Control of Multi-Node Mobile Communications Networks with Time Varying Channels via Stability Methods

Harold J. Kushner
Applied Mathematics Dept.
Brown University
Providence RI 02912 USA
hjk@dam.brown.edu

July 15, 2005

Abstract

Consider a communications network consisting of mobiles, each of which can be scheduled to serve as a receiver and/or transmitter. There are random external data processes arriving at some of the mobiles, each destined for some set of destinations. Each mobile can serve as a node in the possibly multi-hop path from source to destination. At each mobile the data is queued according to the source-destination pair until transmitted. Time is divided into small scheduling intervals. The capacity or quality of the connecting channels are randomly varying due to the motion of the mobiles and consequent scattering. At the beginning of the intervals, the channels are estimated via pilot signals and this information can be used for the scheduling decisions. The issues are the allocation of transmission power and/or time, bandwidth, and perhaps antennas, to the various queues at the various mobiles in a queue and channel-state dependent way to assure stability and good operation. Lost packets might or might not have to be retransmitted. The decisions are made at the beginning of the scheduling intervals. In a recent work, stochastic stability methods were used to develop scheduling policies for the simple system where there is a single transmitter communicating with many mobiles. The resulting controls were readily implementable and allowed a range of tradeoffs between current rates and queue lengths, under very weak conditions. Here the basic methods and results are extended to the network case. The choice of Liapunov function allows a choice of the effective performance criteria. All essential factors are incorporated into a "mean rate" function, so that the results cover many different systems. Because of the non-Markovian nature of the problem, we use the perturbed Stochastic Liapunov function method, which is designed for such problems. Extensions concerning acknowledgments, multicasting, non-unique routes, and others are given to illustrate the versatility of the method, and a useful method for getting the a priori routes is discussed.

keywords: Scheduling in stochastic networks, randomly-varying link capacities, mobile networks, stochastic stability, stability of networks with randomly varying links, routing in ad-hoc networks, perturbed stochastic Liapunov functions,

1 Introduction

Consider a network of M mobiles (to be referred to as nodes). There are S external sources that create bursty data processes. Each one sends its data to its unique origin node in the network, and the data is to be sent through the network to a destination node. Until Section 4 the destination node for each source will be unique. Some of the mobiles can act as intermediate nodes in the possibly multi-hop connections between sources and destinations. The routing is a priori fixed for each of the source-destination pairs. A useful method for getting such a priori routes is discussed in the final section. The route for each source-destination pair need not be unique, and will not generally be so if the links are subject to failure. Comments on the few changes needed for the non-unique route case are discussed in Section 4. At each mobile the data is queued, until transmitted, in an infinite buffer depending on the source-destination pair. The issues are the allocation of transmission power and/or time and bandwidth to the various queues at the various mobiles in a queue and channel-state dependent way to assure stability and good operation. Time is divided into small scheduling intervals. The capacities of the connecting channels are a correlated random process, possibly due to the motion of the mobiles and consequent scattering. At the beginning of the intervals, the capacities (or surrogates such as the S/N ratios) are estimated where possible via pilot signals and this information is used for the scheduling during that interval. The resource allocation decisions are made at the beginning of the intervals. Such an approach greatly improves the performance [1]. An origin node for one source might be an intermediate node for another. This ties the analysis for all nodes together. Owing to the random nature of the arrival and channel processes, the computation or even the existence of stabilizing policies is not at all obvious. The approach is a network extension of the development for the one-node case in [4].

Owing to the usually non-Markovian nature of these processes,[1] classical stability methods cannot be used, and a perturbed Liapunov function method [4, 8] is adapted to obtain stabilizing scheduling policies. Loosely speaking, with this method, and X denoting the vector of queue values at all the nodes (all data quantities are measured in packets), one starts with a basic Liapunov function $V(X)$ that works for a special type of "mean flow" system. Then one seeks a perturbation $\delta V(n)$ to this basic Liapunov function so that $V(X(n)) + \delta V(n)$ can be used as a Liapunov function for the actual non-Markov physical

[1] For example, Rayleigh fading is not Markovian, but will satisfy the assumptions.

system and imply the desired stability. Analogously to the usual "stability method" procedure that is used to get controls, the controls are determined by "approximately" minimizing the (conditional expectation, given the current data) of the rate of change of the basic (not the perturbed) Liapunov function along the random path. The algorithm is readily implemented. For notational simplicity, the development uses a basic Liapunov function that is a polynomial which is the sum of terms, each depending on only one component of the state of the queue. This seems adequate, but strictly convex separable functions can also be used. See the comments below. The basic result is that, if a certain "mean flow" or fluid approximation process is stable, then so is the physical system under our scheduling rule. This stabilizability of the mean flow approximation can often be readily verified. It will be argued that the condition is "nearly" necessary as well.

To illustrate the versatility of the approach, a sampling of extensions are discussed in Section 4. Up to that section, it is not required that acknowledgments of receipt be sent to the source, and it is shown there how to include the failure to receive an ack. Under broad conditions, if the probability of packet loss on the path for source i is p_i, then the path must be able to handle a "mean flow" that is $1/(1-p_i)$ greater, to account for the retransmissions. Multicasting and non unique routes are discussed, as is the case where there is "opportunistic" scheduling, where channels become available depending on the need of priority users. Some channels might be unavailable on random intervals due to other uses or breakdowns. Various cases where the number of users varies randomly can also be handled.

Let the $(n+1)$st scheduling interval be called the nth slot. The time argument (n) denotes the beginning of the nth slot, and is referred to as "time n." Let $X_{i,k}(n)$ denote the queue size at time n at node k of data coming from source i. If node k is not on the path for source i, then $X_{i,k}(n) \equiv 0$. Define the vectors $X_k(n) = \{X_{i,k}(n), i \leq S\}$ and $X(n) = \{X_k(n), k \leq M\}$, with canonical values X_k and X, resp. The basic Liapunov function will be[2]

$$V(X) = \sum_{i,k} w_{i,k} X_{i,k}^p, \quad p \geq 2, \tag{1.1}$$

where the $w_{i,k} \geq 0$ are weights. More generally, one could use powers depending in i,k, with the same end result. The form $V(X) = \sum_{i,k} V_{i,k}(X_{i,k})$, where the $V_{i,k}(\cdot)$ are strictly convex non-negative functions, whose first derivative (with respect to $X_{i,k}$) $DV_{i,k}(X_{i,k})$ is $o(V_{i,k}(X_{i,k}))$ and second derivative is $o(DV_{i,k}(X_{i,k}))$ can be used. One can choose the functions, for example, to model upper bound constraints on some queues. The choice of the functions and powers allows a variety of tradeoffs between queue size and throughput. We use (1.1) since the notation is simpler. But the development is parallel for the other cases, and the same conditions would be used.

[2] In analogy to what was done in [4] one could use the form $\sum_{i,k} w_{i,k}[X_{i,k} + h_{i,k}]^{p_{i,k}}$, where $p_{i,k} \geq 2, h_{i,k} \geq 0$. In this paper, due to the greater notational complexity due to the network formulation, we prefer to compute with a simpler Liapunov function (1.1)

Part of the interest in stability is that it assures a robustness of behavior to small changes in the system. For this reason, as well as because $\{X(n)\}$ is rarely Markovian, it is preferable to use methods that do not require Markovianness. The perturbed Liapunov function method is a powerful tool for checking such robustness since it does not require Markovianness. Generally, there are many criteria that are of interest to each of the users, e.g., mean delay and variance of delay. One should experiment with the form of the Liapunov function to see what the tradeoffs are between competing criteria, a procedure that yields better results than simply working with a single fixed rule, whatever it is, and this variety is facilitated by the wide choice of possibilities for the functions $V(X)$.

There is much good work on scheduling in the face of various types of randomness. But very little is available on scheduling for the general network case when the channels are randomly varying in a non-trivial way. For the one-node case, if the rate of transmission is proportional to power, then [1, 10] gets rules for power allocation whose form is similar to ours when $p = 2$ (and are called "max weight" rules there), and which are based on stability considerations. The method uses large deviations estimates and the setup is Markovian. The reference [13] considered the problem of dynamic power allocation when the channels are time-varying. But, since their channel-rate and data-arrival processes are all i.i.d. sequences, the range of applications is very small.

