_n h_n(P_n(\tau)) = \sum_s \frac{T_s(t)}{t} \sum_{n,c} \frac{\alpha_n^{(c)}(s,t)}{T_s(t)} h_n(P_{tran})$$

Using (38) in the above equality and taking $\tilde{t}_i \rightarrow \infty$ yields:

$$\overline{h} = \sum_s \pi_s \sum_{n,c} \alpha_n^{(c)}(s) h_n(P_{tran})$$

Thus, the value $\overline{h}$ corresponds to particular variables $f_{ab}^{(c)}$, $\alpha_a^{(c)}(s)$, $\theta_{ab}^{(c)}(\Omega_a)$ that satisfy the constraints of Theorem 1. It follows that $\overline{h} \geq \overline{h}^*$, where $\overline{h}^*$ is defined as the minimum value of h for which such variables can be found that satisfy the constraints. This proves the result.

APPENDIX B – PROOF OF LEMMA 1

Proof: (Lemma 1) To show that the DIVBAR control actions minimize the right hand side of (14), define $f(t)$ as the sum of all terms on the right hand side that involve control decision variables:

$$f(t) \triangleq V\mathbb{E}\{h(\boldsymbol{P}(t)) \mid \boldsymbol{U}(t)\} - g(t) \tag{39}$$

where $g(t)$ is defined as follows:

$$g(t) \triangleq \sum_{n,c} U_n^{(c)}(t) \mathbb{E}\left\{ \sum_b \beta_{nb}^{(c)}(t) - \sum_a \beta_{an}^{(c)}(t) \mid \boldsymbol{U}(t) \right\}$$

[4] Equivalently, in (32) we can replace terms x/y with $[x/y]$, where $[x/y]$ is defined to be x/y whenever $y \neq 0$, and is 0 otherwise.

Switching the sums yields:

$$\begin{aligned} g(t) &= \sum_{n,k,c} \mathbb{E}\left\{\beta_{nk}^{(c)}(t) \mid \boldsymbol{U}(t)\right\}\left[U_n^{(c)}(t) - U_k^{(c)}(t)\right] \\ &\leq \sum_{n,k,c} \mathbb{E}\left\{\beta_{nk}^{(c)}(t) \mid \boldsymbol{U}(t)\right\} W_{nk}^{(c)}(t) \end{aligned}$$

where the final inequality follows by definition of $W_{nk}^{(c)}(t)$ in (10). Now note that the control constraints (2) imply:

$$\beta_{nk}^{(c)}(t) = \beta_{nk}^{(c)}(t)\mu_n^{(c)}(t)H_{nk}(t) \text{ for all } n, k, c, t$$

This is because $\mu_n^{(c)}(t)H_{nk}(t) \in \{0, 1\}$, and $\beta_{nk}^{(c)}(t)$ can only be non-zero when $\mu_n^{(c)}(t)H_{nk}(t) = 1$. Therefore:

$$\begin{aligned} g(t) &\leq \sum_{n,k,c} \mathbb{E}\left\{\beta_{nk}^{(c)}(t)\mu_n^{(c)}(t)H_{nk}(t) \mid \boldsymbol{U}(t)\right\} W_{nk}^{(c)}(t) \\ &= \sum_{n,c} \mathbb{E}\left\{\mu_n^{(c)}(t) \sum_k \beta_{nk}^{(c)}(t)H_{nk}(t)W_{nk}^{(c)}(t) \mid \boldsymbol{U}(t)\right\} \\ &\leq \sum_{n,c} \mathbb{E}\left\{\mu_n^{(c)}(t) \max_k \left\{H_{nk}(t)W_{nk}^{(c)}(t)\right\} \mid \boldsymbol{U}(t)\right\} \quad (40) \end{aligned}$$

where the final inequality follows because the constraints (2) imply that $\beta_{nk}^{(c)}(t) \geq 0$ and $\sum_k \beta_{nk}^{(c)}(t) \leq 1$. However, by the law of iterated expectations we have:

$$\begin{aligned} &\mathbb{E}\left\{\mu_n^{(c)}(t) \max_k \left\{H_{nk}(t)W_{nk}^{(c)}(t)\right\} \mid \boldsymbol{U}(t)\right\} \\ &= \mathbb{E}\left\{\mu_n^{(c)}(t) \cdot \right. \\ &\qquad \left. \mathbb{E}\left\{\max_k \left\{H_{nk}(t)W_{nk}^{(c)}(t)\right\} \mid \boldsymbol{U}(t), \mu_n^{(c)}(t)\right\} \mid \boldsymbol{U}(t)\right\} \\ &\leq W_n^*(t)\mathbb{E}\left\{\mu_n^{(c)}(t) \mid \boldsymbol{U}(t)\right\} \end{aligned}$$

