Optimal Utility Based Multi-User Throughput Allocation subject to Throughput Constraints

Matthew Andrews
Bell Laboratories, Lucent Technologies
Murray Hill, NJ 07974.
andrews@research.bell-labs.com

Lijun Qian
Prairie View A&M University
Prairie View, TX 77446.
lqian@pvamu.edu

Alexander Stolyar
Bell Laboratories, Lucent Technologies
Murray Hill, NJ 07974.
stolyar@research.bell-labs.com

***Abstract*— We consider the problem of scheduling multiple users sharing a time-varying wireless channel. (As an example, this is a model of scheduling in 3G wireless technologies, such as CDMA2000 3G1xEV-DO downlink scheduling.) We introduce an algorithm which seeks to optimize a concave utility function $\sum_i H_i(R_i)$ of the user throughputs R_i, *subject to certain lower and upper throughput bounds:* $R_i^{min} \leq R_i \leq R_i^{max}$. The algorithm, which we call the Gradient algorithm with Minimum/Maximum Rate constraints (GMR) uses a *token counter* mechanism, which modifies an algorithm solving the corresponding *unconstrained* problem, to produce the algorithm solving the problem with throughput constraints. Two important special cases of the utility functions are $\sum_i \log R_i$ and $\sum_i R_i$, corresponding to the common Proportional Fairness and Throughput Maximization objectives.**

We study the dynamics of user throughputs under GMR algorithm, and show that GMR is asymptotically optimal in the following sense. If, under an appropriate scaling, the throughput vector $R(t)$ converges to a fixed vector R^* as time $t \to \infty$ then R^* is an optimal solution to the optimization problem described above. We also present simulation results showing the algorithm performance.

Key words and phrases: Scheduling, wireless, CDMA, 3G, time varying channel, QoS, Gradient algorithm, Proportional Fair, Maximum Throughput, rate constraints, guaranteed rate

I. Introduction

We consider a *variable channel scheduling* model that is motivated by the 3G1xEV-DO system for high-speed wireless data. A *channel* serves I *traffic flows* and operates in discrete (slotted) time. In each time slot a *scheduler* chooses one flow to serve. The channel *state* is random, and it determines the service rates of each flow in the current time slot, if that flow is chosen for service.

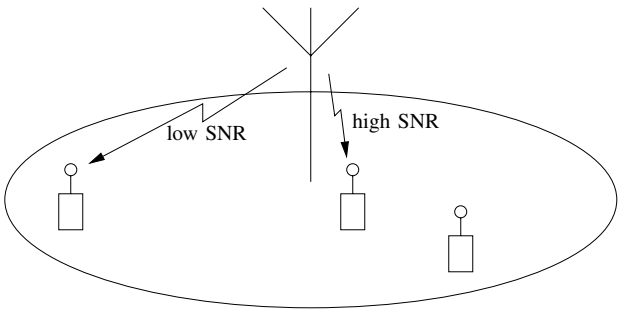


Fig. 1. A wireless system.

In the wireless context, the traffic flows correspond to the downlink data flows from a basestation to multiple mobile users. (See Figure 1). If the signal-to-noise ratio (SNR) for user i is high, then if user i is picked for service in time slot t it can receive data at a high rate. Conversely, if the signal-to-noise ratio is low it can only receive data at a low rate. The scheduler resides at the basestation. In the EV-DO system the scheduler knows the feasible service rates because each user measures a pilot signal-to-noise ratio and reports back to the basestation in a Data Rate Control message (DRC). We shall sometimes use $DRC_i(t)$ to denote the rate at which user i can be served at time t if it is chosen for service. We emphasize that this rate is both *user-dependent* and *time-varying*.

In this paper we shall work in an *infinitely backlogged* model in which for each flow there is always data available for service. The set of all feasible long-term average service rate vectors $R = (R_1, \ldots, R_I)$ is called the system *rate region*, V. Rate region is a convex polyhedron in the positive orthant.

Past work in the infinitely backlogged model has considered scheduling algorithms that optimize (over rate region V) a certain utility function of the average rates R. For example, the Proportional Fair algorithm [20], [3] aims to optimize the function $\sum_i \log R_i$. However, this optimization provides no guarantees to *individual users*. Each user may at times receive unacceptably bad service.

In this paper we demonstrate how to rectify this problem by presenting scheduling algorithms that optimize a utility function of the rate vector R, subject to *minimum and maximum rate constraints* on the individual components R_i. More precisely, we are interested in solving the following optimization problem:

$$\max H(R) \tag{1}$$

subject to

$$R \in V \tag{2}$$

$$R_i \geq R_i^{\min}, \quad i \in I, \tag{3}$$

$$R_i \leq R_i^{\max}, \quad i \in I, \tag{4}$$

for utility functions of the form

$$H(R) = \sum_i H_i(R_i), \tag{5}$$

where each $H_i(x)$ is an increasing concave continuously differentiable function defined for $x \geq 0$. (We also allow the case $\lim_{x\downarrow 0} H_i(x) = H_i(0) = -\infty$.) The rate constraint parameters $R_i^{\min}$ and $R_i^{\max}$ are fixed constants such that $0 \leq R_i^{\min} \leq R_i^{\max}$ and $R_i^{\max} > 0$.

A. Our Results

In Section IV we propose an algorithm, called the Gradient algorithm with Minimum/Maximum Rate constraints (GMR) that aims to solve the problem (1)-(4). In time slot t, GMR always serves flow,

$$i \in \arg\max_{i\in I} e^{a_i T_i(t)} H_i'(R_i(t)) \boldsymbol{DRC}_i(t) \ ,$$

where $R_i(t)$ is the current average service rate received by queue i, $T_i(t)$ is a *token counter* for queue i, and $a_i > 0$ is a parameter. The token counter $T_i(t)$, which we define precisely in Section IV, is the key mechanism by which we enforce the rate constraints. In each time slot it is incremented at rate either $R_i^{\min}$ or $R_i^{\max}$ and it is decremented whenever flow i is served. If in some finite time interval, flow i receives service less than $R_i^{\min}$ then $T_i(t)$ has a positive drift and so flow i is more likely to be served. If flow i receives service more than $R_i^{\max}$ then $T_i(t)$ has a negative drift and so flow i is less likely to be served. (See Remark 2 in Section V on the form of GMR algorithm.)

There are two special cases of the utility (objective) function that are of particular interest.

1) The Proportional Fair objective[1] which corresponds to $H(R) = \sum_i \log R_i$. In this special case, GMR always serves flow,

$$i \in \arg\max_{i\in I} e^{a_i T_i(t)} \frac{\boldsymbol{DRC}_i(t)}{R_i(t)}.$$

We refer to this special case of the algorithm as *Proportional Fair with Minimum/Maximum Rates* (PFMR). We remark that if the rate constraints are trivial, i.e. $R_i^{\min} = 0$ and $R_i^{\max} = \infty$ for all i, then PFMR reduces to the standard Proportional Fair algorithm of [20], [3] (i.e. it always serves flow $i \in \arg\max_{i\in I} \boldsymbol{DRC}_i(t)/R_i(t)$.