The papers [2, 3] deal with related problems, again essentially for one-node systems. There is a set of parallel processors, and the connectivities between the sources and the processors (but not the outgoing channels) vary randomly. They prove results concerning the limit (as $t \to \infty$) of (queue length at t)$/t$, and show that (under appropriate conditions) this limit is zero. This is used to show that the integral of the "rates" of transmission per unit time converges. Such a result does not imply stability of the queue length process, since it can grow sublinearly. They allocate a single resource (e.g., bandwidth) and the rate is proportional to the allocation. Their proof is much more complicated than that here. Our proof could be easily adapted to that problem, with the definition of stability to be used here. See comments on non-unique routing in Section 4. The work [12], for a one node model, has a Markovian channel-state process, the data input sequence is i.i.d., and a "complete resource pooling" condition is required. The decision rule is the same as ours for a quadratic Liapunov function. The emphasis is on stability in the heavy traffic limit, and showing how the problem simplifies there. See also [5] for a stability analysis as the heavy traffic regime is approached.

2 Definitions and Assumptions

We refer to the queue for source i at node k as queue (i, k). If the path for source i does not use node k, then the queue does not exist. Let k denote a canonical node, and $f(i, k)$ the node that the queue of source i at node k feeds to after leaving node k. I.e., queue (i, k) feeds to queue $(i, f(i, k))$. If node k is the final destination for source i data, then terms involving $f(i,k)$ are ignored. Let $b(i,k)$ denote the node that queue i at node k is fed from. I.e., queue $(i, b(i,k))$ feeds to queue (i,k). If node k is the origin node for source i, then terms involving $b(i,k)$ are ignored. Let $\mathcal{F}_n$ denote the minimal σ-algebra that measures all of the systems data up until time n as well as the channel state in slot n. Recall that this channel state is assumed to be available at time n, the beginning of the nth slot. Let E_n denote the expectation conditioned on $\mathcal{F}_n$. For simplicity of terminology, we say that the packets sent in slot n are sent at time n. Let $d_{i,k}(n)$ denote the number of packets sent from queue (i,k) at time n. It will depend on the channel state at that time and will be a function of the resources (e.g., power, frequency, bandwidth) allocated to that queue. It is zero if node k is not on the path for source i. Let $a_{i,k}(n)$ denote the actual random number of arrivals in slot n from the exterior, if any, from source i at node k. These will be non-zero only for the unique node $k(i)$ at which source i enters the network.

Stability: Definition. Owing to the non-Markovianess, an appropriate definition of stability is a "uniform mean recurrence time" property, as follows. This is the definition used in [4]. Suppose that there are $0 < q_0 < \infty$ and a real-valued function $F(\cdot) \geq 0$ such that the following holds: For any n, and $\sigma_1 = \min\{k \geq n : |X(k)| \leq q_0\}$, we have[3]

$$E_n\left[\sigma_1 - n\right] \leq F(X(n)) I_{\{|X(n)| \geq q_0\}}. \tag{2.1}$$

Then the system is said to be stable. The definition implies recurrence to some compact set. If $|X(n)|$ reaches a level $q_1 > q_0$, then the conditional expectation of the time required to return to a value q_0 or smaller is bounded by a function of q_1, uniformly in the past history and in n. This implies that the sequence $\{X(n)\}$ is tight or bounded in probability (see, for example, [8, Theorem 2, Chapter 6]).[4] It is worth noting that the right side of (2.1) depends only on $X(n)$. It does not depend on any other data, even though there is a conditional expectation E_n on the left side, and the channel and arrival processes are random and correlated.

The decision rule. The number of packets transmitted from queue (i,k) in slot n is $d_{i,k}(n)$, and this depends on the fundamental resources that are committed; e.g., power, bandwidth, time, etc. The assignments are always subject to constraints. These might simply be bounds on the total power available at a node or on the number of packets than can be sent in a slot, in which case the determination of the $d_{i,k}(n)$ for all i can be made at node k. If the constraints involve more than one node (e.g., if neighboring nodes cannot use the same carrier frequency in narrowband communications), then the assignments require coordination among the nodes.

Following the idea in classical stability-control theory, the idea is to choose

[3] $\sigma_1 = \infty$, unless otherwise defined.

[4] A sequence $\{X_n\}$ is bounded in probability if $\lim_{\kappa \to \infty} \sup_n P\{|X_n| \geq \kappa\} = 0$.

the $d_{i,k}(n)$ that attains (subject to constraints)

$$\min_{\{d_{i,k};i,k\}} [E_n V(X(n+1)) - V(X(n))]$$

as well as possible. To motivate the form of the decision rule, let us evaluate $E_n V(X(n+1)) - V(X(n))$. We have

$$w_{i,k}\left[E_n X^p_{i,k}(n+1) - X^p_{i,k}(n)\right] = w_{i,k} X^{p-1}_{i,k}(n)\left[-d_{i,k}(n) + E_n a_{i,k}(n) + d_{i,b(i,k)}(n)\right] \\ + \text{ terms of order } (p-2) \text{ in } X_{i,k}(n).$$

Summing over i and k yields, modulo terms of order $p-2$ in $X(n)$ and the "arrival" terms,

$$-\sum_{i,k}\left[w_{i,k} X^{p-1}_{i,k}(n) - w_{i,f(i,k)} X^{p-1}_{i,f(i,k)}(n)\right] d_{i,k}(n) \tag{2.2}$$

or, equivalently,

$$-\sum_{i,k} w_{i,k} X^{p-1}_{i,k}(n)\left[d_{i,k}(n) - d_{i,b(i,k)}(n)\right]. \tag{2.3}$$

The lower order terms in $X_{i,k}(n)$ are nonlinear functions of $d_{i,k}(n)$ and higher conditional moments of the $a_{i,k}(n)$, and would be much too hard to deal with. It turns out, as in [4], that is is enough to work with the term that is first order in the decisions, which are just the terms in (2.2) and (2.3).

If the decisions are made independently at each node k, then our decision rule (for each node k) is a maximizer in

$$\max_{\{d_{i,k}(n):i\}} \sum_i \left[w_{i,k} X^{p-1}_{i,k}(n) - w_{i,f(i,k)} X^{p-1}_{i,f(i,k)}(n)\right] d_{i,k}(n), \tag{2.4}$$

subject to the constraints at node k. If there are constraints that involve the decisions at a set of nodes, then the decisions for the nodes in such a set must be made together, and decision rule is a maximizer in

$$\max_{\{d_{i,k}(n);i,k\}} \sum_{i,k} \left[w_{i,k} X^{p-1}_{i,k}(n) - w_{i,f(i,k)} X^{p-1}_{i,f(i,k)}(n)\right] d_{i,k}(n), \tag{2.5}$$

or in

$$\max_{\{d_{i,k}(n);i,k\}} \sum_{i,k} w_{i,k} X^{p-1}_{i,k}(n)\left[d_{i,k}(n) - d_{i,b(i,k)}(n)\right]. \tag{2.6}$$

It is always understood that the maximizations in (2.4), (2.5), or (2.6) are subject to whatever constraints there are.

$V(X)$ is rarely a Liapunov function for the system since $X(n)$ is rarely Markovian, so classical stochastic stability theory [7] cannot be used directly. However the perturbed Liapunov function method [4, 8, 9] allows us to show, under very reasonable conditions, that the maximizing rules (2.4), (2.5), or (2.6) yield a stable system. The appropriate perturbations to the Liapunov function will be given in the next section.

Let $L_k(n)$ denote the (vector) set of channel states, at time n, of all of the channels originating at node k. It is the channel state for the set of links $\{(i,k),(i,f(i,k)) : \text{all } i \text{ using node } k\}$. For example, if there are N_k possible links from node k to other nodes, then $L_k(n)$ is an N_k-vector, and at node k, the canonical channel state j is also an N_k-vector. $L_k(n)$ could be just the set of S/N ratios at the receiver corresponding to unit transmitted power, or it might be some other indicator of capacity, or it might indicate that no information is available. It is notationally convenient to work with the vector $L_k(n)$, rather than with the individual links, since the decisions at each node k depend on the states of all of the outgoing links. $L_k(n)$ might denote other quantities besides the channel quality. For example, there might be power constraints that vary randomly due to interference issues from exogenous sources. These could be included in the $L_k(n)$. If some link at node k is unavailable at time n, then that fact could also be included in $L_k(n)$. For notational simplicity, we suppose that the range of values of the channel state vector is a finite set for each node k. We use the (vector-valued) symbol j for the canonical value of $L_k(n)$, for any k,n. The range of the variable j will depend on the node k in question, and, for simplicity, will not be specified in the notation.