where the final inequality is due to (12). Plugging the above inequality into (40) yields:

$$\begin{aligned} g(t) &\leq \sum_{n,c} W_n^*(t)\mathbb{E}\left\{\mu_n^{(c)}(t) \mid \boldsymbol{U}(t)\right\} \\ &\leq \sum_n W_n^*(t)\mathbb{E}\left\{\mu_n(t) \mid \boldsymbol{U}(t)\right\} \quad (41) \end{aligned}$$

where the final inequality holds because $\sum_c \mu_n^{(c)}(t) \leq \mu_n(t)$. However, the upper bound (41) can be *achieved* if node n transmits commodity $c_n^*(t)$, receives ACK/NACK feedback, and shifts forwarding responsibilities to the successful receiver k with the largest positive value of $W_{nk}^{(c_n^*(t))}(t)$ (retaining the packet if no successful receivers have positive differential backlog). It follows from (39) that:

$$\begin{aligned} f(t) &\geq V\mathbb{E}\left\{h(\boldsymbol{P}(t)) \mid \boldsymbol{U}(t)\right\} - \textstyle\sum_n W_n^*(t)\mathbb{E}\left\{\mu_n(t) \mid \boldsymbol{U}(t)\right\} \\ &= \textstyle\sum_n \mathbb{E}\left\{Vh_n(P_n(t)) - W_n^*(t)\mu_n(t) \mid \boldsymbol{U}(t)\right\} \end{aligned}$$

Furthermore, this lower bound on $f(t)$ is both *minimized and achieved* by the DIVBAR algorithm that allocates power P_{tran} for transmission from node n whenever $Vh_n(P_{tran}) < W_n^*(t)$, and that chooses the commodity and receiver node as described. This proves that DIVBAR minimizes $f(t)$ over all possible control actions, proving the lemma. □

APPENDIX C — PROOF OF THEOREM 4

Here we prove that generalized DIVBAR yields performance as given in Theorem 4. Recall that $\mu_n(t) \in \{0, 1, \ldots, \mu_{max}^{out}\}$ is the integer number of packet transmission opportunities provided by node n during slot t. The control action is given by $I(t) = (\boldsymbol{\mu}(t), \boldsymbol{P}(t))$. Define $\mu_n^{(c)}(t)$ as the number of commodity c packet opportunities, where:

$$\mu_n^{(c)}(t) \in \{0, 1, \ldots, \mu_n(t)\}\ ,\ \sum_c \mu_n^{(c)}(t) \leq \mu_n(t)\ ,\ \mu_c^{(c)}(t) = 0$$

Define $H_{nk,i}^{(c)}(t) \in \{0, 1\}$ as the random variable representing the reception outcome at receiver k for the ith commodity c packet transmitted from node n (so that $H_{nk,i}^{(c)}(t) = 1$ if $\mu_n^{(c)}(t) \geq i$ and this ith packet is correctly received, and 0 else). The value $H_{nk,i}^{(c)}(t)$ can be viewed as the ACK/NACK information for this packet. Define $\beta_{nk,i}^{(c)}(t) \in \{0, 1\}$ as the packet forwarding decision, being 1 if node n plans to forward this packet to node k, and zero else. The $\beta_{nk,i}^{(c)}(t)$ decision variables are chosen according to the following constraints:

$$\beta_{nk,i}^{(c)}(t) \leq H_{nk,i}^{(c)}(t)\ ,\ \sum_k \beta_{nk,i}^{(c)}(t) \leq 1$$

The first inequality states that we can only transfer responsibility of a packet to node k if node k has correctly received the packet. The second inequality states that we can only transfer responsibility of a given packet to a single receiver. If $\sum_k \beta_{nk,i}^{(c)}(t) = 0$, then node n retains responsibility of this packet. Note that these constraints generalize the constraints (1) and (2) for the basic DIVBAR algorithm.

The queueing dynamics thus satisfy:

$$\begin{aligned} U_n^{(c)}(t+1) \leq \max&\left[U_n^{(c)}(t) - \sum_{k=1}^{N}\sum_{i=1}^{\mu_{max}^{out}} \beta_{nk,i}^{(c)}(t), 0\right] \\ &+ \sum_{a=1}^{N}\sum_{j=1}^{\mu_{max}^{out}} \beta_{an,j}^{(c)}(t) + A_n^{(c)}(t) \quad (42) \end{aligned}$$

This is an inequality due to the fact that there may not be enough commodity c packets available at node a to transfer the full $\sum_{j=1}^{\mu_{max}^{out}} \beta_{an,j}^{(c)}(t)$ packets to node n. Note in particular that, as in the case of the basic DIVBAR algorithm, this in principle allows the $\mu_n^{(c)}(t)$, $\beta_{nk,i}^{(c)}(t)$ values to be chosen independently of queue backlog. Define $\Theta(t) \triangleq [I(t); \{\beta_{nk}^{(c)}(t)\}]$ as the collective control decision at time t.