2) The Throughput Maximization objective which corresponds to $H(R) = \sum_i R_i$. In this case, GMR always serves flow,

$$i \in \arg\max_{i\in I} e^{a_i T_i(t)} \boldsymbol{DRC}_i(t).$$

and we refer to the algorithm as *Maximum Throughput with Minimum/Maximum Rates* (MTMR).

In Section V we show asymptotic optimality of GMR in the sense that if, under an appropriate scaling, $R(t)$ converges to a fixed vector R^* as $t \to \infty$ then R^* is a solution to the problem (1)-(4). (See Remarks 2 and 3 in Section V on this notion of asymptotic optimality, and on the convergence properties of GMR and related algorithms.)

In Section VI we present simulation results to illustrate the behavior of PFMR and MTMR.

B. Motivation for Minimum/Maximum Rate Constraints

A guarantee on minimum bandwidth is arguably the simplest possible Quality-of-Service guarantee. Therefore we believe it is natural that subscribers to an expensive mobile high-speed data service would expect such an assurance. Other reasons why we feel it is important to provide minimum rate constraints are:

1) Some applications need a minimum rate in order to perform well. For example, streaming audio and video can become unusable if the bandwidth is too small.
2) Even for static TCP-based applications such as web browsing if the bandwidth is too small then we typically get a large queue buildup which can lead to TCP time-outs and poor performance. Such effects were discussed by Chakravorty et al. in [7].
3) Providing a minimum rate guarantee can help to smooth out the effects of a variable wireless channel.
4) Providing a minimum rate can allow us to ensure that a slot-based service such as EV-DO is no worse than circuit-based data systems such as wireline dialup or 3G1X wireless service.
5) By setting $R_i^{\min}$ differently for different users we can ensure that high-paying premium customers receive better service than regular customers.

At first, it might seem counterintuitive that a system operator would want to provide maximum rate constraints. However, some possible reasons are:

1) If a user has only paid for a cheap data service, the operator might wish to cap their data rate in order to give them an incentive to upgrade to a more expensive premium service.
2) The first user that signs up for a data service will have the system all to themselves. Then, as more customers sign up, the service to the first user will typically deteriorate as it has to share bandwidth with all the new users. One solution to this deterioration is to cap all users' maximum rates to some reasonable level so that they do not experience perfromance degradation when more users subscribe. (Of course, doing this may involve wasting time slots which might not be so desirable.)

We remark that if the system operator does not wish to have maximum rate constraints then this is easily accomplished by setting $R_i^{\max}$ to infinity (or some suitably large value).

C. Previous Work

The most closely related work is that of [16], [17], [5], where the Gradient algorithm was studied and proved optimal (under various models and other assumptions) for our problem

[1]The reason this objective can be desirable is that multiplying user i's rate by some factor c has the same effect on the objective as multiplying user j's rate by c. Alternatively, if R_i are the rates that maximize this objective, then for any other feasible set of rates R_i' that satisfy the constraints we must have,

$$\sum_i \frac{R_i' - R_i}{R_i} \leq 0.$$

without mimimim and maximum rate constraints. (See [5] for the results in our model setting.) This "unconstrained" Gradient algorithm is a special case of GMR. We therefore have a very natural question: "Why not use the unconstrained Gradient algorithm for our problem as well, and deal with the rate constraints simply by modifying the utility function in a way that penalizes rate constraint violations?" As we demonstrate later in the paper, such an approach does not work well. Roughly, the reason is that an algorithm with a modified utility function typically "overreacts" to temporary rate constraint "violations," and this significantly degrades the achieved value of the utility function.

The problem of utility based throughput allocation was also previously considered in [18], [19]. In particular, these papers addressed throughput allocation subject to certain constraints. There are two key differences between our algorithm and those in [18], [19]. First, GMR optimizes a concave utility function of the average throughputs, while schemes in [18], [19] are in essence restricted to linear utility functions. Secondly, our token counter mechnism for average rate constraints enforcement is substantially different from the stochastic approximation based schemes of [18], [19].

A solution to the problem of maintaining service rates in desired proportions to one another was presented in [12]. A higher-level problem in which each user has a finite amount of data to serve and it leaves the system once this data is served is studied in [13].

Recall that we are concerned with the service rates provided to each user. Our model assumes that whenever a user is scheduled it always has data to serve. Now we would like to contrast our algorithm with algorithms for a different, much studied, model, where each user has a queue that is fed by an arrival process. In this setting, the most widely studied algorithms are *Max-Weight* type algorithms [8], [9], [1]. In our setting, the "queue-length-based" version of such an algorithm always serves the user that maximizes $DRC_i(t)Q_i(t)$ where $Q_i(t)$ is the amount of data in the queue for user i. This algorithm is known to be stable[2], which roughly means that it keeps the queues from running away to infinity whenever possible. Other stable algorithms include the "delay-based" version of MaxWeight [9], [1] and the EXP algorithm [10], [11]. The "delay-based" MaxWeight always serves the user that maximizes $DRC_i(t)W_i(t)$ where $W_i(t)$ is the Head-of-Line delay for user i. EXP is a more complex algorithm that aims to have more control over the delay distributions.

Despite the difference of the above problem from ours, we note that the above stable algorithms such as Max-Weight could be applied, for example, to provide *minimum* rate contraints in our problem, since they could operate on the token counters rather than the actual queue lengths. However, in this case all token counter stabilty means is that the minimum rate constraints are indeed enforced. Such a solution would *not* optimize an objective function subject to minimum rate constraints.

We finally remark that in our model the channel rate process is governed by a stationary stochastic process. The problem of scheduling over a non-stationary wireless channel is addressed in [15].

[2]We remark that in this model Proportional Fair is known to be *not* stable [14].

II. Variable Channel Scheduling Model

We consider the following model introduced in [1]. There is a finite set $I = \{1, 2, \ldots, I\}$ of "traffic" flows served by a *channel*. (We will use the same symbol I for both the set and the number of its elements.) Each flow i consists of discrete *customers* ("bits of data"), which we sometimes call type i customers. In this paper we assume that there is always a sufficient "supply" of customers of each flow to serve.

The system operates in discrete time $t = 0, 1, 2, \ldots$. By convention, we will identify an (integer) time t with the unit time interval $[t, t+1)$, which will sometimes be referred to as the *time slot* t.

The channel has a finite set of *channel states* M. In each time slot, the channel is in one of the states $m \in M$; and the sequence of states $m(t),\ t = 0, 1, 2, \ldots$, forms an (irreducible) finite state Markov chain with stationary distribution $\{\pi_m,\ m \in M\}$,

$$\pi_m > 0,\ \forall m \in M, \qquad \sum_{m \in M} \pi_m = 1\ .$$

If at time t the channel is in state $m \in M$ and it chooses queue i for service, then an integer number $\mu_i^m \geq 0$ of type i customers (e.g., bits of data) are served and depart the system at time $t+1$. We use $\mu^m \doteq (\mu_1^m, \ldots, \mu_I^m)$ to denote the corresponding vector of service rates, and we assume that for each flow i, there is at least one state m such that $\mu_i^m > 0$. (Sometimes, we will write $DRC_i(t) = \mu_i^{m(t)}$ for the service rate available to user i in slot t, if it were to be picked for service. This notation, which we already used in the Introduction, is standard in the CDMA2000 1xEV-DO literature.) Note that the model in which channel state $m(t)$ is in fact a combination of independently randomly varying (according to independent Markov chains) channel states $m_i(t)$ of individual users is essentially a special case of our model (up to very mild assuptions guaranteeing that Markov chain $m(t)$ is irreducible).