Let $u_{i,k}(j,X)$ denote the control function at queue i at node k. It represents the allocated resources (power, time, bandwidth, etc.) at queue (i,k). Also, unless otherwise noted, its dependence on the queues is only on X_k and the required queue values at the immediate upstream nodes, namely the $X_{i,f(i,k)}$ for all i. If source i does not use node k, then ignore $u_{i,k}(j,X)$. The control $u_{i,k}(j,X)$ determines the amount of data that is sent. Let the function $g_{i,k}(j,X_{i,k},u_{i,k}(j,X))$ denote the actual amount of data that is sent from queue (i,k) under channel state j and control $u_{i,k}(\cdot)$. This defines $d_{i,k}(n)$; i.e., the channel rate for queue (i,k) associated with current channel state $j = L_k(n)$ and control $u_{i,k}(j,X(n))$ is $d_{i,k}(n) = g_{i,k}(j,X_{i,k}(n),u_{i,k}(j,X(n)))$. The $X_{i,k}$ appears as an argument of $g_{i,k}(\cdot)$ only because the amount sent cannot be larger than the queue content.

Assumptions. The following assumptions are network analogs of those used in [4] and will be commented on after being stated. (A2.4) basically requires that there are controls under which the mean service rate at queue (i,k) for any i that uses node k is slightly greater than $\bar{\lambda}_i^a$ for all (i,k). Similar conditions occur frequently in studies of stability in stochastic networks.

A2.1. There are constraint sets U_k such that $\{u_{i,k}(j,X), i \le S\} \in U_k$. It is always assumed that the maximizing constrained $d_{i,k}(n)$ exist and are Borel functions of the $\{X(n), L_k(n), i, k\}$.

A2.2. There is a constant K_1 such that for all i, $E_n|a_i(n)|^p \le K_1$. There are

$\bar{\lambda}^a_{i,k}$ such that, for all i, the sums

$$\delta V^a_{i,k}(n) = \sum_{l=n}^{v} \left[E_n \alpha_{i,k}(l) - \bar{\lambda}^a_{i,k} \right]$$

converge as $v \to \infty$, uniformly in n, ω.

The $\bar{\lambda}^a_{i,k}$, which we call the mean external data arrival rate for source i at node k, is zero if node k is not the source node for source i. For future use, write $\bar{\lambda}^a_i = \bar{\lambda}^a_{i,k(i)}$ the mean input rate for source i (measured in packets per slot).

A2.3. For each node k there are $\Pi_{k,j} \geq 0$ such that $\sum_j \Pi_{k,j} = 1$ and $\sum_{l=n}^{v} \left[E_n I_{\{L_k(l)=j\}} - \Pi_{k,j} \right]$ converges as $v \to \infty$, uniformly in n, ω.

A2.4. Define $K_0 = \max_{i,k,j,u,X} g_{i,k}(j, X_{i,k}, u_{i,k}(j, X))$. There is a control $\{\tilde{u}_{i,k}(\cdot); i, k\}$ such that the following holds under it. There are $\{\tilde{q}^j_{i,k}; i, k\}$ such that $\tilde{q}^j_{i,k} = g_{i,k}(j, X_{i,k}(n), \tilde{u}_{i,k}(j, X(n)))$ if $X_{i,k}(n) \geq K_0$. [5] Also, $g_{i,k}(j, X_{i,k}(n), \tilde{u}_{i,k}(j, X(n))) \leq \tilde{q}^j_{i,k}$ if $X_{i,k}(n) < K_0$. The $\tilde{q}^j_{i,k}$ satisfy

$$\bar{q}_{i,k} = \sum_j \tilde{q}^j_{i,k} \Pi_{k,j} > \bar{\lambda}^a_i. \tag{2.7}$$

Comments on (A2.1)–(A2.3). (A2.1) states only that there are constraints on the resources and allocations. (A2.2) and (A2.3) are simply mixing conditions on the data arrival and channel processes, resp., and do not appear to be restrictive. For example, let $\Pi_{k,j}$ in (A2.3) denote the steady state probability of channel state j at node k. Then (A2.3) says that the conditional probability of state j at time l given the data to time n converges to the steady state value as $l - n \to \infty$, and that this convergence is fast enough for the sum to exist. It holds for the the received signal power associated with Rayleigh fading. Suppose that at node k, there are N_k possible links out, with the channel states in the links being mutually independent. Write the canonical channel state j as the N_k-vector $j = (j_1, \ldots, j_{N_k})$, with each j_l taking finitely many values. Then $\Pi_{k,j}$ is the product of the probabilities of the states of the individual N_k channels. If the channel character of the individual links are determined by the discretized S/N ratio resulting from, say, Rayleigh fading with unit signal power, then the probabilities of the states of the individual links are obtained from the associated stationary distribution. The condition (A2.2) is discussed in [4]. It is shown there that it holds under broad conditions. Loosely speaking, it is a quantification of the condition of asymptotic independence of the arrivals in the distant future with those in the remote past.

[5] The lower bound K_0 is introduced in (A2.4) only because if the queue content is smaller than the maximum of what can be transmitted on a scheduling interval, then the mean (weighed with the $\Pi_{k,j}$) output might be too small to assure the $-c_0$ value. For example if a queue is empty, then there are no departures.

Note on (A2.4). By the definition of the $\bar{q}_{i,k}$, for $k \neq k(i)$, the origin node for source i, we have $\bar{q}_{i,b(i,k)} = \sum_j \tilde{q}^j_{i,b(i,k)} \Pi_{b(i,k),j}$. Define $\bar{q}_{i,b(i,k(i))} = \bar{\lambda}^a_i$. It is implied by (A2.4) that there is $c_0 > 0$ such that the $\tilde{q}^j_{i,k}$ can be chosen to satisfy

$$\bar{\lambda}^a_i - \bar{q}_{i,k(i)} \leq -c_0, \tag{2.8a}$$

and, for $k \neq k(i)$,

$$\text{average into } (i,k) - \text{average out of } (i,k) = \bar{q}_{i,b(i,k)} - \bar{q}_{i,k} \leq -c_0 \tag{2.8b}$$

See Section 5 for further discussion of the choice of the $\tilde{q}^j_{i,k}$ and the a priori fixed routes.

Consider an example, where the control is over either power, bandwidth or time within the slot and the rates are proportional to the allocated resources. Let the allocated resource at (i,k) be denoted by $B^j_{i,k}$, let the constants of proportionality be $c^j_{i,k}$ and let the associated rate be $q^j_{i,k} = c^j_{i,k} B^j_{i,k}$. There are the available resource constraints $\sum_i B^j_{i,k} \leq B_k$ for each j,k, and the mean throughput constraints $\sum_j q^j_{i,k} \Pi_{k,j} > \bar{\lambda}^a_i$, all k. If there is a solution, then the corresponding $q^j_{i,k}$ satisfy (A2.4).

3 Liapunov Function Perturbations and Proof

Motivational comment on the perturbed Liapunov function method. Suppose that, for a random process $\{x(n)\}$, $E_n V(x(n+1)) - V(x(n)) = c_n$, where $\{c_n\}$ is a "mixing" random sequence. Let there is a constant $\bar{c} < 0$ such that $\delta V_n = \sum_{i=n}^{\infty} E_n[c_i - \bar{c}]$ is well defined and bounded. Define $V_n = V(x(n\Delta)) + \delta V_n$. Then $E_n \delta V_{n+1} - \delta V_n = -(c_n - \bar{c})$ and $E_n V_{n+1} - V_n = c_n - [c_n - \bar{c}] = \bar{c} < 0$. Thus, with the use of the perturbation we have replaced c_n by a "mean." The perturbed Liapunov function method is a development of this idea. It is applied as follows. Start by evaluating $E_n V(X(n+1)) - V(X(n))$. This will contain terms depending on the random arrivals and random channel states. Then add terms $\delta V(n)$ which are small relative to $V(X(n))$, but such that in

$$E_n[V(X(n+1)) - V(X(n))] + E_n[\delta V(n+1) - \delta V(n)]$$

the "bad" terms are cancelled and replaced by "averages," modulo terms that are suitably dominated.

The perturbations. We will now define the Liapunov function perturbation $\delta V(n)$. This will be a sum of terms, one corresponding to each input process and one corresponding to the input and one to the output of each queue. The motivation for the structure of the perturbations should be apparent from the way that they are used in the proof. Additional motivation is in [5, 8]. Recall that $k(i)$ denotes the arrival node for data from source i. The perturbations for the arrival terms are

$$\delta V_{i,k}^{a}(n) = w_{i,k} X_{i,k}^{p-1}(n) \sum_{l=n}^{\infty} E_n \left[a_{i,k}(l) - \bar{\lambda}_{i,k}^{a} \right]. \tag{3.1}$$

By the definitions, this is zero if $k \neq k(i)$.

