Fair Resource Allocation in Wireless Networks using Queue-length-based Scheduling and Congestion Control

Atilla Eryılmaz and R. Srikant

Abstract— **We consider the problem of allocating resources (time slots, frequency, power, etc.) at a base station to many competing flows, where each flow is intended for a different receiver. The channel conditions may be time-varying and different for different receivers. It is well-known that appropriately chosen queue-length based policies are throughput-optimal while other policies based on the estimation of channel statistics can be used to allocate resources fairly (such as proportional fairness) among competing users. In this paper, we show that a combination of queue-length-based scheduling at the base station and congestion control implemented either at the base station or at the end users can lead to fair resource allocation and queue-length stability.**

Index Terms— **Throughput-optimal scheduling, Congestion control, Proportional fairness, Wireless networks**

I. Introduction

THE wireless channel is a shared medium over which many users compete for resources. Since there are many users, it is important to allocate this shared resource in a fair manner among the users. Further, since the available spectrum is limited, it is also important to efficiently use the channel. However, the time-varying nature of the wireless environment, coupled with different channel conditions for different users, poses significant challenges to accomplishing these goals. Moreover, the lack of availability of channel and arrival statistics further complicates the solution.

We will primarily restrict our attention to the downlink in cellular networks where there is a base station that allocates resources to many competing users. We assume that the packets destined for the different receivers are stored in separate queues. The scheduler at the base station is responsible for allocating resources to the different queues as a function of the current channel conditions as well as the queue lengths. Prior work on this problem can be largely classified into two main categories:

- *Throughput-optimal scheduling*: Here it is assumed that the mean arrival rates of the packets into each queue lie within the capacity region (the set of sustainable arrival rates) of the channel. However, neither the actual arrival rates nor the channel capacity region is assumed to be known. The scheduler is allowed to know the current queue lengths and the current channel conditions. It was then shown in [33] that allocating resources to maximize a queue-length-weighted sum of the rates (which are feasible in the current time slot) is a stabilizing policy. This result was then generalized in many different directions in [3], [29], [28], [10], [4], [20], [9], [25]. Such policies are called *throughput optimal* since the queues are stable under any sustainable throughput.
- *Fair Scheduling:* An obvious drawback of throughput-optimal policies is that no traffic policing is enforced. For instance, if one or more sources misbehave and increase their arrival rates so that the set of arrival rates lies outside the capacity region, then the system becomes unstable. In other words, all flows will be penalized due to the behavior of a few misbehaving flows. Thus, an alternative is to provide some degree of flow isolation at least in the long term, by allocating resources in a fair manner to the various queues. A commonly-used framework for such allocation is the concept of proportional fairness [13]. It was shown in [34] that proportional fairness can be achieved in TDMA cellular networks by scheduling the user which has the largest ratio of the achievable data rate at the current instant to the average rate that it has been allocated so far. The properties of such a policy have been studied empirically in [12] and analytically in [31], [18], and a multiple antenna implementation of such an algorithm over slowly time-varying channels has been proposed in [35]. Related work on channel-state-aware scheduling in wireless networks can also be found in [1], [8], [22], [7], [23].

From an applications point of view, throughput-optimal scheduling as described above is more suitable for inelastic traffic where the sources do not adapt their transmission rate based on congestion in the network. In this case, admission control is required to ensure that the arrival rates lie within the capacity region of the network and further, in the case of wireless networks, due to the time-varying nature of the network, an appropriate scheduling algorithm is required to ensure that the network can stably serve the admitted traffic. On the other hand, fair scheduling is more suited for elastic traffic sources which can adjust their traffic rates in response to feedback from the network regarding the network conditions. Without such a rate-control mechanism, fair scheduling would either lead to under utilization (when a traffic source is not generating enough data to make use of the bandwidth allocated to it) or packet losses or large delays (when a traffic source is generating data at a much

Research supported by a Vodafone Fellowship, DARPA Grant F30602-00-2-0542 and AFOSR URI F49620-01-1-0365.

A. Eryılmaz and R. Srikant are with the Department of Electrical and Computer Engineering and the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, IL, 61801, USA. E-mail: {eryilmaz,rsrikant}@uiuc.edu .

larger rate than the rate allocated to it by the base station).

In this paper, we are interested in allocating resources to elastic sources whose utilities are described by concave functions. Specifically, user i derives a utility $U_i(a_i)$ when it transmits at rate a_i. For ease of exposition, we consider utility functions of the form $U_i(a_i) = \beta_i \log a_i$, where β_i is some fixed weight, which can be different for different users. Thus, we consider weighted proportionally fair resource allocation; however, our results are easily generalizable to more general utility functions as we will comment later. We assume that congestion information is conveyed to the sources by putting the corresponding congestion price in the ACK packets. Each source react to its congestion price by choosing its transmission rates such that its marginal utility ($U_i'(a_i)$) is equal to the congestion price. We take the queue length at the base station to be the congestion price. In the Internet context, this is a special case of the dual algorithm proposed in [24], [38], [26]. In wireline networks, this interpretation of queue length (or delay) as the congestion price naturally arises from an convex optimization perspective where the resource constraints are linear [30]. However, in wireless networks, this interpretation is not immediately obvious since the resource constraints are not necessarily linear. Despite this, we show that the dual algorithm at the sources, along with queue-length-based scheduling at the base station, can be used to approximate weighted proportional fairness arbitrarily closely, where the approximation depends on the choice of a certain parameter used in the congestion control algorithm. Instead of the dual algorithm other algorithms such as the primal algorithm [14], [17] and the primal-dual algorithm [37], [2] can also be used. For a comprehensive description of the many algorithms that can be used to solve the resource allocation problem, see [30]. In this paper, we restrict ourselves to the dual algorithm.

The advantage of such an approach is that the base station does not have to choose the appropriate fairness criterion, but it is rather determined by the needs of the end users. On the other hand, if the end users were to misbehave, then the base station has no way of isolating the misbehaving user from the others (much like TCP in the Internet without any policing at the routers). However, if one is interested in providing long-term flow isolation as in the proportionally fair scheduler in [34], then our algorithm suggests an alternate implementation of the proportionally-fair algorithm which will be discussed in a later section.

The paper is organized as follows. Section II describes the model, including the scheduling and the congestion control algorithms. In Section III, we first analyze a heuristic continuous-time, deterministic fluid model of the system and then use the intuition thus obtained to study the original stochastic system model. Generalizations and implementation considerations are discussed in Section IV. Various simulation results are presented in Section V. Finally, conclusions are drawn in Section VI.

II. System Model

We consider a cellular network shared by n flows in the downlink. Figure 1 depicts the combination of the congestion controller and the base station from the perspective of a single flow. We assume that time is slotted and denote the size of the i^{th} queue at the beginning of the time slot t by $x_i[t]$, the number of arrivals to queue i in time slot t by $a_i[t]$, and the amount of service offered to queue-i in slot t by $\mu_i[t]$. We assume that each of these parameters can only take non-negative integer values. The evolution of the size of the i^{th} queue is given by

$$x_i[t+1] = (x_i[t] + a_i[t] - \mu_i[t])^+, \qquad i = 1, \cdots, n$$

where $(y)^+$ is equal to y if $y > 0$, and zero otherwise.

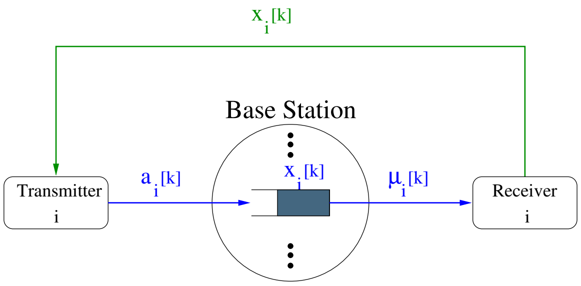

Fig. 1. A pictorial depiction of the system.