Suppose a stochastic matrix $\phi = (\phi_{mi}, m \in M, i = 1, \ldots, I)$ is fixed, which means that $\phi_{mi} \geq 0$ for all m and i, and $\sum_i \phi_{mi} = 1$ for every m. Consider a *Static Service Split* (SSS) scheduling rule, parameterized by the matrix ϕ. When the server is in state m, the SSS rule chooses for service queue i with probability ϕ_{mi}. (Sometimes, we call the matrix ϕ itself an SSS rule.) Clearly, the vector $v = (v_1, \ldots, v_I) = v(\phi)$, where

$$v_i = \sum \pi_m \phi_{mi} \mu_i^m\ ,$$

gives the long term average service rates allocated to different flows under an SSS rule ϕ.

We define the system *rate region* to be the set V of all vectors $v(\phi)$ for all possible SSS rules ϕ. Thus V is the set of

long-term service rate vectors which the system is capable of providing. Rate region V is a convex closed bounded polyhedron in the positive orthant. (See [4].) By V^* we denote the subset of maximal elements of V: namely, $v \in V^*$ if conditions $v \le u$ (componentwise) and $u \in V$ imply $u = v$. Clearly, V^* is a part of the outer ("north-east") boundary of V.

The subset $V^{cond} \subseteq V$ of elements $v \in V$ satisfying conditions $R_i^{\min} \le v_i \le R_i^{\max}$ (i.e. conditions (3) and (4)) for all i, is also a convex closed bounded set.

Since each function $H_i(R_i)$ is continuous and increasing, we have the following simple fact.

Proposition 1: If V^{cond} is non-empty, then at least one solution R^* of problem (1)-(4) exists. If V^{cond} contains at least one point of the set V^*, then any solution $R^* \in V^*$. If all functions $H_i(R_i)$ are strictly concave (for example, $H_i(R_i) = \log(R_i)$), a solution R^* is unique.

III. Basic Notation and Conventions

The sets of real numbers and non-negative real numbers are denoted by $\mathcal{R}$ and $\mathcal{R}_+$ respectively; $\mathcal{R}^I$ and $\mathcal{R}_+^I$ denote their I times products.

For vectors $x, y \in \mathcal{R}^I$,

$$x \cdot y \doteq \sum_i x_i y_i \quad \text{is scalar product,}$$

$$x \times y \doteq (x_1 y_1, \ldots, x_I y_I) \quad \text{is component-wise product,}$$

$$\exp(x) \doteq (\exp(x_1), \ldots, \exp(x_I)) \ .$$

The Euclidean norm $\|x\| \doteq \sqrt{x \cdot x}$ defines metric $\|x - y\|$ on $\mathcal{R}^I$.

The gradient of the function H is denoted by ∇H, i.e.

$$\nabla H(x) = (H_1'(x_1), \ldots, H_I'(x_I)).$$

For a function $\xi = (\xi(t),\ t \ge 0)$, $\theta_d \xi$ denotes its backward shift by time $d \ge 0$, namely

$$[\theta_d \xi](t) = \xi(t + d),\ t \ge 0.$$

IV. GMR Algorithm

We now formulate the Gradient algorithm with Minimum/Maximum Rate constraints (GMR), which seeks to solve the optimization problem (1)-(4). (Recall that $\mu_i^{m(t)} = DRC_i(t)$ is the service rate available to user i in the time slot t, if this user were to be chosen for service.)

GMR: *In a time slot t, serve queue*

$$i \in \arg\max_{i \in I} e^{a_i T_i(t)} H_i'(R_i(t)) \mu_i^{m(t)} \ , \tag{6}$$

where $R_i(t)$ is the current average service rate received by queue i, $T_i(t)$ is a "token counter" for queue i, and $a_i > 0$ is a parameter. The values of average rate R_i are updated as in the Proportional Fair algorithm [20], [3]:

$$R_i(t+1) = (1 - \beta) R_i(t) + \beta \mu_i(t) \ ,$$

where $\beta > 0$ is a small fixed parameter, and $\mu_i(t) = \mu_i^{m(t)} (= \mathrm{DRC}_i(t))$ if user i was actually served in slot t and $\mu_i(t) = 0$ otherwise. The token counter T_i is updated as follows:

$$T_i(t+1) = T_i(t) + R_i^{token} - \mu_i(t) \ , \tag{7}$$

where $R_i^{token} = R_i^{\min}$ if $T_i(t) \ge 0$, and $R_i^{token} = R_i^{\max}$ if $T_i(t) < 0$.

If $R_i^{\max} = \infty$ (i.e. constraint (4) is absent) for some i, the token counter update rule (7) is simplified for this i to:

$$T_i(t+1) = \max\{0, T_i(t) + R_i^{\min} - \mu_i(t)\} \ . \tag{8}$$

If $R_i^{\min} = 0$ for some i, the rule (7) is simplified for this i to:

$$T_i(t+1) = \min\{0, T_i(t) + R_i^{\max} - \mu_i(t)\} \ . \tag{9}$$

Remark. The choice of the units for different variables and parameters used by the GMR algorithm is a matter of implementation convenience. One choice of the units (which is the one we used in the similations, presented in Section VI) is as follows. Amounts of data are measured in bits and the time is measured in slots. Consequently, all data rates $R_i(t)$, $\mu_i^{m(t)} = DRC_i(t)$, $\mu_i(t)$, $R_i^{\min}$, $R_i^{\max}$, R_i^{token} are measuted in bits/slot (or bits, if a calculation involves amount of data served or arrived at the corresponding rate in one slot, as in (7)-(9)). Finally, token counters $T_i(t)$ are measured in bits, and parameters a_i are in 1/bits.