Keep in mind that the channel state j is a vector and that for node k it denotes the canonical state of the set of channels on the forward links, those from the (i,k) to $(i, f(i,k))$ for all i that use node k. In (3.2), we define two sets of perturbations. The first one $(\delta V_{i,k,j}^{d,+}(n))$ is concerned with the effects of the departure of packets from a queue (i,k) on the value of $E_n X_{i,k}^{p}(n+1) - X_{i,k}^{p}(n)$, when the channel state at node k is j, and under the "reference" rates $\tilde{q}_{i,k}^{j}$ of (A2.4). The second one $(\delta V_{i,k,j}^{d,-}(n))$ is concerned with the effects on this value of the inputs to (i,k) from queue $(i, b(i,k))$, when the channel state at node $b(i,k)$ is j, and under the "reference" rates $\tilde{q}_{i,b(i,k)}^{j}$. Define

$$\begin{aligned} \delta V_{i,k,j}^{d,+}(n) &= -w_{i,k} X_{i,k}^{p-1}(n) \tilde{q}_{i,k}^{j} \sum_{l=n}^{\infty} E_n \left[I_{\{L_k(l)=j\}} - \Pi_{k,j} \right], \\ \delta V_{i,k,j}^{d,-}(n) &= \ w_{i,k} X_{i,k}^{p-1}(n) \tilde{q}_{i,b(i,k)}^{j} \sum_{l=n}^{\infty} E_n \left[I_{\{L_{b(i,k)}(l)=j\}} - \Pi_{b(i,k),j} \right]. \end{aligned} \tag{3.2}$$

Finally, define the Liapunov function perturbation $\delta V(n)$ and the full time-dependent Liapunov function $\tilde{V}(n)$ as

$$\begin{aligned} \delta V(n) &= \sum_{i,k} \delta V_{i,k}^{a}(n) + \sum_{i,k,j,\pm} \delta V_{i,k,j}^{d,\pm}(n), \\ \tilde{V}(n) &= V(X(n)) + \delta V(n). \end{aligned} \tag{3.3}$$

Theorem 3.1. *Under* (A2.1)–(A2.4) *the system is stable.*

Proof. The details are more complicated than those for the single node case due to the network structure. The function $\tilde{V}(n)$ is the (time-varying) Liapunov function that is to be used. We need to show that is has the supermartingale property for large queue state values; i.e, that there is $c < 0$ such that $E_n \tilde{V}(n+1) - \tilde{V}(n) \leq -c$ when $|X(n)|$ is large enough, and then that this inequality together with the bounds on the perturbations imply (2.1).

As usual in stability proofs, the first step is to evaluate

$$\begin{aligned} E_n \tilde{V}(n+1) - \tilde{V}(n) &= \sum_{i,k} w_{i,k} E_n \left[X_{i,k}^{p}(n+1) - X_{i,k}^{p}(n) \right] \\ &+ \sum_{i,k} E_n \left[\delta V_{i,k}^{a}(n+1) - \delta V_{i,k}^{a}(n) \right] + \sum_{i,k,j,\pm} E_n \left[\delta V_{i,k,j}^{d,\pm}(n+1) - \delta V_{i,k,j}^{d,\pm}(n) \right]. \end{aligned}$$

This will be done component by component, and then the results added. In the course of adding the components, many "undesirable" terms will be cancelled and replaced either by averages or by terms that can be suitably dominated. This is the key to the effectiveness of the method. Below, by "terms of order $(p-2)$" we mean terms that are bounded by $K|X(n)|^{p-2}+K$ for some constant K.

For the main component (1.1), a first order Taylor expansion yields

$$\sum_{i,k} w_{i,k} E_n \left[X^p_{i,k}(n+1) - X^p_{i,k}(n) \right] = \\ \sum_{i,k} w_{i,k} X^{p-1}_{i,k}(n) \left[E_n a_{i,k}(n) - d_{i,k}(n) + d_{i,b(i,k)}(n) \right] + \text{terms of order } (p-2). \tag{3.4}$$

Now consider the "arrival" component (3.1) for node (i,k). Note that, if k is the origin node for source i data, then a first order expansion yields

$$E_n \delta V^a_{i,k}(n+1) - \delta V^a_{i,k}(n) = -w_{i,k} X^{p-1}_{i,k}(n) \left[E_n a_{i,k}(n) - \bar{\lambda}^a_{i,k} \right] + \text{terms of order } (p-2).$$

Thus,

$$\begin{aligned} &\sum_{i,k} E_n \left[\delta V^a_{i,k}(n+1) - \delta V^a_{i,k}(n) \right] \\ &\qquad = -\sum_{i,k} w_{i,k} X^{p-1}_{i,k}(n) \left[E_n a_{i,k}(n) - \bar{\lambda}^a_{i,k} \right] + \text{terms of order } (p-2). \end{aligned} \tag{3.5}$$

For future reference, note that by adding (3.4) and (3.5), the $w_{i,k} X^{p-1}_{i,k}(n) E_n a_{i,k}(n)$ terms are cancelled, and the mean value term $w_{i,k} X^{p-1}_{i,k}(n) \bar{\lambda}^a_{i,k}$ term and an "error" term of order $p-2$ appear. The error term will be dominated by the terms of order $p-1$ for large values of the queue state. The replacement of the random arrival term by its mean value is crucial and was the main motivation for the form of the perturbation (3.1).

Now deal with the "departure" perturbation that is the first term in (3.2). This will eventually help to "average" the $d_{i,k}(n)$ term in (3.4). By the definitions,

$$\begin{aligned} E_n &\left[\delta V^{d,+}_{i,k,j}(n+1) - \delta V^{d,+}_{i,k,j}(n) \right] = \\ &\quad -w_{i,k} E_n X^{p-1}_{i,k}(n+1) \tilde{q}^j_{i,k} \sum_{l=n+1}^{\infty} E_{n+1} \left[I_{\{L_k(l)=j\}} - \Pi_{k,j} \right] \\ &\quad +w_{i,k} X^{p-1}_{i,k}(n) \tilde{q}^j_{i,k} \sum_{l=n}^{\infty} E_n \left[I_{\{L_k(l)=j\}} - \Pi_{k,j} \right] \end{aligned} \tag{3.6}$$

By splitting off the lowest summand from the sum in the last line, this expression can be written as

$$
\begin{aligned}
&w_{i,k}X_{i,k}^{p-1}(n)\tilde{q}_{i,k}^{j}\left[I_{\{L_k(n)=j\}}-\Pi_{k,j}\right]\\
&\quad -w_{i,k}E_nX_{i,k}^{p-1}(n+1)\tilde{q}_{i,k}^{j}\sum_{l=n+1}^{\infty}E_{n+1}\left[I_{\{L_k(l)=j\}}-\Pi_{k,j}\right]\\
&\quad +w_{i,k}X_{i,k}^{p-1}(n)\tilde{q}_{i,k}^{j}\sum_{l=n+1}^{\infty}E_n\left[I_{\{L_k(l)=j\}}-\Pi_{k,j}\right].
\end{aligned}
\tag{3.7}
$$

Writing $X_{i,k}^{p-1}(n+1)=X_{i,k}^{p-1}(n)+[X_{i,k}^{p-1}(n+1)-X_{i,k}^{p-1}(n)]$ and expanding the bracketed term yields the representation of (3.7)

$$
\begin{aligned}
&E_n\left[\delta V_{i,k,j}^{d,+}(n+1)-\delta V_{i,k,j}^{d,+}(n)\right]\\
&\qquad = w_{i,k}X_{i,k}^{p-1}(n)\tilde{q}_{i,k}^{j}\left[I_{\{L_k(n)=j\}}-\Pi_{k,j}\right]+\text{ terms of order }(p-2).
\end{aligned}
\tag{3.8}
$$

Analogously, one can show that

$$
\begin{aligned}
&E_n\left[\delta V_{i,k,j}^{d,-}(n+1)-\delta V_{i,k,j}^{d,-}(n)\right]=\\
&-w_{i,k}X_{i,k}^{p-1}(n)\tilde{q}_{i,b(i,k)}^{j}\left[I_{\{L_{b(i,k)}(n)=j\}}-\Pi_{b(i,k),j}\right]+\text{ terms of order }(p-2).
\end{aligned}
\tag{3.9}
$$

Adding all terms in (3.4), (3.5), (3.8), and (3.9), and cancelling where possible yields

$$
\begin{aligned}
E_n\tilde{V}(n+1)-\tilde{V}(n) &= \sum_{i,k}w_{i,k}X_{i,k}^{p-1}(n)\bar{\lambda}_{i,k}^{a}\\
&+\sum_{i,k}\left[-w_{i,k}X_{i,k}^{p-1}(n)d_{i,k}(n)+w_{i,k}X_{i,k}^{p-1}(n)d_{i,b(i,k)}(n)\right]\\
&+\sum_{i,k,j}w_{i,k}X_{i,k}^{p-1}(n)\tilde{q}_{i,k}^{j}\left[I_{\{L_k(n)=j\}}-\Pi_{k,j}\right]\\
&-\sum_{i,k,j}w_{i,k}X_{i,k}^{p-1}(n)\tilde{q}_{i,b(i,k)}^{j}\left[I_{\{L_{b(i,k)}(n)=j\}}-\Pi_{b(i,k),j}\right]\\
&+\text{ terms of order }(p-2).
\end{aligned}
\tag{3.10}
$$