We assume that the channel between the base station and the receivers can be in one of J states in a given slot. We use $s[t]$ to denote the state in time slot t. The channel state is assumed to be fixed within a time slot, but may vary from one slot to another, and hence captures the time-varying characteristics of a fading environment. Corresponding to each channel state, say j, is an *achievable rate region*, $\mathcal{C}_j$, that is defined to be the convex hull of the rate vectors, $\vec{\eta}[t] := (\eta_1[t], \cdots, \eta_n[t])'$, that can be offered to the queues. We assume that each $\mathcal{C}_j$ is a bounded region, which is a reasonable assumption since in practice, power constraints and code designs allow only finite rate levels. We use $\hat{\eta} < \infty$ to denote the upper bound on the achievable rates for all channel states. The channel state process is assumed to be independent and identically distributed in each time slot (although it is straightforward to generalize our results to allow Markovian channel state processes), but we do not require that the statistics be known at the base station. We define the *mean achievable rate region* as

$$\bar{\mathcal{C}} := \{\vec{\eta} : \vec{\eta} = \sum_{j=1}^{J} \pi_j^{ch} \vec{\eta}^{(j)}, \vec{\eta}^{(j)} \in \mathcal{C}_j\},$$

where π_j^{ch} stands for the stationary distribution of the channel state process being in state j. The scheduler implemented at the base station is described as follows.

Queue-length-based Scheduler: In time slot t, given the current queue length vector, $\vec{x}[t] := (x_1[t], \cdots, x_n[t])'$, and the current channel state, $s[t]$, the scheduler chooses a service rate vector $\vec{\mu}[t] := (\mu_1[t], \cdots, \mu_n[t])' \in \mathcal{C}_{s[t]}$ that satisfies

$$\vec{\mu}[t] \in \arg\max_{\vec{\eta} \in \mathcal{C}_{s[t]}} \sum_{i=1}^{n} x_i[t] \eta_i. \qquad (1)$$

$\diamond$

A geometric interpretation of the above policy for two users is given in Figure 2, where the queue-length state space and the achievable service rate region are superimposed on each other. We observe that the allocated service rates always lies on the boundary of the current achievable rate region, and that if the rate vector $\vec{\mu}[t]$ satisfies (1) for some $\vec{x}[t]$, then it satisfies the same condition for all $\xi\vec{x}[t]$, for all $\xi > 0$. As mentioned earlier, this policy is known to be throughput-optimal for inelastic flows.

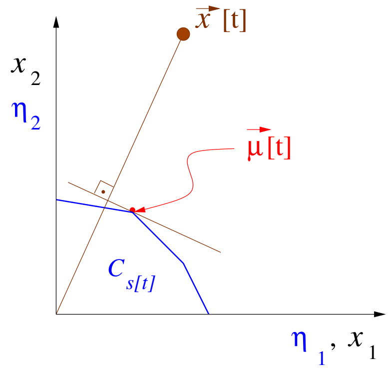

Fig. 2. Given $\vec{x}[t]$ and $s[t]$, the vector $\vec{\mu}[t]$ is chosen on the boundary of the current achievable rate region as in this figure.

The packet arrival rate into the queue is assumed to be congestion controlled according to the dual controller [24], [38]. We note that the dual controller is derived for the Internet applications where all users sharing a common resource can receive the same treatment from the router. However, here the situation is fundamentally different due the fact that the channel conditions for the different users can be different. One of the contributions of this paper is show that, somewhat surprisingly, the dual controller still leads to fair resource allocation provided that the base station uses the scheduler described above. Such a scheduling mechanism is not necessary in the Internet, but is crucial in the wireless network context studied here. We now describe the congestion controller.

CONGESTION CONTROLLER: For the i^{th} flow, given its current buffer occupancy $x_i[t]$, the data generation rate in slot t, equal to $a_i[t]$ in our notation, is a random variable satisfying the following:

$$E[a_i[t] \mid x_i[t]] = \frac{\alpha_i K}{x_i[t] + \gamma_i}, \qquad (2)$$
$$E[a_i^2[t] | x_i[t]] \le A < \infty, \qquad \forall x_i[t]$$

where $\{\gamma_i\}, \{\alpha_i\}$ and A are positive constants. We also assume $a_i[t]$ is independent across time slots. (Again, it is straightforward to generalize this assumption to allow dependence in the arrivals across time slots.) $\diamond$

We have allowed $a_i[t]$ to be a random variable to allow for various sources of randomness in the actual implementation (e.g., window-based implementations of congestion control) that are not precisely modelled here. Further, we have used $\gamma_i > 0$ (which can be arbitrarily small) to ensure that the arrival rate is bounded when $x_i[t] = 0$. There are other ways to ensure that the arrival rate is bounded (by imposing an upper bound on the arrival rate) and our results can be easily extended to these models. We note the well-known basic characteristics of a congestion controller from the above mechanism: the higher the congestion level, which is indicated by an increased level of buffer occupancy, the lower the data generation rate.

Notice that we have introduced a constant K in the congestion control algorithm. This corresponds to assuming that the utility function of user i is $K\alpha_i \log a_i$. Since K is the same for all users, this will not affect the relative resource allocation among the users. However, this constant plays a crucial role in determining how well we approximate weighted proportional fairness in a wireless network. Indeed, we will show that weighted proportional fair allocation is closely approximated for large K.

III. CHARACTERIZING THE SYSTEM PERFORMANCE

In this section, we will analyze the system described in the previous section. To accomplish this, we will start with a heuristic continuous-time fluid model and understand its behavior. Later, we will use these observations in the analysis of the original model and show that the original model behaves like the fluid model for large K.

A. Continuous-time Fluid Model

In the fluid-model, we assume that the channel state process is not random, but constant at its mean level. In other words, the achievable rate region is fixed at $\bar{\mathcal{C}}$. Also, the arrival rate is no longer a random variable, but is taken to be equal to its mean, i.e., $a_i(t) = \frac{\alpha_i K}{x_i(t)+\gamma_i}$. Here, (t), instead of $[t]$, is used to signify that time is a continuous variable. Then, the evolution of the i^{th} queue-length is described by

$$\dot{x}_i(t) = \left(\frac{\alpha_i K}{x_i(t) + \gamma_i} - \bar{\mu}_i(t)\right)^+_{x_i}, \qquad (3)$$

where $\vec{\mu}(t) \in \arg\max_{\vec{\eta}\in\bar{\mathcal{C}}} \sum_{i=1}^{n} x_i(t)\eta_i$ and $(y)^+_x$ is equal to y when $x > 0$ and is equal to $\max(y, 0)$ when $x \le 0$. In this formulation, the queue-length state space is assumed to be continuous. We now identify the set of service rates and queue lengths such that if the these queue lengths are chosen as the initial state and at each time instant, resources are allocated to achieve these service rates, then the queue lengths will remain at the initial conditions forever.

Definition 1—Invariant pair: The pair $(\vec{x}^\star, \vec{\mu}^\star)$ forms an *invariant pair* (more precisely, $\vec{\mu}^\star$ is an *invariant service rate vector* and $\vec{x}^\star$ is an *invariant queue-length vector*) if they satisfy both of the following conditions:

$$(D_1)\ \ \vec{\mu}^\star \in \arg\max_{\vec{\eta}\in\bar{\mathcal{C}}} \sum_{i=1}^{n} x_i^\star \eta_i,$$
$$(D_2)\ \ x_i^\star = \frac{\alpha_i K}{\mu_i^\star} - \gamma_i \qquad \forall i \in \{1, \cdots, n\}.$$

$\diamond$

Notice that, if at any time t' we have $\vec{x}(t') = \vec{x}^\star$ and $\vec{\mu}(t') = \vec{\mu}^\star$, then, due to (D_1) and (D_2), $\dot{x}(t)$ given by (3) will be zero for all $t \ge t'$. We now show that the invariant pair exists and is unique.

Proposition 1: [Existence and Uniqueness of $(\vec{x}^\star, \vec{\mu}^\star)$] An invariant pair of rates and queue lengths exists and is unique.