Suppose the topology state process $S(t)$ is i.i.d. over slots, and suppose exogenous arrivals are i.i.d. over slots with rate matrix $\boldsymbol{\lambda} = (\lambda_n^{(c)})$. As before, define $L(\boldsymbol{U}) = \frac{1}{2}\sum_{n,c}(U_n^{(c)}(t))^2$, and define $\Delta(\boldsymbol{U}(t))$ as the one step Lyapunov drift. The drift can be computed from (42) and satisfies (compare with (14)):

$$\begin{aligned} &\Delta(\boldsymbol{U}(t)) + V\mathbb{E}\left\{h(\boldsymbol{P}(t)) \mid \boldsymbol{U}(t)\right\} \leq \\ &\qquad NB + \sum_{n,c} U_n^{(c)}(t)\lambda_n^{(c)} + \mathbb{E}\left\{Q(\Theta(t)) \mid \boldsymbol{U}(t)\right\} \quad (43) \end{aligned}$$

where $Q(\Theta(t))$ is the portion of the drift bound that depends on the control decisions, and is defined:

$$Q(\Theta(t)) \triangleq Vh(\boldsymbol{P}(t)) - \sum_{n,c} U_n^{(c)}(t) \left[\sum_{b,i} \beta_{nb,i}^{(c)}(t) - \sum_{a,i} \beta_{an,i}^{(c)}(t) \right] \quad (44)$$

Claim 1: The generalized DIVBAR algorithm in Section VI that uses the sub-optimal control decisions specified in Theorem 4 makes control decisions $I(t) = (\boldsymbol{\mu}(t), \boldsymbol{P}(t))$ and $\beta_{nk,i}^{(c)}(t)$ satisfy:

$$\mathbb{E}\left\{Q(\Theta(t)) \mid \boldsymbol{U}(t)\right\} \leq C + \gamma \mathbb{E}\left\{Q(\Theta^{opt}(t)) \mid \boldsymbol{U}(t)\right\} \quad (45)$$

where $\Theta^{opt}(t)$ are the control decisions that minimize $\mathbb{E}\left\{Q(\Theta(t)) \mid \boldsymbol{U}(t)\right\}$ over all feasible controls.

Proof: It suffices to prove that the optimal resource allocation decisions of the generalized DIVBAR algorithm minimize $Q(\Theta(t))$ over all alternative controls. The proof of this fact is similar to the proof of Lemma 1 in Section V-A, and is omitted for brevity. □

Define Λ as the closure of all stabilizable rate matrices, and suppose the exogenous input rate matrix is $\boldsymbol{\lambda}$ is within Λ.

Claim 2: If $\boldsymbol{\lambda} \in \Lambda$, then there exists a stationary randomized control algorithm that makes decisions $\Theta^*(t) = [I^*(t); \{\beta_{nk}^{*(c)}(t)\}]$ according to the system constraints and that satisfies for all slots t:

$$\begin{aligned} \sum_{a,i} \mathbb{E}\left\{\beta_{an,i}^{*(c)}(t)\right\} + \lambda_n^{(c)} &\leq \sum_{b,i} \mathbb{E}\left\{\beta_{nb,i}^{*(c)}(t)\right\} \forall n \neq c \\ \mathbb{E}\left\{h(\boldsymbol{P}^*(t))\right\} &= \Phi(\boldsymbol{\lambda}) \end{aligned}$$

The proof of Claim 2 is similar to that of Theorem 1 and its corollary from Section III (compare with (8) and (9)), and is omitted for brevity. Suppose now that there is a positive value ϵ_{max} such that $(\lambda_n^{(c)} + \epsilon_{max} 1_n^{(c)}) \in \gamma\Lambda$, where γ is the constant in (45) and satisfies $0 < \gamma \leq 1$. Below we prove Theorem 4.