Separate out the terms in the middle three lines of (3.10) that do not involve the $\Pi_{k,j}$ variables, getting

$$
\begin{aligned}
&-\sum_{i,k}w_{i,k}X_{i,k}^{p-1}(n)\left[d_{i,k}(n)-d_{i,b(i,k)}(n)\right]\\
&+\left\{\sum_{i,k}w_{i,k}X_{i,k}^{p-1}(n)\sum_{j}\left[\tilde{q}_{i,k}^{j}I_{\{L_k(n)=j\}}\right]\right.\\
&\qquad\left.-\sum_{i,k}w_{i,k}X_{i,k}^{p-1}(n)\sum_{j}\left[\tilde{q}_{i,b(i,k)}^{j}I_{\{L_{b(i,k)}(n)=j\}}\right]\right\}.
\end{aligned}
$$

For each k, the indicator functions in the sums over j merely pick out the current channel state $j=L_k(n)$ or $L_{b(i,k)}(n)$, as appropriate. Hence, we can rewrite the last expression as

$$
\begin{aligned}
&-\sum_{i,k} w_{i,k} X_{i,k}^{p-1}(n)\left[d_{i,k}(n) - d_{i,b(i,k)}(n)\right] \\
&+\sum_{i,k} w_{i,k} X_{i,k}^{p-1}(n)\left[\tilde{q}_{i,k}^{L_k(n)} - \tilde{q}_{i,b(i,k)}^{L_{b(i,k)}(n)}\right].
\end{aligned}
\tag{3.11}
$$

Suppose, for the moment, that all $X_{i,k}(n) \geq K_0$. Then, by (A2.4) there is a resource allocation $\{\tilde{u}_{i,k}(\cdot)\}$ such that, for each state j, the output from queue (i,k) will be $\tilde{q}_{i,k}^j = g_{i,k}(j, X_{i,k}(n), \tilde{u}_{i,k}(j, X(n)))$. Since the $d_{i,k}(n)$ are chosen either by the maximization rule (2.4) (which is implied by both (2.5) and (2.6)), or by the rules (2.5) or (2.6) (which are equivalent), and the $\tilde{q}_{i,k}^j$ outputs defined in (A2.4) are not necessarily maximizers in (2.6), the expression (3.11) is non-positive. Using this fact in (3.10) together with the definition of $\bar{q}_{i,k}$ in (A2.4) yields the following upper bound to (3.10):

$$
\begin{aligned}
&\sum_{i,k} w_{i,k} X_{i,k}^{p-1}(n)\left[\bar{\lambda}_{i,k}^a - \bar{q}_{i,k} + \bar{q}_{i,b(i,k)}\right] \\
&+ \text{terms of order } (p-2).
\end{aligned}
\tag{3.12}
$$

By (2.7), the terms in the brackets in the first line of (3.12) are $\leq -c_0 < 0$. Thus we have proved that

$$
E_n\tilde{V}(n+1) - \tilde{V}(n) \leq -c_0 \sum_{i,k} w_{i,k} X_{i,k}^{p-1}(n) + O(|X(n)|^{p-2}), \tag{3.13}
$$

We also have

$$
|\delta V(n)| = O(|X(n)|^{p-1}) \tag{3.14}
$$

and, by (3.13),

$$
E_n\tilde{V}(n+1) - \tilde{V}(n) \to \infty, \text{ uniformly in } n \text{ as } X(n) \to \infty. \tag{3.15}
$$

By (3.15), there are $c_1 > 0$ and $q_0 > 0$, such that, for $|X(n)| \geq q_0$,

$$
E_n\tilde{V}(n+1) - \tilde{V}(n) \leq -c_1. \tag{3.16}
$$

Given small $\delta > 0$, (3.14) implies that for q_0 sufficiently large,

$$
|V(X(n)) - \tilde{V}(n)| \leq \delta(1 + V(X(n)). \tag{3.17}
$$

Let σ_0 be a stopping time for which $|X(\sigma_0)| = c_2 > q_0$, and define the stopping time $\sigma_1 = \min\{n > \sigma_0 : |X(n)| \leq q_0\}$. Then, by (3.16), we have

$$
E_{\sigma_0}\tilde{V}(\sigma_1) - \tilde{V}(\sigma_0) \leq -c_1 E_{\sigma_0}[\sigma_1 - \sigma_0]. \tag{3.18}
$$

Using (3.18) and the bound (3.17) on $\tilde{V}(n) - V(X(n))$ to bound $\tilde{V}(\sigma_i) - V(X(\sigma_i))$, $i = 0, 1$, yields

$$
\begin{aligned}
&-\delta E_{\sigma_0}[1 + V(X(\sigma_1))] + E_{\sigma_0} V(X(\sigma_1)) \\
&\quad \leq E_{\sigma_0}\tilde{V}(\sigma_1) \leq -c_1 E_{\sigma_0}(\sigma_1 - \sigma_0) + [\delta + V(X(\sigma_0))(1+\delta)]
\end{aligned}
$$

or

$$E_{\sigma_0}(\sigma_1 - \sigma_0) \leq \frac{2\delta + V(X(\sigma_0))(1+\delta) + \delta E_{\sigma_0} V(X(\sigma_1))}{c_1},$$

which implies that the definition of stability (2.1) holds since $V(X(\sigma_1)) \leq \sup_{|x| \leq q_0} V(x)$.

Now, let us complete the details when some components of $X(n)$ are less than K_0. The required adjustments are minor. Recall the definition of $\tilde{u}_{i,k}(\cdot)$ and $\tilde{q}^j_{i,k}$ in (A2.4). Define $\tilde{g}_{i,k}(L_k(n), X(n)) = g_{i,k}(L_k(n), X_{i,k}(n), \tilde{u}_{i,k}(L_k(n), X(n)))$. For $X_{i,k}(n) \geq K_0$ we have $\tilde{g}_{i,k}(L_k(n), X(n)) = \tilde{q}^{L_k(n)}_{i,k}$. Otherwise $\tilde{g}_{i,k}(L_k(n), X(n))$ is smaller. By adding and subtracting identical terms, rewrite (3.11) as follows.

$$\begin{aligned}
&-\sum_{i,k} w_{i,k} X^{p-1}_{i,k}(n) \left[d_{i,k}(n) - d_{i,b(i,k)}(n)\right] \\
&+\sum_{i,k} w_{i,k} X^{p-1}_{i,k}(n) \left[\tilde{g}_{i,k}(L_k(n), X(n)) - \tilde{g}_{i,b(i,k)}(L_{b(i,k)}(n), X(n))\right] \\
&+\sum_{i,k: X_{i,k}(n) \geq K_0} w_{i,k} X^{p-1}_{i,k}(n) \left[\tilde{q}^{L_k(n)}_{i,k} - \tilde{g}_{i,k}(L_k(n), X(n))\right] \\
&+\sum_{i,k: X_{i,k}(n) < K_0} w_{i,k} X^{p-1}_{i,k}(n) \left[\tilde{q}^{L_k(n)}_{i,k} - \tilde{g}_{i,k}(L_k(n), X(n))\right] \\
&-\sum_{i,k} w_{i,k} X^{p-1}_{i,k}(n) \left[\tilde{q}^{L_{b(i,k)}(n)}_{i,b(i,k)} - \tilde{g}_{i,b(i,k)}(L_{b(i,k)}(n), X(n))\right]
\end{aligned}$$

Just as for the case where all $X_{i,k}(n) \geq K_0$, the sum of the first two lines is non-positive, since the $d_{i,k}(n)$ are chosen by the maximization rule. The third line is zero since the terms in the brackets are zero, by the definition of the $\tilde{g}_{i,k}$ when $X_{i,k}(n) \geq K_0$. Also, by (A2.4), the bracketed terms in the last line are non-negative, hence the last line is non-positive. Thus the only possible positive line is the fourth, and this is $O(1)$ since it is a sum over i, k for which $X_{i,k}(n) \leq K_0$. Thus (3.11) is $O(1)$. The rest of the details are as for the case where all $X_{i,k}(n) \geq K_0$. ■

Comments. The rule (2.4) requires that each node k know the value of the $X_{i,k}(n)$ and $X_{i,f(i,k)}(n)$ for all i that use node k. It is easily seen that the value of $X_{i,f(i,k)}(n)$ need only be known approximately at node k. Suppose that node k knows it subject to a bounded error, random or not. Then the proof still goes through under the same conditions. So, we need only have an occasional approximate estimate of the queues at the upstream nodes. The theorem asserts stability. But the quantities q_0, δ, c_1 all depend on the values of the sums in (A2.2) and (A2.3) and on the excess capacity of the system as quantified by c_0. If the rate of mixing of the channel process is very slow, then the queues will often have very large excursions, despite the fact of stability. The channel state can have "fast" and "slow" components, provided that (A2.3) hold. Suppose that no information is ever available on a channel. Then its state is constant, and the probabilistic information that is available will be the basis of the coding. Consider the possibility that some links are preempted by priority users from time to time, where the intervals of availability are defined by a renewal process that is independent of the arrival and channel rate processes. Then it can be shown that the results continue to hold, but with the $\bar{q}_{i,k}$ multiplied by the fraction of time that the channel is available, so the capacity must be sufficient to handle the down times. See also the comments concerning non-unique routes in the next section.