As we mentioned earlier, we refer to the special cases of the GMR algorithm, corresponding to utility functions $H(R) = \sum_i \log(R_i)$ and $H(R) = \sum_i R_i$, as PFMR and MTMR algorithms, respectively. By specializing (6), we see that

the **PFMR** scheduling rule is

$$i \in \arg\max_{i \in I} e^{a_i T_i(t)} \frac{DRC_i(t)}{R_i(t)} \ , \tag{10}$$

and **MTMR** rule is

$$i \in \arg\max_{i \in I} e^{a_i T_i(t)} DRC_i(t) \ . \tag{11}$$

The token counter T_i provides the key mechanism trying to ensure that the user i received (long term) service rate stays between $R_i^{\min}$ and $R_i^{\max}$. The dynamics of the token counter process $T_i(t)$ (see (7)) is roughly described and interpreted as follows. There is a virtual "token queue" (which may take negative values) corresponding to each flow i. The tokens "arrive in the (token) queue" (i.e. T_i is incremented) at the rate $R_i^{\min}$ or $R_i^{\max}$ per slot, if $T_i(t)$ is positive or negative, respectively. (For this reason, we sometimes refer to $R_i^{\min}$ and $R_i^{\max}$ as the *token rates*.) If user i is served in slot t, then $\mu_i^{m(t)} = DRC_i(t)$ tokens are "removed from the queue" (i.e. T_i is decremented). Thus, if in a certain time interval, the average service rate of flow i is less than $R_i^{\min}$, the token queue size T_i has "positive drift", and therefore the chances of flow i being served in each time slot *gradually* increase. If the average service rate of flow i is above $R_i^{\max}$, then T_i has negative drift, thus *gradually* decreasing the chances of user

i being picked for service. If average service rate of flow i is between $R_i^{\min}$ and $R_i^{\max}$, then T_i has positive drift when T_i is negative and negative drift when T_i is positive; as a result, in this case T_i will stay "around 0."

V. User throughput dynamics under GMR with small parameters β and a_i

In this section we consider the dynamics of user throughputs and token counters under the GMR algorithm when parameters β and a_i are small. Namely, we consider the asymptotic regime such that β converges to 0, and each $a_i = \beta\alpha_i$ with some fixed $\alpha_i > 0$. We study the dynamics of *fluid sample paths* (FSP), which are (roughly speaking) possible trajectories $(r(t), \tau(t))$ of a random process which is a limit of the process $(R(t/\beta), \beta T(t/\beta))$ as $\beta \to 0$. (Thus, $r(t)$ approximates the behavior of the vector of throughputs $R(t)$ when β is small and we "speed-up" time by the factor $1/\beta$; $\tau(t)$ approximates the vector $T(t)$ scaled down by factor β, and with $1/\beta$ time speed-up.) The main result of this section is a "necessary condition for throughput convergence" (Theorem 1), which roughly says that if FSP is such that the vector of throughputs $r(t)$ converges to some vector R^* as $t \to \infty$, then R^* is necessarily a solution to the problem (1)-(4).

We now define the asymptotic regime and an FSP more precisely. Consider a sequence of positive values of β, converging to 0, and assume that $a_i = \beta\alpha_i$, $\alpha_i > 0$, for each β. (We will denote $\alpha \doteq (\alpha_1, \ldots, \alpha_I)$.) For each β we consider a *realization* (that is, a fixed sample path) of the channel state process $m^\beta = (m^\beta(t),\ t = 0, 1, 2, \ldots)$. We assume that the sequence of (fixed realizations) m^β satisfies the law of large numbers condition, namely, that for any $t > 0$ and any $m \in M$,

$$\frac{1}{t/\beta} \sum_{0 \le n \le t/\beta} I\{m^\beta(n) = m\} \to \pi_m, \tag{12}$$

where $I\{\cdot\}$ denotes here the indicator function. For each β, let $R^\beta(\cdot)$ and $T^\beta(\cdot)$ be the realizations of the throughput and token counter vector-processes corresponding to the GMR algorithm. They are uniquely defined by the realization m^β and the fixed initial states $R^\beta(0)$ and $T^\beta(0)$. Finally, we extend the domain of functions $R^\beta(t)$ and $T^\beta(t)$ to all real $t \ge 0$ by adopting the convention that they are constant within each time slot $[t, t+1)$ for all integer t, and consider the following rescaled rate and token counter trajectories:

$$r^\beta(t) = R^\beta(t/\beta), \quad \tau^\beta(t) = \beta T^\beta(t/\beta), \quad t \ge 0.$$

A pair of vector-functions $(r = (r(t),\ t \ge 0),\ \tau = (\tau(t),\ t \ge 0))$ is called a *fluid sample path* (FSP), if the uniform on compact sets (u.o.c.) convergence

$$(r^\beta, \tau^\beta) \to (r, \tau)$$

holds for at least one sequence (r^β, τ^β) defined as above. (In our case, the u.o.c. convergence means that for any fixed $b \ge 0$, the convergence is uniform over $t \in [0, b]$.)

We can now formulate the main result of this section.

Theorem 1: Suppose FSP (r, τ) is such that

$$r(t) \to R^* \quad \text{as } t \to \infty$$

and $\tau(t)$ remains uniformly bounded for all $t \ge 0$. Then, R^* is a solution to the problem (1)-(4) and, moreover, $R^* \in V^{cond} \cap V^* \ne \emptyset$.

Remark 1. It is easy to show using FSP properties described below in Lemmas 1 and 2, that if $V^{cond} \cap V^* = \emptyset$, then for any FSP the vector $\tau(t)$ cannot remain bounded and in fact $\|\tau(t)\| \to \infty$ as $t \to \infty$. Therefore, the uniform boundedness of $\tau(t)$ alone implies that $V^{cond} \cap V^* \ne \emptyset$.

Remark 2. It will be easy to see from our proofs that Theorem 1 still holds if factor $e^{a_i T_i(t)}$ in the GMR rule (6) is replaced by $\alpha(a_i T_i(t))$, where $\alpha(x)$ is an arbitrary continuous, strictly increasing function, such that $\alpha(0) = 1$, $\alpha(x) \downarrow 0$ as $x \downarrow -\infty$, and $\alpha(x) \uparrow \infty$ as $x \uparrow \infty$. Moreover, the theorem still holds if rule (6) has the following "additive form:"

$$i \in \arg\max_{i \in I} [H_i'(R_i(t)) + \nu(a_i T_i(t))] \mu_i^{m(t)}\ , \tag{13}$$

where $\nu(x)$ is an arbitrary continuous, strictly increasing function, such that $\nu(0) = 0$, $\nu(x) \downarrow -\infty$ as $x \downarrow -\infty$, and $\nu(x) \uparrow \infty$ as $x \uparrow \infty$. In this paper we choose to work with the specific form (6) of GMR algorithm, to simplify the exposition to some degree, and also because this specific "multiplicative" form shows good convergence properties in our simulations and is in fact very convenient for practical implementation.

Remark 3. We remind that Theorem 1 does *not* assert that the convergence of throughputs $r(t)$ to the set of optimal solutions of the problem (1)-(4) in fact holds. (Our simulation experiments show good convergence properties of the GMR in the form (6).) Subsequently to the present work, it has been proved recently in [6] that such convergence *does* hold for a quite general *Greedy Primal-Dual* (GPD) algorithm, which, for our model, specializes roughly to the scheduling rule

$$i \in \arg\max_{i \in I} [H_i'(R_i(t)) + a_i T_i(t)] \mu_i^{m(t)}\ . \tag{14}$$

We note however, that the GPD algorithm convergence proof in [6] does *not* apply to the GMR algorithm (6).

To prove Theorem 1, we will first describe the basic FSP properties in Lemmas 1 and 2. Then we prove two special (increasingly general) cases of Theorem 1 in Lemmas 3 and 4, and conclude this section with the proof of Theorem 1 itself.