Proof: Note that the conditions on $\mu^\star$ given by (D_1) and (D_2) can be concisely written as

$$\sum_{i=1}^{n}(\mu_i^\star - \eta_i)\left(\frac{\alpha_i K}{\mu_i^\star} - \gamma_i\right) \quad \geq \quad 0 \qquad \forall \vec{\eta} \in \bar{\mathcal{C}}.$$

This is simply the condition for $\mu^\star$ to be an optimal solution to the following problem of maximizing a concave function over a convex set [6]:

$$\max_{\vec{\mu} \in \bar{\mathcal{C}}} \sum_{i=1}^{n} K\alpha_i \log \mu_i - \gamma_i \mu_i.$$

Since $\bar{\mathcal{C}}$ is a bounded set, clearly a solution exists to the above optimization problem. Further, since the objective is strictly concave the solution is unique. ∎

The above proof shows that the invariant point is simply the set of rates and queue lengths achieved under fair resource allocation, which is our goal. Next, we are interested in showing that the queue lengths described by the invariant point is attractive, i.e., all trajectories eventually converge to it.

Proposition 2: Starting from any initial queue-length vector, $\vec{x}(0)$, the queue-length vector $\vec{x}(t)$ eventually reaches $\vec{x}^\star$ as $t \to \infty$.

Proof: Consider the Lyapunov function

$$W(\vec{x}) = \frac{1}{2}\sum_{i=1}^{n}(x_i - x_i^\star)^2.$$

Next, we study the time derivative of this Lyapunov function at time t.

$$\begin{aligned}\dot{W}(\vec{x}(t)) &= \sum_{i=1}^{n}(x_i(t) - x_i^\star)\dot{x}_i(t)\\ &= \sum_{i=1}^{n}(x_i(t) - x_i^\star)\left(\frac{\alpha_i K}{x_i(t) + \gamma_i} - \bar{\mu}_i(t) + u_i(t)\right),\end{aligned}$$

where $u_i(t)$ is a non-negative quantity which denotes the wasted service given to queue i at time t. Note that $u_i(t) = 0$ whenever $x_i(t) > 0$, which allows us to write

$$\dot{W}(\vec{x}(t)) \quad \leq \quad \sum_{i=1}^{n}(x_i(t) - x_i^\star)\left(\frac{\alpha_i K}{x_i(t) + \gamma_i} - \bar{\mu}_i(t)\right).$$

Note that

$$\begin{aligned}&(x_i(t) - x_i^\star)\left(\frac{\alpha_i K}{x_i(t) + \gamma_i} - \bar{\mu}_i(t)\right)\\ &\qquad = (x_i(t) - x_i^\star)\left(\tfrac{\alpha_i K}{x_i(t) + \gamma_i} - \mu_i^\star\right) \quad (4)\\ &\qquad\quad + (x_i(t) - x_i^\star)(\mu_i^\star - \bar{\mu}_i(t)).\end{aligned}$$

Consider (4): if $x_i(t) > x_i^\star$, then

$$\frac{\alpha_i K}{x_i(t) + \gamma_i} - \mu_i^\star < 0,$$

which follows from the definition of $(\vec{\mu}^\star, \vec{x}^\star)$. Similarly, if $x_i(t) < x_i^\star$, then

$$\frac{\alpha_i K}{x_i(t) + \gamma_i} - \mu_i^\star > 0.$$

Therefore, unless $\vec{x}_i^\star = \vec{x}_i(t)$, the expression (4) is negative. When $\vec{x}_i^\star = \vec{x}_i(t)$, the expression (4) is zero.

Now consider the expression

$$\sum_{i=1}^{n}(x_i(t) - x_i^\star)(\mu_i^\star - \bar{\mu}_i(t)). \qquad (5)$$

We will now show that this expression is negative unless $\vec{x}^\star = \vec{x}(t)$. First we note the following two inequalities which follow from the definition of the invariant point and our scheduling policy which is the solution to the optimization problem (1).

$$\sum_{i=1}^{n} x_i^\star \mu_i^\star \quad \geq \quad \sum_{i=1}^{n} x_i^\star \bar{\mu}_i(t), \qquad (6)$$

$$\sum_{i=1}^{n} x_i(t)\bar{\mu}_i(t) \quad \geq \quad \sum_{i=1}^{n} x_i(t)\mu_i^\star. \qquad (7)$$

By adding both sides of the inequalities (6) and (7) and rearranging the terms, we get

$$\sum_{i=1}^{n}(x_i^\star - x_i(t))(\mu_i^\star - \bar{\mu}_i(t)) \quad \geq \quad 0. \qquad (8)$$

This shows that $(5) \leq 0$. Thus, we have

$$\begin{aligned}&\dot{W}(\vec{x}(t)) < 0, \quad \text{if } \vec{x}(t) \neq \vec{x}^\star, \text{ and}\\ &\dot{W}(\vec{x}(t)) = 0, \quad \text{if } \vec{x}(t) = \vec{x}^\star.\end{aligned}$$

The result follows from Lyapunov's global stability theorem [15]. ∎

B. Original System Model

In this section, we return to our original system model, where the arrivals and departures are integer-valued and random. We observe that the queue-length state vector $\vec{x}[t]$ evolves according to a discrete-time, discrete-space Markov chain. We first show that this Markov chain is stable, i.e., positive recurrent. The fact that the Markov chain is stable is not surprising since by our choice of congestion control, we have ensured that the mean arrival rate into a queue reduces when the queue length is large. However, the Lyapunov function used to establish stability can be used to obtain a useful upper bound on the Euclidean distance between the invariant queue-length vector $\vec{x}^\star$ and $\vec{x}[t]$ for large K and large t. This upper bound is then used to establish the properties of our joint congestion control-scheduling algorithm.

Theorem 1—Positive Recurrence: The Markov chain $\vec{x}[t]$ is positive recurrent.

Proof: It is easy to see that the Markov chain has a countable state space and is irreducible and aperiodic. Consider the Lyapunov function $W(\vec{x}) = \sum_{i=1}^{n}\frac{(x_i - x_i^\star)^2}{2}$. It is shown in the appendix that there exist positive constants δ, ζ and c such that

$$\begin{aligned}E[\Delta W_t(\vec{x})] &:= E[W(\vec{x}[t+1]) - W(\vec{x}[t]) \mid \vec{x}[t] = \vec{x}]\\ &\leq -\frac{\delta}{\sqrt{K}}\|\vec{x} - \vec{x}^\star\|\mathcal{I}_{\vec{x} \in \mathcal{D}^c} + \zeta \mathcal{I}_{\vec{x} \in \mathcal{D}}, \qquad (9)\end{aligned}$$

where $\mathcal{D} := \{\vec{y} : \|\vec{y}-\vec{x}^\star\| < c\sqrt{K}\}$ and $\mathcal{D}^c$ is the complement of $\mathcal{D}$ in $(\Re^+)^n$. Thus, we have a Lyapunov function that has negative conditional mean drift for all queue-length vectors that have sufficiently large magnitude. Now the result follows from Foster-Lyapunov criterion [5, Proposition 5.3]. ■

Inequality (9) is stronger than what is necessary to prove the stability of the Markov chain. However, it is useful in establishing the following theorem which characterizes the mean distance between $\vec{x}[t]$ for large t and the invariant queue-length vector $\vec{x}^\star$.

Theorem 2—Mean Distance Bound: There exists a positive constant $\bar{c}$, that depends on the mean achievable rate region, the algorithm parameters $\{\alpha_i, \gamma_i\}$, and the moments of the channel and arrival processes, such that

$$E\left[\|\vec{x}^\infty - \vec{x}^\star\|\right] \leq \bar{c}\sqrt{K}, \qquad \text{for large} K \tag{10}$$

where $\vec{x}^\infty$ is a notation used to denote the state of the Markov chain in steady-state and $\|\cdot\|$ denotes the Euclidean distance in the $\Re^n$.