Under the other assumptions, (A2.4) is sufficient but not necessary for stability, but it is "nearly" necessary in the following sense. Suppose that for each allowable choice of the $\{\tilde{q}^j_{i,k}\}$, there is some (i_0, k_0) such that $\bar{q}_{i_0,b(i_0,k_0)} - \bar{q}_{i_0,k_0} > 0$. Then the system is not stable.

4 Extensions

The basic idea has useful extensions in many directions. Only a few will be described, in order to illustrate some of the possibilities. It is also possible to model situations where the number of users and destinations vary randomly or routes change randomly.

1. Acknowledgments of packet receipt required. The discussion up to this point supposed that if a packet were lost, then it would not be retransmitted, an assumption that is common in discussions of ad-hoc networks (eg., [6]). Suppose that packets that are not acknowledged within an appropriate interval must be retransmitted. We can include in such non-acks packets that are believed to have too many decoding errors. In the most general case, the loss process for packets leaving any given node will depend on the traffic in the channels that are travelled, the channel characteristics, the contents of the upstream buffers and the delays. Taking all of this into account in the analysis is very difficult. Because of this one often supposes that the loss is largely a consequence of uncontrolled traffic and imposes a predetermined loss model.

Starting from the last observation, we will take the following common approach. Acks of received packets at the destination are sent through the network to the source node. It is supposed that these are always received. (Otherwise, the packet is assumed to be lost.) We are assuming that acks go to the origin node. An alternative is to suppose that transmissions on all links must be acknowledged. The development is essentially the same for both cases. We use the origin node since the notation is simpler. In particular, if an ack for a source i packet, originating at node $k = k(i)$, is not received there within a (source-destination-dependent) delay of $W_{i,k}$ scheduling intervals, then it must be retransmitted. The development in Section 3 is readily modified to account for the losses and ack requirement. [6]

Let $\zeta_{i,k}(n)$ denote the fraction of the packets transmitted from (i, k) at time $n - W_{i,k}$ that were not acknowledged by time n and must be readded to queue (i, k) and retransmitted. This addition takes place *during* the nth slot, and so

[6]One could use a more detailed model that accounts for the reduction in the intermediate queues due to a loss. In effect, we obtain an upper bound to the necessary increase in capacity.

the $\zeta_{i,k}(n)$ are not known until time $n+1$. If $k \neq k(i)$ then set $\zeta_{i,k}(n) = 0$. Suppose that the loss process (i.e, non-receipt of an ack) within the desired windows is random. The ack processes are thus mutually independent and independent of the channel states, queue lengths, decisions, and arrivals, and iid for the packets from each source. Augment $\mathcal{F}_n$ by adding the ack/no-ack processes up to but not including time n. Thus it measures $\{\zeta_{i,k}(l); i, k, l < n\}$ but not $\zeta_{i,k}(n)$. Define $p_{i,k} = E_n\zeta_{i,k}(n) = E\zeta_{i,k}(n)$. Thus $p_{i,k} = 0$ if $k \neq k(i)$. It is useful to keep the k subscript to simplify the representations of various sums that will appear.

Taking into account the packets that need to be retransmitted, the queue dynamics are

$$X_{i,k}(n+1) = X_{i,k}(n) + a_{i,k}(n) - d_{i,k}(n) + d_{i,b(i,k)}(n) + d_{i,k}(n - W_{i,k})\zeta_{i,k}(n),$$

$$\begin{aligned} &X^p_{i,k}(n+1) = X^{p-1}_{i,k}(n)\left[-d_{i,k}(n) + d_{i,b(i,k)}(n) + d_{i,k}(n-W_{i,k})\zeta_{i,k}(n)\right] \\ &+X^{p-1}_{i,k}(n)a_{i,k}(n) + \text{terms of order } (p-2). \end{aligned} \tag{4.1}$$

The new consideration is the $d_{i,k}(n - W_{i,k})\zeta_{i,k}(n)$ term and this must be accounted for in the construction of the perturbation. The new component of the perturbation is

$$\delta V^W(n) = \sum_{i,k} p_{i,k} w_{i,k} X^{p-1}_{i,k}(n) \sum_{m=n-W_{i,k}}^{n-1} d_{i,k}(l). \tag{4.2}$$

This component will help us deal with averaging the increases in the source queue due to acks not arriving in time. Since $E_n\zeta_{i,k}(n) = p_{i,k}$, we can write

$$\begin{aligned} &E_n[V(X(n+1)) - V(X(n))] + E_n[\delta V^W(n+1) - \delta V^W(n)] \\ &= \sum_{i,k} w_{i,k} X^{p-1}_{i,k}(n)\left[E_n a_{i,k}(n) - d_{i,k}(n) + d_{i,b(i,k)}(n) + p_{i,k} d_{i,k}(n - W_{i,k})\right] \\ &\quad + \sum_{i,k} p_{i,k} w_{i,k} X^{p-1}_{i,k}(n)\left[d_{i,k}(n) - d_{i,k}(n - W_{i,k})\right] \\ &\quad + \text{terms of order } (p-2), \end{aligned} \tag{4.3}$$

where the second line is the dominant (highest order) term in $E_nV(X(n+1)) - V(X(n))$, and the third line is the dominant term from the expansion of $E_n[\delta V^W(n+1) - \delta V^W(n)]$. Note that the terms with $d_{i,k}(n - W_{i,k})$ in the second and third lines cancel each other. This was the motivation for the form of the perturbation component (4.2).

The appropriate decision rule is (2.4), (2.5), or (2.6) with $d_{i,k} w_{i,k} X^{p-1}_{i,k}$ multiplied by $1 - p_{i,k}$: I.e., replace (2.4) by

$$\max_{\{d_{i,k}(n):i\}} \sum_i \left[w_{i,k} X^{p-1}_{i,k}(n) d_{i,k}(n)(1 - p_{i,k}) - w_{i,f(i,k)} X^{p-1}_{i,f(i,k)}(n) d_{i,k}(n) \right].$$

For use below, note that, if k is the origin node for source i, then $d_{i,b(i,k)}(n) = 0$. Define the new perturbed Liapunov function

$$\tilde{V}^W(n) = V(X(n)) + \delta V^W(n) + \sum_{i,k} \delta V^a_{i,k}(n) + \sum_{i,k,j} (1-p_{i,k}) \delta V^{d,+}_{i,k,j}(n) + \sum_{i,k,j} \delta V^{d,-}_{i,k,j}(n), \tag{4.4}$$

where we recall that $p_{i,k} = 0$ if $k \neq k(i)$. Then, using (4.3), (3.5), (3.8), and (3.9),

$$\begin{aligned}
&E_n \tilde{V}^W(n+1) - \tilde{V}^W(n) = \sum_{i,k} w_{i,k} X^{p-1}_{i,k}(n) \bar{\lambda}^a_{i,k} \\
&\quad + \sum_{i,k} \left[-(1-p_{i,k}) w_{i,k} X^{p-1}_{i,k}(n) d_{i,k}(n) + w_{i,k} X^{p-1}_{i,k}(n) d_{i,b(i,k)}(n) \right] \\
&\quad + \sum_{i,k,j} (1-p_{i,k}) w_{i,k} X^{p-1}_{i,k}(n) \tilde{q}^j_{i,k} \left[I_{\{L_k(n)=j\}} - \Pi_{k,j} \right] \\
&\quad - \sum_{i,k,j} w_{i,k} X^{p-1}_{i,k}(n) \tilde{q}^j_{i,b(i,k)}) \left[I_{\{L_{b(i,k)}(n)=j\}} - \Pi_{b(i,k),j} \right] \\
&\quad + \text{terms of order } (p-2).
\end{aligned}$$

The second line is due to (the non-arrival parts of) the second and third lines of (4.3). The third line is due to $\delta V^{d,+}_{i,k,j}(n)$ and the fourth line to $\delta V^{d,-}_{i,k,j}(n)$. Dominating terms as in the part of the proof of the theorem concerning (3.11) yields the following upper bound to the last expression:

$$\begin{aligned}
&\sum_{i,k} w_{i,k} X^{p-1}_{i,k}(n) \left[-(1-p_{i,k}) \bar{q}_{i,k} + \bar{q}_{i,b(i,k)} \right] \\
&+ \text{terms of order } (p-2),
\end{aligned}$$

where we define $\bar{q}_{i,b(i,k(i))} = \bar{\lambda}^a_i$, where $k(i)$ is the origin node for source i. Keep in mind that $p_{i,k} > 0$ only for $k = k(i)$.