Lemma 1: For any fluid sample path, all its component functions are Lipschitz continuous in $[0, \infty)$, with the Lipschitz constant upper bounded by $C + \|r(0)\|$, where $C > 0$ is a fixed constant depending only on the system parameters.

Proof is analogous to that in [5]. ∎

Since all component functions of an FSP are Lipschitz, they are absolutely continuous, and therefore almost all points $t \ge 0$ (with respect to Lebesgue measure) are such that all component functions of an FSP have derivatives.

Lemma 2: The family of fluid sample paths satisfies the following additional properties.

(i) For almost all $t \geq 0$ (with respect to Lebesgue measure) we have:

$$r'(t) = v(t) - r(t), \tag{15}$$

where

$$v(t) \in \arg\max_{v \in V}[\exp(\alpha \times \tau(t)) \times \nabla H(r(t))] \cdot v \ , \tag{16}$$

and

$$\tau'(t) = \alpha(r^{token}(t) - v(t)) \ , \tag{17}$$

where the components $r_i^{token}(t), \ i = 1, \ldots, I$ of vector $r^{token}(t)$ are such that

$$r_i^{token}(t) \begin{cases} = R_i^{\min} & \text{if } \tau_i(t) > 0, \\ \in [R_i^{\min}, R_i^{\max}] & \text{if } \tau_i(t) = 0, \\ = R_i^{\max} & \text{if } \tau_i(t) < 0. \end{cases} \tag{18}$$

(ii) "Shift property." If (r, τ) is an FSP, then for any $d \geq 0$, $(\theta_d r, \theta_d \tau)$ is also an FSP.
(iii) "Compactness." If a sequence of FSPs $(r^{(j)}, \tau^{(j)}) \rightarrow (r, \tau)$ uniformly on compact sets as $j \rightarrow \infty$, then (r, τ) is also an FSP.

The proof of properties (i)(15) and (iii) is completely analogous to that of the corresponding FSP properties in [5]. Property (i)(17) is easy to verify directly, using the definition of an FSP - we omit the proof to save space. The shift property (ii) (as well as compactness (iii)) is an inherent property of fluid sample paths, valid for FSPs defined in many different settings (see for example [4] for a proof); and it is easily verified directly as well. ∎

Lemma 3: Suppose (r, τ) is a stationary FSP, namely

$$r(t) \equiv R^* \text{ and } \tau(t) \equiv \tau^* \ \text{ for all } t \geq 0.$$

Then, R^* is a solution to the problem (1)-(4) and $R^* \in V^{cond} \cap V^* \neq \emptyset$.

Proof. Let vector $\eta \in \mathcal{R}^I$ be defined as $\eta \doteq \exp(\alpha \times \tau^*)$. In view of property (15), it follows from $r(t) \equiv R^*$ that we have $v(t) \equiv R^*$ as well. By (16), for almost all $t \geq 0$ we have

$$v(t) \in \arg\max_{v \in V}[\eta \times \nabla H(r(t))] \cdot v.$$

We see (since $v(t) \equiv R^*$ and $r(t) \equiv R^*$) that $v = R^*$ solves the problem

$$\max_{v \in V}[\eta \times \nabla H(R^*)] \cdot v \tag{19}$$

or, equivalently, the problem

$$\max_{v \in V}[\nabla H(R^*) \cdot v + \lambda^{min} \cdot v - \lambda^{max} \cdot v] \ , \tag{20}$$

where the vectors $\lambda^{min}, \lambda^{max} \in \mathcal{R}_+^I$ have the following components:

$$\lambda_i^{min} = \max\{(\eta_i - 1)H_i'(R_i^*), 0\} \geq 0,$$
$$\lambda_i^{max} = -\min\{(\eta_i - 1)H_i'(R_i^*), 0\} \geq 0 \ .$$

Adding the constant $-\lambda^{min} \cdot R^{min} + \lambda^{max} \cdot R^{max}$ to the objective function in (20), we see that $v = R^*$ maximizes the Lagrangian

$$\nabla H(R^*) \cdot v + \lambda^{min} \cdot (v - R^{min}) - \lambda^{max} \cdot (v - R^{max})$$

for the optimization problem

$$\max_{v \in V}[\nabla H(R^*) \cdot v] \tag{21}$$

subject to constraints

$$v \geq R^{min} \ \text{ and } \ v \leq R^{max} \ . \tag{22}$$

Moreover, the complimentary slackness conditions are satisfied for the (Lagrange multipliers) λ_i^{min} and λ_i^{max}. Indeed, if for some i we have $R_i^* > R_i^{\min}$, then $\tau_i^* \leq 0$ (otherwise, by (17)-(18), $\tau_i(t)$ could not possibly be constant), and therefore $\eta_i \leq 1$. This means that $R_i^* > R_i^{\min}$ implies $\lambda_i^{min} = 0$. Using an analogous argument, we see that $R_i^* < R_i^{max}$ implies $\lambda_i^{max} = 0$.

Thus, by the Kuhn-Tucker theorem (cf. [2]), $v = R^*$ solves the problem (21)-(22), which is equivalent to the problem

$$\max_{v \in V^{cond}} \nabla H(R^*) \cdot v \ . \tag{23}$$

This in turn means that point R^* is a maximal point of the set V^{cond} (i.e., it lies on its outer - "north-east" - boundary), and that vector $\nabla H(R^*)$ is normal to the (convex) set V^{cond} at point R^*. This implies that R^* is a solution to (1)-(4).

Since R^* solves the problem (19), R^* is a point on the outer boundary of the entire rate region V, i.e. $R^* \in V^*$. This implies that R^* belongs to the (non-empty) intersection of V^{cond} and V^*. ∎

Lemma 4: Suppose FSP (r, τ) is such that

$$r(t) \equiv R^* \ \text{ for all } t \geq 0$$

and $\tau(t)$ remains uniformly bounded for all $t \geq 0$. Then, R^* is a solution to the problem (1)-(4) and $R^* \in V^{cond} \cap V^* \neq \emptyset$.