Proof: We use (9) and the idea behind the proof of Pakes' lemma [27] to prove the theorem. We consider the following $T-$step mean drift. For any $\vec{y} \geq \mathbf{0}$,

$$\begin{aligned}
&E[W(\vec{x}[T])|\vec{x}[0]=\vec{y}] - W(\vec{y}) \\
&= \sum_{t=0}^{T-1} E[W(\vec{x}[t+1]) - W(\vec{x}[t])|\vec{x}[0]=\vec{y}] \\
&= \sum_{t=0}^{T-1} \sum_{\vec{x}\in\mathcal{Z}_+^n} \{P(\vec{x}[t]=\vec{x}|\vec{x}[0]=\vec{y}) \\
&\qquad \times E[W(\vec{x}[t+1]) - W(\vec{x}[t])|\vec{x}[t]=\vec{x}]\} \qquad (11)\\
&\leq \zeta \sum_{\vec{x}\in\mathcal{D}} \sum_{t=0}^{T-1} P(\vec{x}[t]=\vec{x}|\vec{x}[0]=\vec{y}) \qquad (12)\\
&\quad -\frac{\delta}{\sqrt{K}} \sum_{\vec{x}\in\mathcal{D}^c} \|\vec{x}-\vec{x}^\star\| \sum_{t=0}^{T-1} P(\vec{x}[t]=\vec{x}|\vec{x}[0]=\vec{y}),
\end{aligned}$$

where $\mathcal{Z}_+^n$ denotes the set of all non-negative n dimensional integer valued vectors. In the above derivation, (11) follows from the fact that $\vec{x}[t]$ is a Markov chain, and we have used (9) to get the inequality in (12). We note that for any $\vec{x} \in \mathcal{Z}_+^n$,

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} P(\vec{x}[t-1]=\vec{x}|\vec{x}[0]=\vec{y}) = \pi_{\vec{x}}^\infty,$$

for all $\vec{y}$, where $\vec{\pi}^\infty$ denotes the stationary distribution of the Markov chain $\vec{x}[t]$. Next, we move $W(\vec{y})$ to the other side of the inequality in (12), divide both sides by T, and let T go to infinity. This operation yields

$$0 \leq \zeta \sum_{\vec{x}\in\mathcal{D}} \pi_{\vec{x}}^\infty - \frac{\delta}{\sqrt{K}} \sum_{\vec{x}\in\mathcal{D}^c} \|\vec{x}-\vec{x}^\star\| \pi_{\vec{x}}^\infty.$$

Re-arranging the terms and with minor manipulations, this inequality can be written as

$$\frac{\delta}{\sqrt{K}} \sum_{\vec{x}\in\mathcal{Z}_+^n} \|\vec{x}-\vec{x}^\star\| \pi_{\vec{x}}^\infty \leq \sum_{\vec{x}\in\mathcal{D}} \left(\zeta + \frac{\delta}{\sqrt{K}}\|\vec{x}-\vec{x}^\star\|\right) \pi_{\vec{x}}^\infty$$
$$\leq (\zeta+\delta c) \sum_{\vec{x}\in\mathcal{D}} \pi_{\vec{x}}^\infty \leq (\zeta+\delta c),$$

where the second inequality follows from the definition of $\mathcal{D}$. Here, the left-hand-side is nothing but $\frac{\delta E[\|\vec{x}^\infty-\vec{x}^\star\|]}{\sqrt{K}}$. So we multiply both sides with $\frac{\sqrt{K}}{\delta}$ to get

$$E[\|\vec{x}^\infty - \vec{x}^\star\|] \leq \left(c + \frac{\zeta}{\delta}\right)\sqrt{K},$$

which completes the proof for $\bar{c} := \left(c + \frac{\zeta}{\delta}\right)$. ■

Notice that Theorem 2 implies, due the Markov's inequality, that $\forall \epsilon > 0$

$$P\left(\frac{1}{K}|x_i^\infty - x_i^\star| > \epsilon\right) \leq \frac{\bar{c}}{\epsilon\sqrt{K}}.$$

Therefore, the right-hand-side can be made arbitrarily small by choosing K sufficiently large. On the other hand, observe that $\frac{x_i^\star}{K} \to \frac{\alpha_i}{\mu_i^\star}$ as $K \to \infty$. Hence, in the above probabilistic sense, $\frac{x_i^\infty}{K}$ is approximately equal to $\frac{\alpha_i}{\mu_i^\star}$. We also know from Theorem 1, that $\vec{x}[t] \to \vec{x}^\infty$ as $t \to \infty$. Combining these arguments, we can expect a typical sample path of $x[t]$ to converge to a region around the vector $(\frac{\alpha_1}{\mu_1^\star}, \cdots, \frac{\alpha_n}{\mu_n^\star})$ when K is large. In Section V, we will present simulation results which will reinforce this observation.

Next, we address the implications of the above analysis on the delay and fairness characteristics of the system. It was observed in the above analysis that, for sufficiently large values of K, the queue-length vector will evolve towards the invariant point $\vec{x}^\star$ and stay relatively close to it in a probabilistic sense. Therefore, we can make the following approximation: $\vec{x}[t] \approx \vec{x}^\star$, when t becomes large. Consequently, the local scheduler described by (1) will make its decision at time t, roughly as

$$\vec{\mu}[t] \in \arg\max_{\vec{\eta}\in\mathcal{C}_{s[t]}} \sum_{i=1}^n x_i^\star \eta_i.$$

Thus, from the proof of Proposition 1, our proposed combination of the local scheduler with the end-to-end congestion controller guarantees stability and *weighted proportional fairness* among the flows. In Section V, we will present simulation results to support the above arguments.

Recalling Theorem 2 and the discussion following it, we can expect the mean of $\vec{x}^\infty$ to be very close to $\vec{x}^\star$. In fact, using a Lyapunov function of the form

$$L(\vec{x}) = \frac{1}{2}\sum_{i=1}^n x_i^2,$$

and using arguments very similar to the ones used in the proof of Theorem 2, we can prove that for large enough K, $E[\vec{x}^\infty] = \Theta(\vec{x}^\star)$.[1] Therefore, Little's Law suggests that the delays experienced by flow i is

$$E[\text{Delay}_i] = \frac{1}{\mu_i^\star}\Theta(x_i^\star) \approx \frac{\alpha_i}{(\mu_i^\star)^2}\Theta(K),$$

[1] $f(u) = \Theta(g(u))$ implies that for large enough u, there exist positive constants, $c_1 \leq c_2$, such that $c_1 g(u) \leq f(u) \leq c_2 g(u)$.

where the last step is true for large K. Hence, we can see a linear dependence of K on the mean delay experienced by the flows. Since we need large K to ensure a close approximation to weighted proportional fairness, this also implies large delays in the queues. This can be alleviated by a virtual queue implementation of the scheduling mechanism which will be discussed in the next section.

IV. Generalizations and Implementation Considerations

A. Other Schedulers and Utility Functions

Our fairness and stability results will continue to hold even if the basic algorithms are modified slightly. For example, the local queue-length-based scheduler can be modified to maximize

$$\vec{\mu}[t] \in \arg\max_{\vec{\eta} \in \mathcal{C}_{s[t]}} \sum_{i=1}^{n} f_i(x_i[t])\eta_i,$$

for appropriately chosen non-negative, increasing, unbounded set of functions $\{f_i(\cdot)\}$. The stabilizing properties of such schedulers have been addressed in [10], where it is also shown by simulations that the queue lengths of the different users can be controlled by choosing $\{f_i(\cdot)\}$ appropriately.

Another possible modification that we can allow is to change the type of congestion control mechanism implemented at the end users. For general increasing concave utility functions with the property $U(a) \to -\infty$ as $a \to 0$, user i's arrival rate should be modified as follows

$$E(a_i|x_i) = U_i'^{-1}(x_i)$$

with obvious modifications necessary to ensure that the arrival rate is bounded. In this case, instead of proportional fairness, the mean service rates μ_i will be allocated to maximize

$$\sum_i U_i(\mu_i).$$

Thus, instead of proportional fairness, one can modify the algorithm to achieve other forms of fairness.