The net effect of the loss if packets is: for the same input rates, the channel along the path for source i must be able to handle a mean rate flow that is increased by a factor of $1/(1 - p_{i,k(i)})$.

The assumed independence of the queue-length processes and the process of node $k(i)$ not receiving an ack from the final destination for source i, within the $W_{i,k(i)}$ intervals, is questionable. But the same approach can be used for the case where acks for each link are required. Then (A2.4) is modified to require that, for each (i,k), $(1 - p_{i,k(i)})\bar{q}_{i,k(i)} > \bar{\lambda}^a_i$, and, for $k \neq k(i)$, $(1 - p_{i,k})\bar{q}_{i,k} \geq (1 - p_{i,b(i,k)})\bar{q}_{i,b(i,k)}$, where $p_{i,k}$ is now defined to be the probability of loss of a packet on the link from node k to node $f(i,k)$ and the losses on the individual inks are mutually independent.

2. Non-unique routes. Up to now, we have supposed that the route from source to destination is unique. The results are readily extendable to the case where several different routes can be used. The main differences are notational. If node k is on some route for source i, then packets in queue (i,k) might be sent to and/or received from several different nodes. The term $d_{i,k}(n)$ previously denoted the amount to be sent from queue i at node k, to the unique upstream node $f(i,k)$. Replace $d_{i,k}(n)$ and $f(i,k)$ by $d_{i,k,\alpha}(n)$ and $f(i,k,\alpha)$, resp., where the index α denotes the canonical upstream node and takes values in a set that depends on (i,k) and is known at node k. Analogously, replace $b(i,k)$ by $b(i,k,\beta)$, where β indexes the possible nodes that can transmit to queue (i,k). The index β takes values in a set that depends on (i,k) and is known at node k.

Define $u_{i,k,\alpha}(\cdot)$, $\tilde{u}_{i,k,\alpha}(\cdot)$, $g_{i,k,\alpha}(\cdot)$ and $\tilde{q}^j_{i,k,\alpha}$ analogously to the definitions without the α. Replace (2.4) by

$$\max_{\{d_{i,k.\alpha}(n);i,\alpha\}} \sum_{i,\alpha} \left[w_{i,k} X^{p-1}_{i,k}(n) - w_{i,f(i,k,\alpha)} X^{p-1}_{i,f(i,k,\alpha)}(n) \right] d_{i,k,\alpha}(n),$$

subject to the constraints at node k. Condition (A2.4) needs to be modified as

$$\bar{q}_{i,k} = \sum_{i,k,\alpha} q^j_{i,k,\alpha} \Pi_{k,j} > \bar{\lambda}^a_i.$$

$L_k(n)$ still denotes the set of channel states of all possible links out of node k. For each node k and its channel state vector value j, the decision rule will make a unique (or indifferent) allocation of resources among the possible outgoing links. The method of Section 5 for getting the a priori routes might yield multiple routes for some source-destination pairs.

An important case where non-unique routes are useful is where the links can break down, be preempted by priority users, or be severely impaired from time to time. A special case is that of random connectivity as in [2, 3]. Let us incorporate these variations in the process $L_k(\cdot)$ and suppose that (A2.3) holds. Then, under the other conditions, appropriately modified as noted above, the stability results hold.

3. Multicasting. Suppose that some sources have multiple destinations, with a unique route for each source-destination pair. Let the route network for each source form a tree, with the source as the root and the final destinations as the end branches. Suppose that if the tree branches at node k, then transmissions must be done to all of the branches simultaneously. Redefine $f(i,k,\gamma)$ to index the forward nodes for queue (i,k), where γ indexes the next nodes in the tree. Then (2.4) is replaced by

$$\max_{\{d_{i,k}(n):i\}} \sum_{i,\gamma} \left[w_{i,k} X^{p-1}_{i,k}(n) - \sum_{\gamma} w_{i,f(i,k,\gamma)} X^{p-1}_{i,f(i,k,\gamma)}(n) \right] d_{i,k}(n),$$

subject to the constraints at node k. Modify (2.5) and (2.6) analogously. The criterion (A2.4) is modified in an obvious manner to take account of the new flows.

4. Randomly available frequencies: Opportunistic frequency allocation. Consider the problem where there is a set (say, F) of available carrier frequencies, and these must be assigned to the mobiles in a way that precludes interference between other nodes transmitting at the same frequencies. This requires a centralized control. Of particular interest is the so-called "opportunistic" frequency scheduling, where several frequencies have a priority assignment to other users, but they are available to the controlled users when not required by the priority ones. They will be available on random intervals. The frequency availability process could be incorporated into the definition of the channel processes $\{L_k(\cdot), k\}$ and (2.5) or (2.6) used. The frequency assignments will vary randomly due to the random periods of availability. The perturbed Liapunov function method can be used to "average" the frequency assignments and use the associated mean approximation to prove that the rules (2.5) or (2.6) are stabilizing. Our main interest is the illustration of the definition and use of the perturbations. So, for simplicity, let us suppose that a node can be assigned at most one carrier frequency in each slot and that it, in turn, can be assigned to its links, it in any way that is consistent with the coding and modulation limitations. The frequency assignments are made a priori in some fair way, and each node knows well in advance if it will be assigned a particular frequency at a specific time, if that frequency will be available. The availability is the random part. Hence the a priori frequency assignments can be supposed to be independent of the channel and queue values. If a node cannot use an assigned frequency due to empty queues at that time, then it can be reassigned to another node.

Let us spell this out in a little more detail. Let $\psi_k^m(n)$ denote the indicator function of the event that frequency m has been a priori assigned to node k for use at time n. First suppose that there are $\bar{\psi}_k^m$, the average fraction of time that frequency m is available and is assigned to node k, such that the sums

$$\sum_{l=n}^{\infty} E_n \left[\psi_k^m(l) I_{\{L_k(l)=j\}} - \bar{\psi}_k^m \Pi_{k,j} \right]$$

are well defined and bounded uniformly in n, ω. Functions $g_{i,k}^m(\cdot)$ and $u_{i,k}^m(\cdot)$ are defined analogously to $g_{i,k}(\cdot)$ and $u_{i,k}(\cdot)$, resp., and are the outputs and controls, resp., when frequency m is assigned to (i,k). Suppose that there are allowable outputs $\{\tilde{q}_{i,k}^{m,j}; i,k,m,j\}$ such that Condition (A2.4) holds with the definition $\bar{q}_{i,k}(X) = \sum_{j,m} \tilde{q}_{i,k}^{m,j} \Pi_{j,k} \bar{\psi}_k^m$,

Replace the set of perturbations $\{\delta V_{i,k,j}^{d,+}(n); i,j,k\}$ by a set $\{\delta V_{i,k,j}^{d,m,+}(n); i,j,k,m\}$, where

$$\delta V_{i,k,j}^{d,m,+}(n) = w_{i,k} X_{i,k}^{p-1}(n) \tilde{q}_{i,k}^{j,m} \sum_{l=n}^{\infty} E_n \left[\psi_k^m(l) I_{\{L_k(l)=j\}} - \bar{\psi}_k^m \Pi_{k,j} \right],$$

with analogous replacements for the set $\{\delta V_{i,k,j}^{d,-}(n); i,j,k\}$. Then it can be shown that the theorem continues to hold.

5 An A Priori Routing Selection

Various approaches to getting the a priori routes have been suggested; for example, minimal hop routes, selecting the next node to be the one closest to the destination, or averaging the channels and then selecting a minimum power route. None of these account for the random variations in the channels nor are sufficiently sensitive to the total power requirements. A potentially useful approach for getting the routing and the $\tilde{u}(\cdot)$ functions is based on a type of fluid controlled-flow approximation. In applications the algorithm would be run periodically to produce acceptable routings as conditions change. For the purpose of of exposition we concentrate on one simple situation. The example is intended to be illustrative of the possibilities only.