Proof. As shown in the proof of Lemma 3, $v(t) \equiv R^*$. Then, it follows from (17)-(18) that $R_i^* \in [R_i^{\min}, R_i^{\max}]$ for each i - otherwise $\tau_i(t)$ could not remain bounded. Consider a function $\tau_i(\cdot)$. If $\tau_i(0) \geq 0$ and $R_i^* = R_i^{\min}$ then (from (17)-(18)) $\tau_i(t) \equiv \tau_i(0)$ for $t \geq 0$. If $\tau_i(0) \geq 0$ and $R_i^* > R_i^{\min}$ then $\tau_i(t)$ will decrease linearly at the rate $R_i^{\min} - R_i^*$ until it hits 0, and then will stay at 0. Similarly, if $\tau_i(0) \leq 0$, $\tau_i(t)$ either stays at $\tau_i(0)$ (in the case $R_i^* = R_i^{\max}$) or increases linearly until it hits 0 and then stays at 0 (in the case $R_i^* < R_i^{\max}$). Thus, for some fixed $d \geq 0$ and a fixed vector τ^*, we must have $\tau(t) \equiv \tau^*$ for $t \geq d$. The time shifted path $(\theta_d r, \theta_d \tau)$ is also an FSP, and, as we have shown above, it is stationary. An application of Lemma 3 completes the proof. ∎

Proof of Theorem 1. For each integer $d \geq 0$, consider the FSP $(r^{(d)}, \tau^{(d)}) \doteq (\theta_d r, \theta_d \tau)$, which is a time shifted version of (r, τ). Since all component functions of all FSPs $(r^{(d)}, \tau^{(d)})$ are uniformly Lipschitz continuous (because $\|r(t)\|$ is uniformly bounded) and the sequence of functions $r^{(d)}(\cdot)$ converges uniformly to the function identically equal to R^*, we can choose a subsequence $(r^{(j)}, \tau^{(j)})$ converging (uniformly on compact sets) to a path (r°, τ°) such that $r^\circ(t) \equiv R^*$ and $\tau^\circ(\cdot)$ being uniformly bounded. But, the path (r°, τ°) is also an FSP. Application of Lemma 4 completes

the proof. ∎

VI. Simulations

A. Achieving minimum rates

In this section we report on simulation results for our algorithms. The DRC traces that we use are determined by a DRC predictor. At each time slot t, the predictor gives the value of $DRC_i(t)$ for each user i based on user position and a simulated channel fading process. The possible values for $DRC_i(t)$ are (in kbits per second), $\{0, 38.4, 76.8, 153.6, 307.2, 614.4, 921.6, 1228.8, 1843.2, 2457.6\}$. The average value of $DRC_i(t)$ for each user is presented in Figure 2.

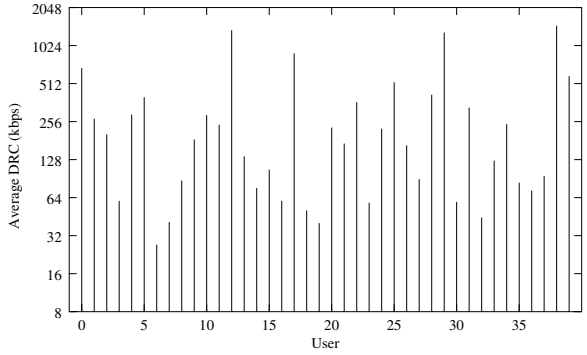


Fig. 2. The average DRC value for each user.

The 3G1xEV-DO system for high-speed data has two features that for simplicity of exposition we did not consider in the earlier theoretical sections of this paper. (In practice they have little effect on actual system performance.) First, if the basestation decides to transmit at a low rate (e.g. 38.4kbps), we are forced to assign multiple time slots (e.g. 16 slots) to the same user. Otherwise, the amount of data transmitted would be too small, which would lead to implementation problems. Second, if the basestation decides to serve user i, the actual data transmission rate might be slightly different from $DRC_i(t)$, due to the error-correcting coding schemes that are employed. In our simulations we do take these two features into account.

We also remark that PFMR and MTMR as described in Section IV are designed to provide constraints on the *long term* received service rates. In practice we wish to bound the service rate received over shorter time intervals. We achieve this by slightly increasing the token rate, R_i^{token}, when $T_i(t) \geq 0$. (For the definition of R_i^{token} see Equation 7.) In particular, for our simulations we set $R_i^{token} = 1.2 \times R_i^{\min}$ when $T_i(t) \geq 0$.

We run the traces for 90 seconds. At the end of each 10 second interval we calculate the average data rate that the user received during that interval. We then plot the cumulative distribution function of all these rates for all users. We discard the measurements for the first 10 seconds in order that our results are not skewed by transient effects.

In Figure 3 we show cumulative distribution functions of these rates on a logarithmic scale for 10, 20, 30 and 40 users. We show plots for Proportional Fair [20], [3], PFMR and MTMR. For the latter two algorithms we take $R_i^{\min} = 9.6$kbps for all i and so the token rate for each user is 1.2×9.6kbps = 11.52kbps. For these initial plots we do not impose a maximum rate constraint (i.e. we set $R_i^{\max} = \infty$ for all i). The length of the time slot is 1.667ms. The value of a_i is 6.25×10^{-5} for all users.

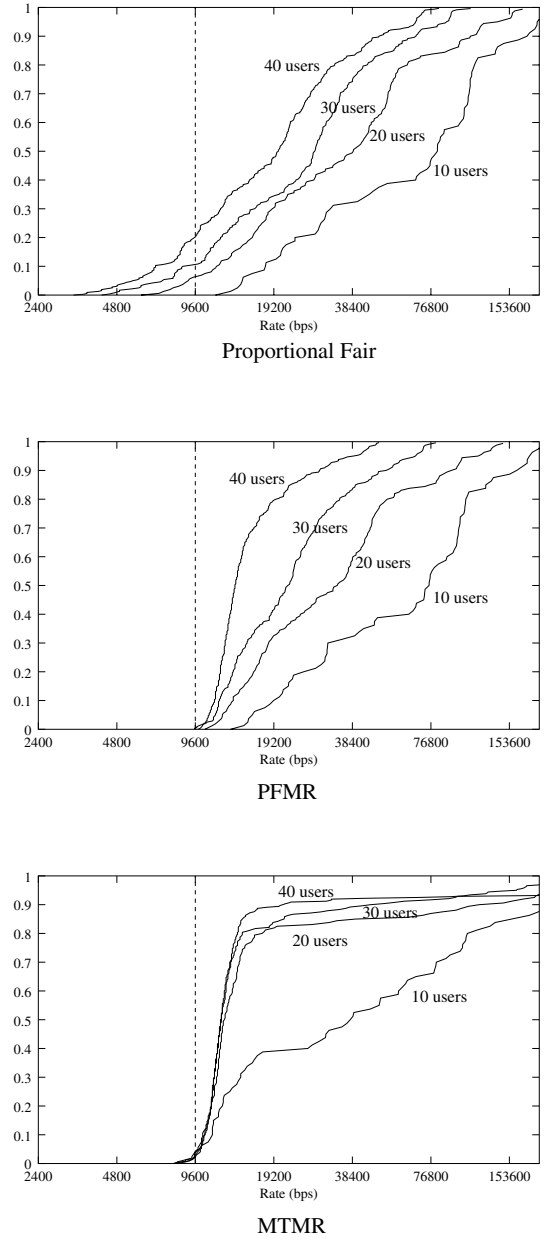


Fig. 3. (Top) Proportional Fair. (Middle) PFMR with $R_i^{\min} = 9.6$kbps. (Bottom) MTMR with $R_i^{\min} = 9.6$kbps.