B. Multi-hop Wireless Networks

The model analyzed so far focuses on cellular networks with a base station to perform the scheduling, i.e., the single-hop case. The results can also be extended to cover the multi-hop network case. Consider a multi-hop radio network where each user is identified by a origin-destination (O-D) pair and a collection of nodes describing the route between the O-D pair. The congestion controller is described in the same manner as before except that the congestion price will no longer be a single queue length, but will now be the sum of the queue lengths of all the queues in a user's path. However, the scheduler will now have to be a version of the multi-hop resource allocation described in [32], [25]. The algorithm can also be extended to the case where a collection (more than one) of routes exists between an O-D pair. The congestion control algorithm in this case would be the multi-path routing algorithm proposed in [14], [36], [21] along with the resource allocation algorithm in [25]. In principle, one can prove similar results as we have done in this paper for the multi-hop problem also, but the result is quite meaningless in the absence of a base station since then there is no central scheduler to solve the wireless resource allocation problem. The challenging problem of obtaining reasonable distributed solutions for the scheduling problem is still the hurdle whether one considers the throughput-optimal scheduling problem as in [32], [25] or the fair scheduling problem that we have considered here.

C. Reducing Delays Using Virtual Queues

As we have discussed earlier, one penalty for achieving user-defined fairness (as opposed to network-dictated fairness) is the possibility of large delays at the base station buffers. We can alleviate this problem by implementing the base station scheduler using virtual queues [11], [16], [19]. For each flow, the base station maintains a counter called the virtual queue. As an example, consider flow i. The virtual queue of flow i keeps track of a virtual queue length, where the virtual queue length of flow i is simply the length of a queue whose arrivals are the same as that of flow i, but whose service rate is always a fixed fraction $\beta < 1$ of the actual service rate. Therefore, the size of the virtual queue (denoted by $\tilde{x}_i$ for flow i) will always be larger than the actual queue-length x_i. The congestion feedback given to user i is $\tilde{x}_i[t]$ and therefore, user i will reduce its arrival rate well before its real queue builds up significantly. See Figure 3 for the model from flow i's perspective.

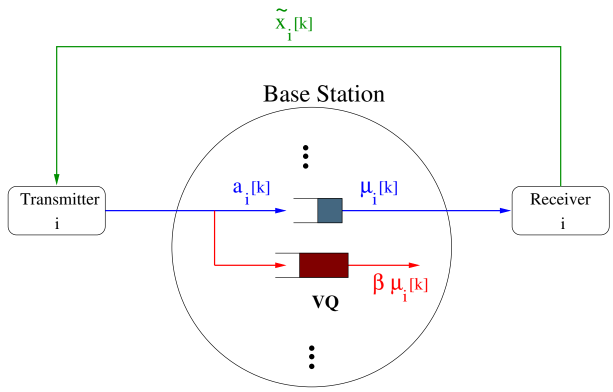

Fig. 3. The virtual queue implementation at the base station.

By choosing the β parameter appropriately the delay levels and the packet loss probabilities can be adjusted: the lower the β, the lower the actual queue lengths. However, there is a possible loss in throughput by choosing $\beta < 1$. In Section V, we will provide simulation results which show that, by choosing β close to 1, but not equal to 1, we can reduce the queue lengths dramatically while maintaining close to 100% throughput.

D. End-to-End versus Last-Hop Congestion Control

The advantage of end-user-implemented congestion control is that it allows each user to choose a congestion control algorithm based on its utility function. Thus, instead of the base station imposing a particular notion of fairness, the resource allocation truly reflects the needs (as defined by the user utility functions) of the users. Further, resource allocation based on a base station-defined notion of fairness could result in underutilization of the resources or large backlogs if the allocated data rate is not matched to the users' transmission rates.

An argument against congestion control at the end-user is that it does not provide for isolation among the flows. In other words, if a user misbehaves by transmitting at a much larger rate than is dictated by its congestion control algorithm, then there is no policing at the base station to prevent other users from experiencing poor quality-of-service. A base-station-defined fair resource allocation solves this problem by allocating resources to the users independent of the user's behavior. Here, we discuss a last-hop congestion control strategy, in place of the end-user congestion control algorithm, which can provide the same type of flow isolation albeit with the same drawbacks as described in the previous paragraph.

LOCAL IMPLEMENTATION STRATEGY: The base station imitates the effect of the end-to-end congestion controller by maintaining a two-stage queueing system. The queue at the first stage for each user is used to store packets arriving from the user. The contents of this first-stage are drained into another queue at a rate specified by the congestion control algorithm. The second-stage queues are the one used to implement the queue-length-based controller. Figure 4 depicts this model from flow i's perspective. $\diamond$

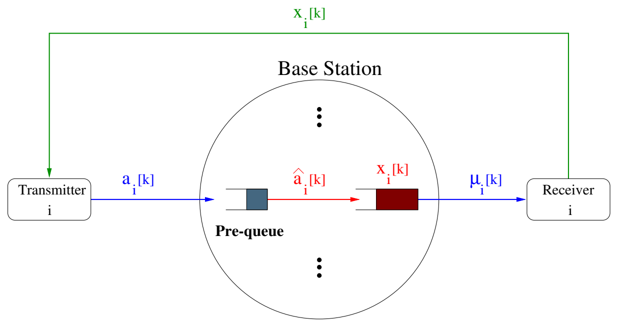

Fig. 4. With the use of a pre-queue, the number of packets arriving at the second stage queue, denoted in the figure by $\hat{a}_i[t]$, can be chosen as a function of $x_i[t]$ with mean $\frac{\alpha_i K}{x_i[t]+\gamma_i}$, thus mimicking the behavior of the congestion controller.

In the original implementation of proportional fairness given in [34], [35], the base station must decide on a time window over which the average rate allocated to each user is computed. This parameter is now replaced by the choice of the parameter K in our algorithm. It is interesting to note that both parameters directly impact the ability of the respective algorithms to precisely mimic weighted proportional fairness.

V. SIMULATION RESULTS

In this section, we provide simulation results to complement the analysis in the previous sections. We present results for a simple two-user scenario since the observations are no different in the case of multiple users. We make the reasonable assumption that the channels between the base station and each of the users fade independently. The base station is only allowed to serve a single queue in a given slot. For this scenario, the scheduling algorithm (1) is equivalent to serving the user that solves

$$i^*[t] = \arg\max_{i=1,2} \eta_i x_i[t],$$

at the rate $\eta_{i^*}[t]$. Ties are broken randomly. We let the rate vectors, $\eta_1[t]$ and $\eta_2[t]$, be Poisson distributed with means 0.5 and 1, respectively. Also, the number of arrivals in each slot to each queue is Poisson distributed with the mean determined by the congestion controller. We take $\alpha_i = 10$ and $\gamma_i = 1$ for all i.

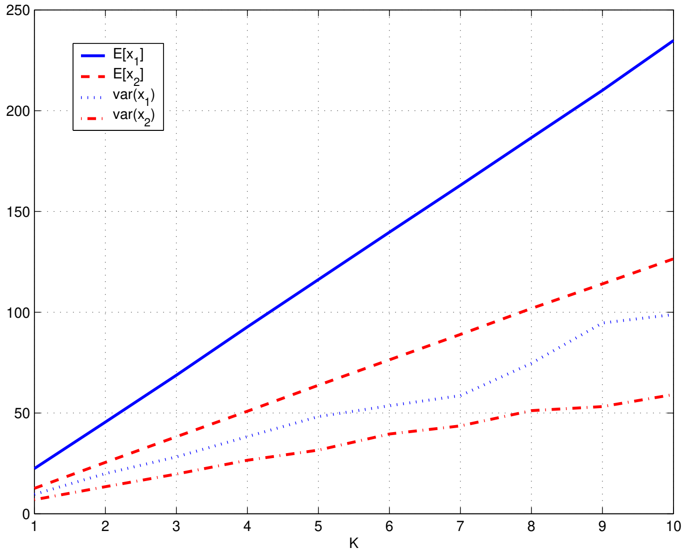

Fig. 5. Empirical change in the mean and variance of users' queue-lengths with K.