Suppose that power only is to be allocated and that any received packet must have a given minimum signal to noise ratio. We allow that the routing for each source-destination pair is not necessarily unique and they might depend on the channel states. The development below gets a routing. But, given the routing, the decision rules (2.4), (2.5), or (2.6) are used and for each (i,k,j,X) will yield a unique choice of outgoing link, subject to random selection due to ties. Since we are concerned only with getting the routings, we need only deal with flows when all the queue levels are large.

Let $q^j_{i,k,m}$ denote the number of packets scheduled to be sent per slot from queue (i,k) to queue (i,m) at node m when the channel state is j. Suppose that there are upper bounds Q_k such that for each j,k,

$$\sum_{i,m} q^j_{i,k,m} \le Q_k. \tag{5.1}$$

This constraint reflects the fact that each packet takes a fixed time, and the slot duration is fixed. Suppose that, for the connection from queue (i,k) to node m under channel state j, each packet must have $p^j_{i,k,m}$ units of energy to attain the minimum required S/N ratio at the receiver at m. Suppose that each node k has a constraint of the form

$$\sum_{i,m} p^j_{i,k,m} q^j_{i,k,m} \le P_k, \quad \text{each } j, \tag{5.2}$$

where P_k is the total energy/slot available at node k.[7] [8]

We also need a constraint that assures that the average output for each non-source node equals the average input, and we write this as follows, for each i,k:

$$\overline{\text{out}} = \sum_{m,j} q^j_{i,k,m}\Pi_{k,j} \ge \sum_{l,j} q^j_{i,l,k}\Pi_{l,j} = \overline{\text{in}}\,. \tag{5.3}$$

[7] If the constraint is over average power, then use $\sum_{i,m,j} p^j_{i,k,m} q^j_{i,k,m}\Pi_{k,j} \le P_k$.

[8] Since $q^j_{i,k,m}$ might be positive for more than one value of m for some (i,k,j), the functions $u_{i,k}(\cdot)$ and $g_{i,k}(j,X,u_{i,k})$ and the $\tilde{u}_{i,k}(\cdot), \tilde{q}^j_{i,k}$ of (A2.4) are replaced by $u_{i,k,m}(\cdot)$ and $g_{i,k,m}(j,X,u_{i,k,m})$, and $\tilde{u}_{i,k,m}(\cdot), \tilde{q}^j_{i,k,m}$, resp. Then, for large queue levels we can suppose that $\tilde{q}^j_{i,k,m} = \tilde{u}_{i,k,m}(j,X) = g_{i,k,m}(j, X_{i,k}, \tilde{u}_{i,k,m}(j,X)) = q^j_{i,k,m}$.

If node $k(i)$ is the input node for source i, then replace (5.3) by

$$\overline{\text{out}} = \sum_{m,j} q^j_{i,k(i),m} \Pi_{k(i),j} = \bar{\lambda}^a_i + \epsilon. \tag{5.4}$$

The (arbitrarily small) $\epsilon > 0$ is used to assure slight overcapacity so that (A2.4) will hold and the stability argument of Theorem 3.1 can be used. Suppose that $c(i)$ is the destination node for source i. Then to assure that all packets end up where they are intended, for each i use the constraint

$$\sum_{k,j} q^j_{i,k,c(i)} \Pi_{k,j} = \bar{\lambda}^a_i + \epsilon. \tag{5.5}$$

Any flows $q^j_{i,k,m}$ that satisfy the constraints (5.1)–(5.5) will yield an acceptable a priori route. But it makes sense to select one via an optimization problem. One reasonable cost criterion is the total average power given by

$$\sum_{i,k,m,j} p^j_{i,k,m} q^j_{i,k,m} \Pi_{k,j}. \tag{5.6}$$

Minimize (5.6), subject to (5.1)–(5.5). The above approach to getting the a priori routes might yield a distributed flow for some sources. However, given these routes, the maximization rules (2.4), (2.5), or (2.6), still work. At any node k, we queue all of the packets for each source i together. Replace (2.4) by

$$\max_{\{d_{i,k,m}(n); i,m\}} \sum_i \left[w_{i,k} X^{p-1}_{i,k}(n) - w_{i,f(j,i,k,m)} X^{p-1}_{i,f(j,i,k,m)}(n) \right] d_{i,k,m}(n),$$

where for each $i, j, k,$ $f(j,i,k,m)$ indexes the links for which $q^j_{i,k,m} > 0$ and $d_{i,k,m}(n)$ is the amount sent to node m. The proof of Theorem 3.1 requires the slight modification discussed in the note concerning non-unique routes in the previous section.

For multicasting, simply use (5.5) for all destination nodes for source i.

Comment. In any application, to get feasible routes and $\tilde{q}^j_{i,k}$ or $\tilde{q}^j_{i,k,m}$, one must use the correct relations between the resources and the rates, and (5.2) and (5.6) appropriately modified, as might be required. The optimization criterion (5.6) is reasonable. However, for an alternative criterion that might be of interest, rewrite (5.3) as

$$\sum_{m,j} q^j_{i,k,m} \Pi_{k,j} - \sum_{l,j} q^j_{i,lk} \Pi_{l,j} = b_{i,k},$$

where $b_{i,k} > 0$. With appropriate definitions, this can be made to include (5.3) and (5.4). Then maximize $\sum_{i,k} b_{i,k}$, or, alternatively, seek $\max \min_{i,k} b_{i,k}$ where the max is over all feasible solutions. This approach will get routes and $\tilde{q}^j_{i,k,m}$ that yield the best c_0 in (A2.4). In addition, the dual variables associated with the constraints provide "price" guidelines, that tell us the places where an increase in the resources would do the most good (in the sense of the mathematical programming formulation). Constraints on the minimal distance between communicating nodes can be added if we are concerned with routes with too many hops.

Comment on bandwidth allocation. Suppose that the basic control is over bandwidth allocation, with the number of packets/slot being proportional to bandwidth as $q^j_{i.k,m} = b^j_{i,k,m} p^j_{i,k,m}$, where the $p^j_{i,k,m}$ are the constants of proportionality and $b^j_{i,k,m}$ is the assigned bandwidth.. There would be a total BW constraint of the form $\sum_{i,m} b^j_{i,k,m} \le B_k$ at each node, replacing (5.2). The input-output constraints (5.3), (5.4), and (5.5), are still to hold. To get the routes, one could minimize the total average bandwidth:

$$\sum_{i,k,m,j} b^j_{i,k,m} \Pi_{k,j}.$$

References

[1] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, R. Vijayakumar, and P. Whiting. Providing quality of service over a shared wireless link. *IEEE Communications Magazine*, 2001.

[2] N. Bambos and G. Michailidis. Queueing and scheduling in random environments. *Adv. in Appl. Prob.*, 36:293–317, 2004.

[3] N. Bambos and G. Michailidis. Queueing dynamics of random link topology: Stationary dynamics of maximal throughput schedules. *Queueing Systems*, 50:5–52, 2004.

[4] R. Buche and H.J.Kushner. Control of mobile communication systems with time-varying channels via stability methods. *IEEE Trans on Autom. Contr.*, 49:1954–1962, 2004.

[5] R. Buche and H.J. Kushner. Analysis and control of mobile communications with time varying channels in heavy traffic. *IEEE Trans. Autom. Control*, 47:992–1003, 2002.

[6] F. Eshghi, A.K. Elhakeem, and Y.R. Shayan. Performance evaluation of multihop ad hoc WLANs. *The Communications Magazine*, 43:107–115, 2005.

[7] H.J. Kushner. *Stochastic Stability and Control*. Academic Press, New York, 1967.

[8] H.J. Kushner. *Approximation and Weak Convergence Methods for Random Processes with Applications to Stochastic Systems Theory*. MIT Press, Cambridge, Mass., 1984.

[9] H.J. Kushner and G. Yin. *Stochastic Approximation Algorithms and Applications*. Springer-Verlag, Berlin and New York, 1997. Second edition, 2003.

[10] S. Shakkoti and A. Stolyar. Scheduling for multiple flows sharing a time-varying channel: The exponential rule. In M Suhov, editor, *Analytic Methods in Applied Probability: In Memory of Fridrih Karpelevich, American Math. Soc. Transl. , Series 2, Volume 207*, pages 185–202. American Mathematical Society, Providence, 2002.

[11] V. Solo and X. Kong. *Adaptive Signal Processing Algorithms*. Prentice-Hall, Englewood Cliffs, NJ, 1995.

[12] S. Stolyar. Max weight scheduling in a generalized switch: state space collapse and workload minimization in heavy traffic. *Ann. of Appl. Probab*, 14:1–53, 2004.

[13] L. Tassiulas and A. Ephremides. Dynamic server allocation to parallel queues with randomly varying connectivity. *IEEE Trans. Automatic Control*, 39:466–478, 1993.