The top plots show the cumulative distribution functions for Proportional Fair. We can see that as the number of users increases, the minimum of the rate distribution decreases. The middle plot shows the curves for PFMR. We see that now, regardless of the number of users, the minimum of the distribution is clipped at around 9.6kbps. We note that for

the case of 10 users, the curve for PFMR is almost identical to the curve for Proportional Fair. This is because for this small number of users, Proportional Fair already achieves $R_i^{\min} = 9.6$kbps for all users. Hence for PFMR the token levels remain small and so the two algorithms are essentially the same. However, for the cases of 20, 30 and 40 users, Proportional Fair cannot provide $R_i^{\min} = 9.6$kbps for all users. In this case, some of the token levels in PFMR rise to become nonzero so as to increase the rates of the low rate users. In the bottom plot we show the curves for MTMR. Here we also clip the minimum of the distribution. However, the curves are a different shape from the curves for PFMR. This is because more of the service is given to a few users with the best DRC values.

We recall that the aim of PFMR is to maximize $\sum_i \log R_i$ subject to the $R_i^{\min}$ constraints. The aim of MTMR is to maximize $\sum_i R_i$ subject to the $R_i^{\min}$ constraints. In the following table we show the values of these objective functions for the case of 30 users. We see that Proportional Fair has a slightly higher value of $\sum_i \log R_i$ than PFMR since it is not trying to satisfy the $R_i^{\min}$ constraints. We also note that PFMR has a higher value of $\sum_i \log R_i$ than MTMR but MTMR has a higher value of $\sum_i R_i$.

	$\sum_i \log R_i$	$\sum_i R_i$
Prop. Fair	303.1	940262
PFMR	300.4	776563
MTMR	291.5	1181069

In Figure 4 we change the minimum rate constraint so that $R_i^{\min} = 30.0$kbps for all users. We show cumulative distribution functions for 6, 8, 10 users under Proportional Fair, PFMR and MTMR. Once again, both PFMR and MTMR achieve the minimum rate constraints whereas Proportional Fair does not. (Note that we consider smaller numbers of users than before since *no algorithm* could provide a 30.0kbps rate constraint to larger numbers of users).

In Figure 5 we study how PFMR peforms when we have a maximum rate constraint. In particular we show simulation results for the PFMR algorithm with $R_i^{\min} = 9.6$kbps and $R_i^{\max} = 50.0$kbps for all users. (Compare this figure to the middle of Figure 3, which corresponds to exactly same simulation scenario, but with $R_i^{\max} = \infty$.) We see that users' throughputs are indeed "capped" at 50 kbps, as desired.

B. The token processes

In Figure 6 we illustrate the behavior of the token processes for PFMR in the 30 user case with $R_i^{\min} = 9.6$kbps and $R_i^{\max} = \infty$. User 7 has small DRC values and so it would not be able to achieve $R_7^{\min} = 9.6$kbps under Proportional Fair. Hence the token level for this user stabilizes in a range that is strictly above zero. This increases the likelihood that user 7 is served in each time slot. In contrast, user 29 has larger DRC values which means that it *would* achieve $R_{29}^{\min} = 9.6$kbps under Proportional Fair. For this user the token level falls to zero since it does not need "extra help" from the tokens to obtain the minimum service rate.

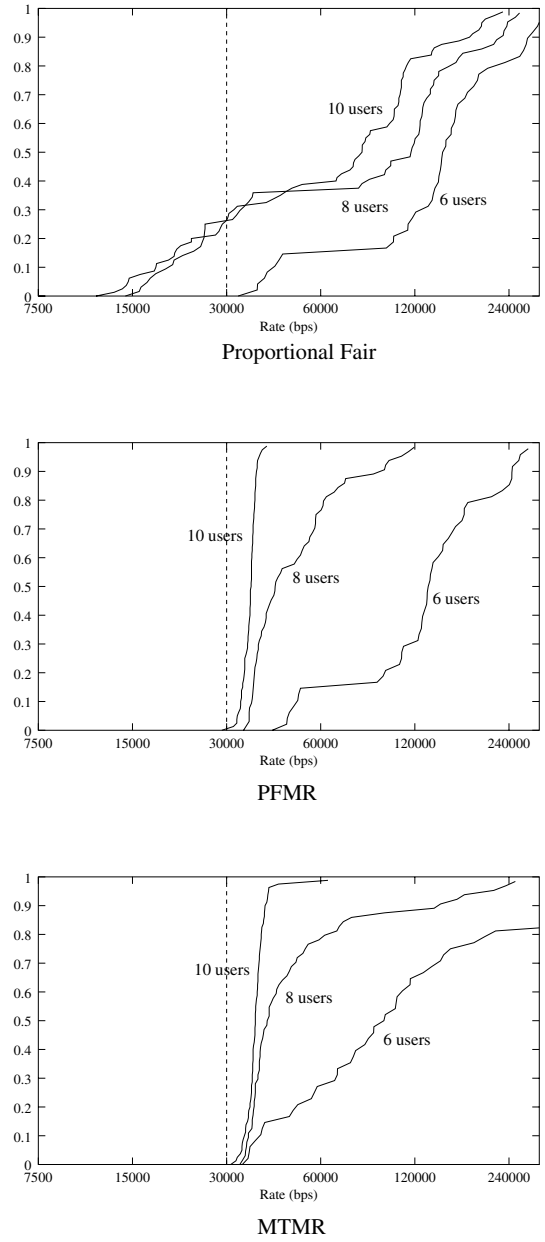


Fig. 4. (Top) Proportional Fair. (Middle) PFMR with $R_i^{\min} = 30.0$kbps. (Bottom) MTMR with $R_i^{\min} = 30.0$kbps.

We note that in any finite time interval, the user can actually get a rate that is slightly less than the token rate (which is 11.52kbps). If in some interval the token level rises then its service rate is slightly less than than the token rate. If the token level falls then the service rate is slightly higher than the token rate. We illustrate this in the plot of user 7 in Figure 6. Service rates are given for two intervals of length 10 seconds. This phenomenon is also the reason why the minimum rates for MTMR in Figure 3 are slightly less than 9.6kbps. However, we emphasize that for longer time intervals the minimum service rates become closer to the token rate whenever the token levels are bounded. In particular, consider a time interval $[t_1, t_2]$. Let $T(t_1)$ and $T(t_2)$ be the token levels at time t_1 and t_2,