The effect of K on the mean and variance of the queue-length values in Figure 5. We observe the linear increase in both the means and the variances of the queue lengths with increasing K. Although we did not explicitly give a bound on the variance (which can be done by choosing a different Lyapunov function), our earlier analysis showed that $\vec{x}[t]$ will fluctuate around $\vec{x}^\star$ with the expected difference in the mean of the order of $\sqrt{K}$.

In Figure 6, we plot the average service rates provided to the two queues for different K values. We observe that $\mu_1^\star \approx E[\mu_1] \approx 0.425$ and $\mu_2^\star \approx E[\mu_2] \approx 0.785$, almost independent of K. Going back to Figure 5, we observe that the slopes of the mean queue-length values for users 1 and 2 can be numerically computed to be 23.3 and 12.7, respectively. Our derivations suggest that these values should be close to $\frac{\alpha_i}{\mu_i^\star}$ for each $i = 1, 2$. Note that the values $\frac{\alpha_1}{\mu_1^\star} = 23.529$ and $\frac{\alpha_2}{\mu_2^\star} = 12.738$, obtained from Figure 6 are in very good agreement with our estimates of 23.5 and 12.7, respectively. Hence, we observe that the mean queue-length values are approximately equal to the corresponding invariant queue-length levels. In Figure 7, we plot $\frac{x_i[t]}{K}$ for $i = 1, 2$, and observe that both of them exhibit small fluctuations around their respective $x_i^\star$ values, where $x_1^\star = 23.5$ and $x_2^\star = 12.7$.

Next, we compare the above implementation to a queue-length-based base station scheduler and an end-to-end congestion controller with the proportionally-fair scheduler suggested by [34]. This scheduler is described next.

PROPORTIONALLY FAIR SCHEDULER [34]: The scheduler keeps track of the average service rates provided to each of the

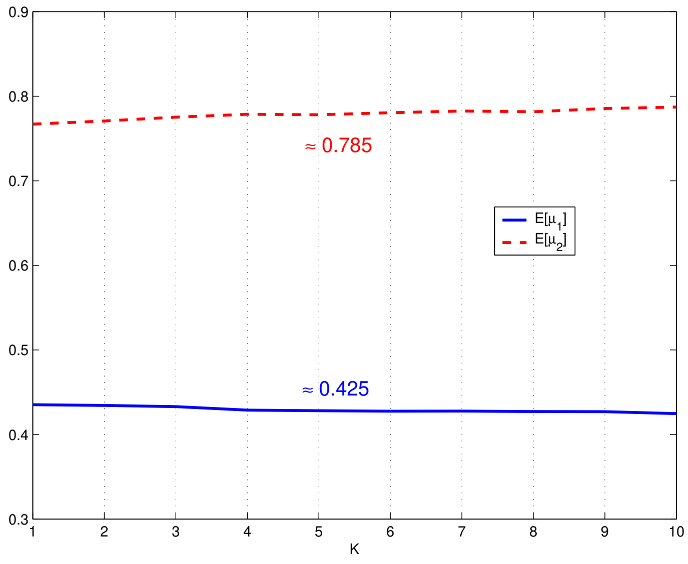

Fig. 6. Empirical average of the service rates provided to the flows for various K.

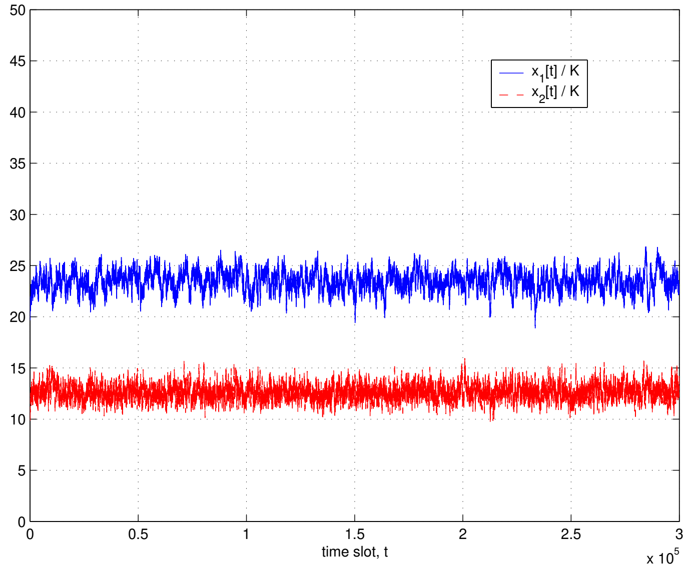

Fig. 7. Evolution of $\frac{x_i[t]}{K}$, for $i = 1, 2$ and $K = 10$.

flows in the last t_c slots. We denote this parameter by $T_i[t]$ for the i^{th} flow in slot t. Then, given the achievable service rates $\{\eta_i[t]\}$, the scheduler serves the queue, $i^*[t]$, that satisfies

$$i^*[t] = \arg\max_{i=1,2} \frac{\eta_i[t]}{T_i[t]},$$

where $T_i[t]$ is updated using an exponential weighted low-pass filter as follows:

$$T_i[t+1] = \begin{cases} \left(1 - \frac{1}{t_c}\right) T_i[t] + \frac{\eta_i[t]}{t_c} & i = i^*[t] \\ \left(1 - \frac{1}{t_c}\right) T_i[t] & i \neq i^*[t] \end{cases}$$

$\diamond$

In our simulations, we let t_c to be 100. Figure 8 shows the empirical average obtained by this proportionally fair scheduler for the same scenario we considered previously.

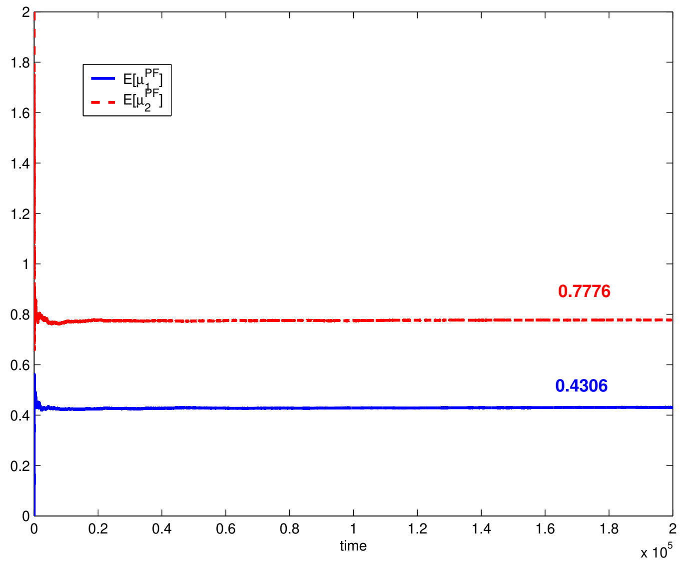

Fig. 8. Empirical average of the service rates provided to the flows with the proportionally fair scheduler.

Comparing Figures 6 and 8, we observe that the empirical mean service rate allocated to the users under the proportionally fair algorithm in [34] and our scheduler are in fact nearly identical as is to be expected.

Finally, we simulate the virtual queue implementation described in Section IV. We demonstrate the effect of β on the mean queue-length levels of the two queues in Figure 9. It can

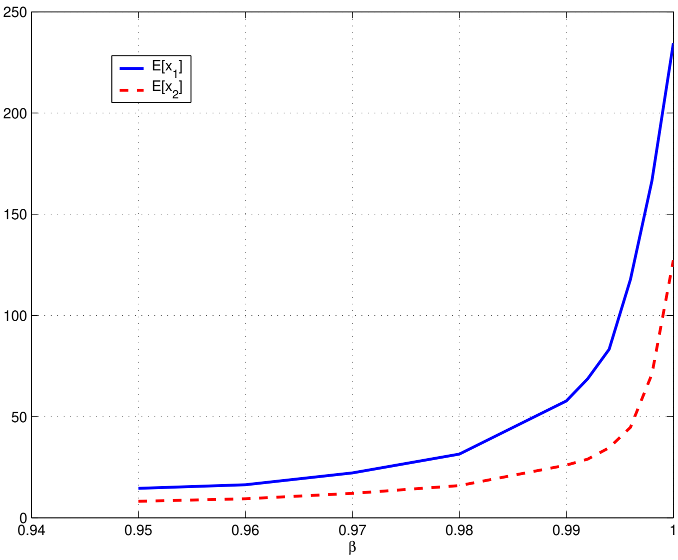

Fig. 9. The mean queue lengths as a function of β, simulated for $K = 10$.

be seen from this simulation that modifying β has a dramatic impact on the mean queue-length levels as we had argued. Of course, the choice of virtual queue parameter β will also have an effect on the mean service rates seen by the two users. For example, if β is chosen to be 0.99 (which leads to a dramatic decrease in the queue lengths as seen from Figure 9), the mean service rate for user 1 does not decrease significantly while the mean service rate for user 2 drops from 0.425 to 0.41. Thus, we see that by sacrificing throughput minimally one can dramatically reduce the queue lengths which is consistent with the observation for the Internet [11], [16], [19].