Proof: (Theorem 4) From Claim 1, we know that the drift bound in (43) satisfies:

$$\Delta(\boldsymbol{U}(t)) + V\mathbb{E}\left\{h(\boldsymbol{P}(t)) \mid \boldsymbol{U}(t)\right\} \leq NB + \sum_{n,c} U_n^{(c)}(t)\lambda_n^{(c)} + C + \gamma \mathbb{E}\left\{Q(\Theta^{opt}(t)) \mid \boldsymbol{U}(t)\right\} \quad (46)$$

Define $\boldsymbol{\epsilon} \triangleq (\epsilon 1_n^{(c)})$ (for a given $\epsilon > 0$). We know that $\boldsymbol{\lambda} + \boldsymbol{\epsilon} \in \gamma\Lambda$ whenever $\epsilon \leq \epsilon_{max}$, and hence $\frac{1}{\gamma}(\boldsymbol{\lambda} + \boldsymbol{\epsilon}) \in \Lambda$. Therefore, we know by Claim 2 that there exists a stationary randomized control action $\Theta^*(t)$ that makes decisions independent of queue backlog, and that yields (using (44)):

$$\mathbb{E}\left\{Q(\Theta^*(t)) \mid \boldsymbol{U}(t)\right\} \leq V\Phi(\frac{1}{\gamma}(\boldsymbol{\lambda} + \boldsymbol{\epsilon})) - \sum_{n,c} U_n^{(c)}(t) \frac{(\lambda_n^{(c)} + \epsilon)}{\gamma} \quad (47)$$

(note that $U_n^{(c)}(t) = 0$ whenever $1_n^{(c)} = 0$). Because $\Theta^{opt}(t)$ minimizes $\mathbb{E}\left\{Q(\Theta(t)) \mid \boldsymbol{U}(t)\right\}$ over all alternative controls, we have that:

$$\mathbb{E}\left\{Q(\Theta^{opt}(t)) \mid \boldsymbol{U}(t)\right\} \leq \mathbb{E}\left\{Q(\Theta^*(t)) \mid \boldsymbol{U}(t)\right\}$$

Using this fact in (46) and (47) yields:

$$\begin{aligned} \Delta(\boldsymbol{U}(t)) + V\mathbb{E}\left\{h(\boldsymbol{P}(t)) \mid \boldsymbol{U}(t)\right\} &\leq NB + \sum_{n,c} U_n^{(c)}(t)\lambda_n^{(c)} + C \\ &\quad + \gamma V \Phi(\frac{1}{\gamma}(\boldsymbol{\lambda} + \boldsymbol{\epsilon})) - \sum_{n,c} U_n^{(c)}(t)(\lambda_n^{(c)} + \epsilon) \\ &= NB + C + \gamma V \Phi(\frac{1}{\gamma}(\boldsymbol{\lambda} + \boldsymbol{\epsilon})) - \epsilon \sum_{n,c} U_n^{(c)}(t) \end{aligned}$$

The above drift expression is in the exact form for application of the Lyapunov drift theorem (Theorem 3), and hence:

$$\limsup_{t \to \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\left\{h(\boldsymbol{P}(\tau))\right\} \leq \gamma \Phi(\frac{1}{\gamma}(\boldsymbol{\lambda} + \boldsymbol{\epsilon})) + \frac{NB + C}{V}$$

$$\limsup_{t \to \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \sum_{n,c} \mathbb{E}\left\{U_n^{(c)}(t)\right\} \leq \frac{NB + C + \gamma V h_{max}}{\epsilon}$$

The above inequalities hold for any ϵ such that $0 < \epsilon \leq \epsilon_{max}$. Taking a limit as $\epsilon \to 0$ and using continuity of $\Phi(\cdot)$ yields the power cost bound of Theorem 4, and setting $\epsilon = \epsilon_{max}$ yields the congestion bound. □

REFERENCES

[1] M. J. Neely. Optimal backpressure routing for wireless networks with multi-receiver diversity. *Proc. of Conf. on Information Sciences and Systems (CISS), Invited paper on Optimization of Communication Networks*, March 2006.

[2] 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.

[3] 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.

[4] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides. Algorithms for energy-efficient multicasting in ad hoc wireless networks. *Proc. IEEE Military Communications Conf.*, pp. 1414-1418, 1999.

[5] A. Srinivas and E. Modiano. Minimum energy disjoint path routing in wireless ad-hoc networks. *IEEE Proc. of Mobicom*, September 2003.

[6] 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.

[7] S. Biswas and R. Morris. Exor: Opportunistic multi-hop routing for wireless networks. *Proc. of Sigcomm*, 2005.

[8] F. Baccelli, B. Blaszczyszyn, and P. Muhlethaler. An aloha protocol for multihop mobile wireless networks. *IEEE Transactions on Information Theory*, 2006.

[9] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transacations on Automatic Control*, vol. 37, no. 12, pp. 1936-1949, Dec. 1992.