VI. Conclusions

We showed that the interaction between the end-to-end congestion controller and the local queue-length-based scheduler interestingly results in a proportionally fair allocation of the services. Furthermore, using virtual queues, the buffer levels are kept low and hence the delays experienced by the flows are also low. We also provided a characterization of the point of operation for the queue-length and service rate levels for each of the flows.

Appendix

Proof: [Proof of (9)] We write the evolution of the i^{th} queue as

$$x_i[t+1] = x_i[t] + a_i[t] - \mu_i[t] + u_i[t], \tag{13}$$

where $u_i[t]$ denotes the amount of unused service that is offered to the queue. Clearly, $u_i[t] \le \mu_i[t] \le \hat{\eta}$. Therefore, if $x_i[t] > \hat{\eta}$, then we have $u_i[t] = 0$.

Now we study the conditional mean drift of $W(\cdot)$. $E[\Delta W_t(\vec{x})]$

$$\begin{aligned} &= \sum_{i=1}^n \frac{E[(x_i[t+1]-x_i^\star)^2 - (x_i[t]-x_i^\star)^2 \mid \vec{x}[t] = \vec{x}]}{2} \\ &= \frac{1}{2}\sum_{i=1}^n E[(x_i[t+1]-x_i[t]) \\ &\qquad \times (x_i[t+1] + x_i[t] + 2x_i^\star) \mid \vec{x}[t] = \vec{x}]. \end{aligned}$$

By substituting (13) and dropping the time variable t for convenience, we get
$E[\Delta W_t(\vec{x})]$

$$\begin{aligned} &= \frac{1}{2}\sum_{i=1}^n E[(a_i - \mu_i + u_i) \\ &\qquad \times (2x_i + a_i - \mu_i + u_i - 2x_i^\star) \mid \vec{x}] \\ &\le \sum_{i=1}^n \{(x_i - x_i^\star)\left(\frac{\alpha_i K}{x_i+\gamma_i} - \bar{\mu}_i + E[u_i|\vec{x}]\right) \\ &\quad + \frac{E[a_i^2 + \mu_i^2 + u_i^2|\vec{x}]}{2} + E[u_i(a_i-\mu_i)|\vec{x}] - \frac{\alpha_i K}{x_i+\gamma_i}\bar{\mu}_i\}, \end{aligned}$$

where

$$\begin{aligned} \vec{\bar{\mu}}[t] &\in \arg\max_{\vec{\eta}\in\bar{\mathcal{C}}} \sum_{i=1}^n x_i[t]\eta_i, \\ \vec{\mu}[t] &\in \arg\max_{\vec{\eta}\in\mathcal{C}_{s[t]}} \sum_{i=1}^n x_i[t]\eta_i. \end{aligned}$$

Recall that $E[a_i^2|\vec{x}] \le A$, for some finite A. Also, $\mu_i < \hat{\eta}$ implies that $E[u_i^2 + \mu_i^2|\vec{x}] < 2\hat{\eta}^2$. Further, observing that u_i takes positive values only if $a_i < \mu_i$ implies that $E[u_i(a_i - \mu_i)|\vec{x}] \le 0$. Therefore, we can upper-bound all the terms in the last line of the above expression by a finite value, say B, independently of K. Hence, we obtain

$$\begin{aligned} E[\Delta W_t(\vec{x})] \;\le\; &\sum_{i=1}^n (x_i - x_i^\star)\left(\frac{\alpha_i K}{x_i+\gamma_i} - \bar{\mu}_i + E[u_i|\vec{x}]\right) \\ &+B. \end{aligned}$$

Notice that for all j with $x_j > x_j^\star$, we must have $u_j = 0$ when K is large, since we have $x_j > x_j^\star > \hat{\eta}$ when K is taken large enough. We already argued that there cannot be any unused service if the queue-length is larger than $\hat{\eta}$. On the other hand, for all m with $x_m < x_m^\star$, we have $E[u_m|\vec{x}] \ge 0$. Combining these two observations, we obtain the following upper bound:

$$(x_i - x_i^\star)E[u_i|\vec{x}] \le 0, \qquad \forall i, \qquad \text{for large K}.$$

Thus,

$$\begin{aligned} E[\Delta W_t(\vec{x})] \;&\le\; \sum_{i=1}^n (x_i - x_i^\star)\left(\frac{\alpha_i K}{x_i+\gamma_i} - \bar{\mu}_i\right) + B \\ &= \sum_{i=1}^n (x_i - x_i^\star)\left(\frac{\alpha_i K}{x_i+\gamma_i} - \mu_i^\star\right) \\ &\quad + \sum_{i=1}^n (x_i - x_i^\star)(\mu_i^\star - \bar{\mu}_i) + B \end{aligned}$$

We observed in the proof of Proposition 2 that

$$\sum_{i=1}^n (x_i - x_i^\star)(\mu_i^\star - \bar{\mu}_i) \le 0.$$

We make the substitution $\mu_i^\star = \frac{K\alpha_i}{x_i^\star+\gamma_i}$ to obtain

$$E[\Delta W_t(\vec{x})] \;\le\; \left(-K\sum_{i=1}^n \frac{\alpha_i(x_i - x_i^\star)^2}{(x_i+\gamma_i)(x_i^\star+\gamma_i)} + B\right).$$

Substituting $\frac{\alpha_i K}{\mu_i^\star}$ for $(x_i^\star + \gamma_i)$ in the denominator yields

$$E[\Delta W_t(\vec{x})] \;\le\; -\sum_{i=1}^n \frac{\mu_i^\star(x_i - x_i^\star)^2}{(x_i+\gamma_i)} + B.$$

Each of the terms in the above summation is non-negative. Let $i^* = \arg\max_i |x_i - x_i^\star|$ and re-write the upper bound as

$$E[\Delta W_t(\vec{x})] \le |x_{i^*} - x_{i^*}^\star|\left(-\frac{\mu_{i^*}^\star|x_{i^*} - x_{i^*}^\star|}{x_{i^*}+\gamma_{i^*}} + \frac{B}{|x_{i^*} - x_{i^*}^\star|}\right).$$

Now, suppose that $\|\vec{x}^\star - \vec{x}\| \ge c\sqrt{K}$, where $c > 0$. Noting the inequalities

$$|x_{i^*} - x_{i^*}^\star| \le \|\vec{x} - \vec{x}^\star\| \le \sqrt{n}|x_{i^*} - x_{i^*}^\star|, \tag{14}$$

we have $|x_{i^*} - x_{i^*}^\star| \ge \frac{c}{\sqrt{n}}\sqrt{K}$. Therefore, we can write

$$E[\Delta W_t(\vec{x})] \;\le\; |x_{i^*} - x_{i^*}^\star|\left(-\frac{\mu_{i^*}^\star\frac{c}{\sqrt{n}}\sqrt{K}}{\frac{\alpha_{i^*}}{\mu_{i^*}^\star}K + c\sqrt{K}} + \frac{B}{\frac{c}{\sqrt{n}}\sqrt{K}}\right)$$

$$= \frac{|x_{i^*} - x^\star_{i^*}|}{\sqrt{K}} \left(-\frac{\mu^\star_{i^*} \frac{c}{\sqrt{n}}}{\frac{\alpha_{i^*}}{\mu^\star_{i^*}} + \frac{c}{\sqrt{K}}} + \frac{B}{\frac{c}{\sqrt{n}}} \right)$$
$$\leq \frac{|x_{i^*} - x^\star_{i^*}|}{\sqrt{K}} \left(-\frac{\mu^\star_{i^*} \frac{c}{\sqrt{n}}}{\frac{\alpha_{i^*}}{\mu^\star_{i^*}} + c} + \frac{B}{\frac{c}{\sqrt{n}}} \right).$$

It is not difficult to see that the expression in the parentheses can be made strictly negative by choosing c sufficiently large, independent of K. Then the previous expression becomes

$$E[\Delta W_t(\vec{x})] \leq -\frac{\hat{\delta}}{\sqrt{K}} |x_{i^*} - x^\star_{i^*}|,$$

for some $\hat{\delta} > 0$. Using (14), we can further write

$$E[\Delta W_t(\vec{x})] \leq -\frac{\delta}{\sqrt{K}} \|\vec{x} - \vec{x}^\star\|,$$

for all $\|\vec{x} - \vec{x}^\star\| \geq c\sqrt{K}$, for K large enough and where $\delta := \frac{\hat{\delta}}{\sqrt{n}}$.

When $\|\vec{x} - \vec{x}^\star\| < c\sqrt{K}$, it is not difficult to see that

$$E[\Delta W_t(\vec{x})] \leq \zeta$$

for some $\zeta \in (0, \infty)$. Combining the previous two inequalities completes the proof. ∎

REFERENCES

[1] R. Agrawal, A. Bedekar, R. J. La, and V. Subramanian. Class and channel condition based weighted proportionally fair scheduler. In *Proceedings of the International teletraffic Congress*, pages 553–65, 2001.

[2] T. Alpcan and T. Başar. A utility-based congestion control scheme for internet-style networks with delay. In *Proceedings of IEEE Infocom*, San Francisco, California, March-April 2003.

[3] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, R. Vijayakumar, and P. Whiting. Scheduling in a queueing system with asynchronously varying service rates, 2000. Bell Laboratories Technical Report.

[4] M. Armony and N. Bambos. Queueing dynamics and maximal throughput scheduling in switched processing systems. Technical Report Netlab-2001-09/01, Stanford University.

[5] S. Asmussen. *Applied Probability and Queues*. Springer-Verlag, New York, 2003.

[6] D. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, MA, 1995.

[7] S. C. Borst. User-level performance of channel-aware scheduling algorithms in wireless data networks. In *Proceedings of IEEE INFOCOM*, 2003.

[8] S.C. Borst and P.A. Whiting. Dynamic rate control algorithms for HDR throughput optimization. In *Proceedings of IEEE INFOCOM*, pages 976–985, 2001.

[9] R. Buche and H. J. Kushner. Control of mobile communication systems with time-varying channels via stability methods. *IEEE Transactions on Automatic Control*, 2004. To appear.

[10] A. Eryilmaz, R. Srikant, and J. Perkins. Stable scheduling policies for fading wireless channels. In *Proceedings of IEEE International Symposium on Information Theory*, 2003. To appear in the *IEEE/ACM Transactions on Networking, 2004*.

[11] R. J. Gibbens and F. P. Kelly. Resource pricing and the evolution of congestion control. *Automatica*, 35:1969–1985, 1999.

[12] A. Jalali, R. Padavoni, and R. Pankaj. Data throughput of CDMA-HDR: a high efficiency-high data rate personal communication system. In *Proceedings IEEE Vehicular Technology Conference*, pages 1854–1858.

[13] F. P. Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 8:33–37, 1997.

[14] F. P. Kelly, A. Maulloo, and D. Tan. Rate control in communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49:237–252, 1998.

[15] H. Khalil. *Nonlinear Systems*. 2nd edition, Prentice Hall, Upper Saddle River, NJ, 1996.

[16] S. Kunniyur and R. Srikant. Analysis and design of an adaptive virtual queue algorithm for active queue management. In *Proceedings of ACM Sigcomm*, pages 123–134, San Diego, CA, August 2001.

[17] S. Kunniyur and R. Srikant. A time-scale decomposition approach to adaptive ECN marking. *IEEE Transactions on Automatic Control*, 47(6):882–894, June 2002.

[18] H. J. Kushner and P. A. Whiting. Convergence of proportional-fair sharing algorithms under general conditions, February 2003. Preprint.

[19] A. Lakshmikantha, C. Beck, and R. Srikant. Robustness of real and virtual queue based active queue management schemes. *IEEE/ACM Transactions on Networking*, 2004. To appear. An earlier version appeared in the *Proceedings of the American Control Conference*, June 2003.

[20] R. Leelahakriengkrai and R. Agrawal. Scheduling in multimedia wireless networks. In *Proceedings of ITC*, Brazil, 2001.

[21] X. Lin and N. Shroff. The multi-path utility maximization problem. In *Proceedings of the Allerton Conference on Communications, Control and Computing*, Monticello, IL, October 2003.

[22] X. Liu, E. Chong, and N. Shroff. Opportunistic transmission scheduling with resource-sharing constraints in wireless networks. *IEEE Journal on Selected Areas in Communications*, 19(10):2053–2064, October 2001.

[23] Y. Liu and E. Knightly. Opportunistic fair scheduling over multiple wireless channels. In *Proceedings of IEEE INFOCOM*, San Francisco, CA, April 2003.

[24] S. H. Low and D. E. Lapsley. Optimization flow control, I: Basic algorithm and convergence. *IEEE/ACM Transactions on Networking*, pages 861–875, December 1999.

[25] M.J. Neely, E. Modiano, and C.E. Rohrs. Dynamic power allocation and routing for time varying wireless networks. *Proceedings of IEEE Infocom*, April 2003. To appear.

[26] F. Paganini. A global stability result in network flow control. *Systems and Control Letters*, 46(3):153–163, 2002.

[27] A. G. Pakes. Some conditions on the ergodicity and recurrence of Markov chains. *Operations Research*, 17, 1969.

[28] S. Shakkottai, R. Srikant, and A. Stolyar. Pathwise optimality of the exponential scheduling rule for wireless channels. In *Proceedings of ISIT*, Lausanne, Switzerland, July 2002. To appear in the *Advances in Applied Probability*, 2004.

[29] S. Shakkottai and A. Stolyar. Scheduling for multiple flows sharing a time-varying channel: The exponential rule. *Translations of the AMS, Series 2,* A volume in memory of F. Karpelevich, 207, 2002.

[30] R. Srikant. *The Mathematics of Internet Congestion Control*. Birkhauser, 2004.

[31] V. Subramanian and R. Agrawal. A stochastic approximation analysis of channel condition aware wireless scheduling algorithms. In *Proceedings of the INFORMS Telecommunications Conference*, 2002.

[32] L. Tassiulas. Scheduling and performance limits of networks with constantly varying topology. *IEEE Transactions on Information Theory*, pages 1067–1073, May 1997.

[33] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control*, pages 1936–1948, December 1992.

[34] D. N. Tse. Multi-user diversity and proportional fairness. US Patent 6449490.

[35] P. Viswanath, D. Tse, and R. Laroia. Opportunistic beamforming using dumb antennas. *IEEE Transactions on Information Theory*, 48(6):1277–1294, June 2002.

[36] W.-H. Wang, M. Palaniswami, and S. H. Low. Optimal flow control and routing in multi-path networks. *Performance Evaluation*, pages 119–132, 2003.

[37] J.T. Wen and M. Arcak. A unifying passivity framework for network flow control. In *Proceedings of IEEE Infocom*, April 2003.

[38] H. Yaiche, R. R. Mazumdar, and C. Rosenberg. A game-theoretic framework for bandwidth allocation and pricing in broadband networks. *IEEE/ACM Transactions on Networking*, 8(5):667–678, October 2000.