[10] 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.

[11] P.R. Kumar and S.P. Meyn. Stability of queueing networks and scheduling policies. *IEEE Trans. on Automatic Control*, vol.40,.n.2, pp.251-260, Feb. 1995.

[12] N. McKeown, V. Anantharam, and J. Walrand. Achieving 100% throughput in an input-queued switch. *Proc. IEEE INFOCOM*, 1996.

[13] N. Kahale and P. E. Wright. Dynamic global packet routing in wireless networks. *Proc. IEEE INFOCOM*, 1997.

[14] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, and P. Whiting. Providing quality of service over a shared wireless link. *IEEE Communications Magazine*, vol. 39, no.2, pp.150-154, 2001.

[15] E. Leonardi, M. Mellia, F. Neri, and M. Ajmone Marsan. Bounds on average delays and queue size averages and variances in input-queued cell-based switches. *Proc. IEEE INFOCOM*, 2001.

[16] 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*, vol. 11, no. 1, pp. 138-152, Feb. 2003.

[17] 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.

[18] 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*, vol. 23, no. 1, pp. 89-103, January 2005.

[19] M. J. Neely. Energy optimal control for time varying wireless networks. *Proc. IEEE INFOCOM*, March 2005.

[20] M. J. Neely, E. Modiano, and C. Li. Fairness and optimal stochastic control for heterogeneous networks. *Proc. IEEE INFOCOM*, March 2005.

[21] R. Cruz and A. Santhanam. Optimal routing, link scheduling, and power control in multi-hop wireless networks. *Proc. IEEE INFOCOM*, April 2003.

[22] 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.

[23] A. Stolyar. Maximizing queueing network utility subject to stability: Greedy primal-dual algorithm. *Queueing Systems*, vol. 50, pp. 401-457, 2005.

[24] J. W. Lee, R. R. Mazumdar, and N. B. Shroff. Opportunistic power scheduling for dynamic multiserver wireless systems. *IEEE Transactions on Wireless Communications*, vol. 5, no.6, pp. 1506-1515, June 2006.

[25] C. Swannack, E. Uysal-Biyikoglu, and G. Wornell. Low complexity multiuser scheduling for maximizing throughput in the mimo broadcast channel. *Proc. of 42nd Allerton Conf. on Communication, Control, and Computing*, September 2004.

[26] M. Kobayashi. *On the Use of Multiple Antennas for the Downlink of Wireless Systems*. PhD thesis, Ecole Nationale Superieure des Telecommunications, Paris, 2005.

[27] M. Kobayashi, G. Caire, and D. Gesbert. Impact of multiple transmit antennas in a queued SDMA/TDMA downlink. In *Proc. of 6th IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, June 2005.

[28] T. Ho and H. Viswanathan. Dynamic algorithms for multicast with intra-session network coding. In *Proc. of 43rd Allerton Conf. on Communication, Control and Computing*, September 2005.

[29] 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.

[30] A. Khandani, J. Abounadi, E. Modiano, and L. Zheng. Cooperative routing in wireless networks. *Proc. of Allerton Conf. on Communications, Control, and Computing*, pp. 1270-1279, Oct. 2003.

[31] S. Katti, D. Katabi, W. Hu, H. Rahul, and M. Medard. The importance of being opportunistic: Practical network coding for wireless environments. *Proc. of* 43^{rd} *Annual Allerton Conf. on Communication, Control, and Computing*, Oct. 2005.

[32] M. J. Neely. Energy optimal control for time varying wireless networks. *IEEE Transactions on Information Theory*, vol. 52, no. 7, July 2006.

[33] L. Georgiadis, M. J. Neely, and L. Tassiulas. Resource allocation and cross-layer control in wireless networks. *Foundations and Trends in Networking*, vol. 1, no. 1, pp. 1-149, 2006.

[34] X. Lin and N. B. Shroff. The impact of imperfect scheduling on cross-layer rate control in wireless networks. *Proc. IEEE INFOCOM*, 2005.

[35] X. Wu and R. Srikant. Regulated maximal matching: A distributed scheduling algorithm for multi-hop wireless networks with node-exclusive spectrum sharing. *Submitted to IEEE Conf. on Decision and Control*, 2005.

[36] P. Chaporkar, K. Kar, and S. Sarkar. Throughput guarantees through maximal scheduling in wireless networks. *Proc. of 43rd Annual Allerton Conf. on Communication Control and Computing*, September 2005.

[37] D. P. Bertsekas and R. Gallager. *Data Networks*. New Jersey: Prentice-Hall, Inc., 1992.