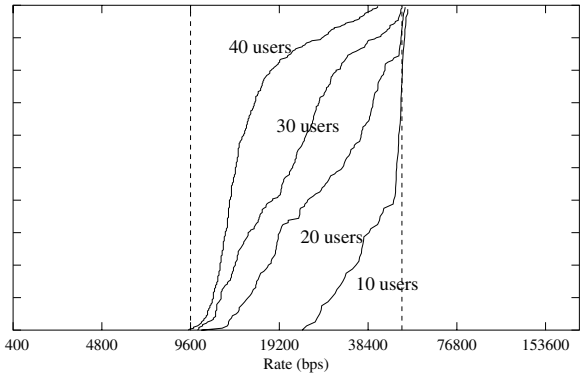
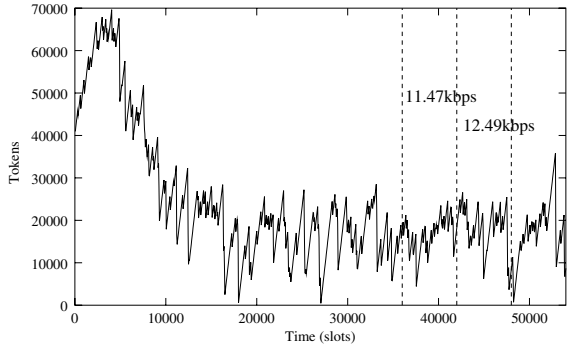
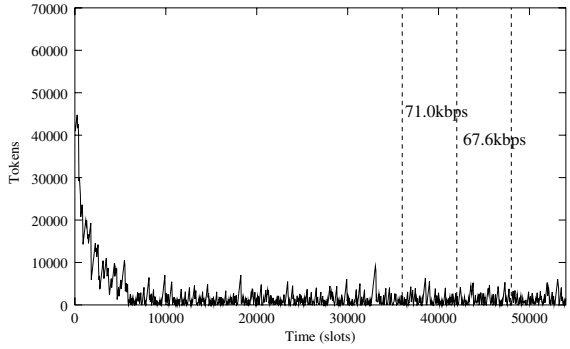


Fig. 5. PFMR with $R_i^{\min} = 9.6$kbps and $R_i^{\max} = 50.0$kbps.



User 7



User 29

Fig. 6. The token processes for users 7 and 29.

respectively, and assume for simplicity that $T(t)$ never hits 0 in this interval. Then the service rate that the user receives in the interval $[t_1, t_2]$ is equal to,

$$\text{(Token rate)} + \frac{T(t_1) - T(t_2)}{t_2 - t_1}.$$

If $T(t_1)$ and $T(t_2)$ remain bounded as $t_2 - t_1$ becomes large, then the second term approaches zero and so the service rate approaches the token rate. (If $T(t)$ sometimes hits 0 in $t_2 - t_1$, boundedness of $T(t_1)$ and $T(t_2)$ implies that the service rate becomes *at least* the token rate.)

C. Differentiating users

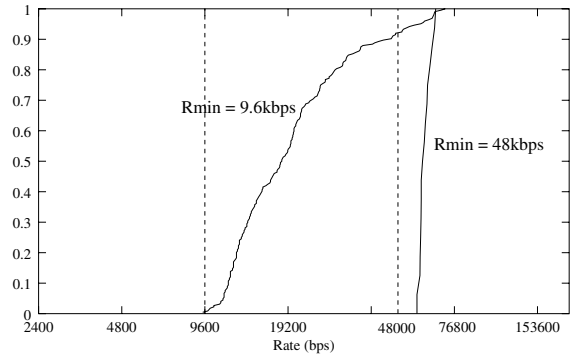


Fig. 7. Service rates for two classes of users.

In Figure 7 we show that by using PFMR we can achieve different values of $R_i^{\min}$ for different users. In particular we assume that in the 30 user case, two users have paid extra so that they are assigned an $R_i^{\min}$ value of 48kbps. The remaining users have $R_i^{\min} = 9.6$kbps. The token rate for the two "premium" users is 1.2×48kbps = 57.6kbps. We show in the figure that the distribution of rates for these users is lower bounded by 48kbps. For the remaining users, the minimum of the rate distribution is clipped around 9.6kbps.

D. Comparision to enforcing rate constraints via modified utility function

Recall that in Section I-C we discussed that a plausible method for achieving minimum rates would be to modify the utility function so that rate constraint violations are penalized. As an example, suppose that we modify Proportional Fair so that it aims to maximize the utility function,

$$H(R) =$$

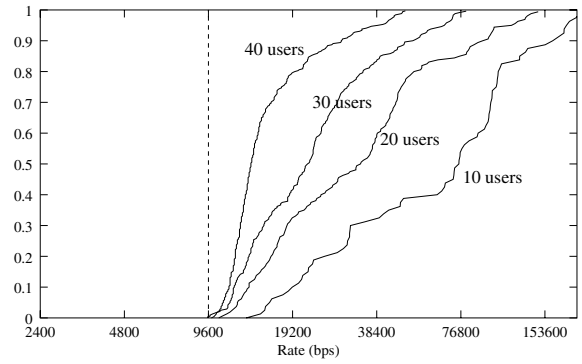
$$\sum_i \left(\log R_i - \frac{1}{3} \left[\max\{0, 1.2 \times R_i^{\min} - R_i\} \right]^3 \right).$$

(Here we use $1.2 \times R_i^{\min}$, as opposed to just $R_i^{\min}$, as a threshold below which we impose penalty on the "low" values of $R_i^{\min}$. The reason for doing that is the same as the reason for using $1.2 \times R_i^{\min}$ as the token rate in PFMR.) The corresponding Gradient algorithm always tries to serve flow,

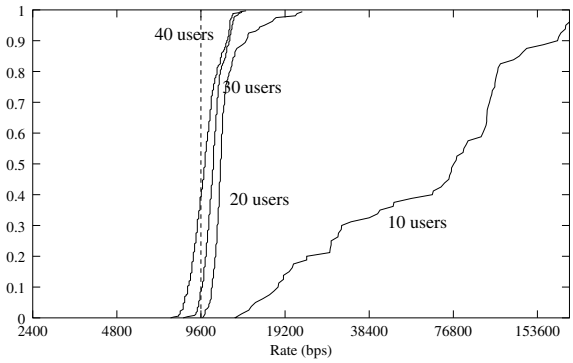
$$i \in \arg\max_{i \in I} \mathit{DRC}_i(t) \times$$

$$\left(\frac{1}{R_i(t)} + [\max\{0, 1.2 \times R_i^{\min} - R_i\}]^3 \right).$$

In Figure 8 we compare this algorithm with PFMR for the case of 30 users and $R_i^{\min} = 9.6$kbps. We can see that it is less effective than PFMR at providing minimum rates and achieves significantly less system throughput.



PFMR



Proportional Fair with Modified Utility Function

Fig. 8. Comparison of PFMR with a modified version of Proportional Fair that attempts to enforce minimum rates via a penalty in the utility function.

VII. Discussion

We have proposed the GMR algorithm and proved an optimality result showing that if the user throughputs converge, then the corresponding stationary throughputs do in fact maximize the desired utility function, subject to minimum/maximum rate constraints. The rate constraints are enforced via a very generic token counter mechanism. We note that token counters provide a very natural overload detection and control mechanism. A large positive value of a user's token counter indicates that this user "needs help" to reach the desired minimum throughput. A large number of such users indicates that air interface resources are "stretched" in trying to provide minimum rate for all users - this can serve as a trigger of an overload control action.

Our simulation results show good performance and robustness of the algorithm, which, along with its simplicity and "compatibility" with the widely employed Proportional Fair algorithm, make this algorithm very attractive for practical use.

References

[1] M. Andrews, K. Kumaran, K. Ramanan, A. L. Stolyar, R. Vijayakumar, P. Whiting. Scheduling in a Queueing System with Asynchronously Varying Service Rates. *Probability in Engineering and Informational Sciences*, vol.14, 2004, pp.191-217.

[2] P.E.Gill and W.Murray. *Numerical Methods for Constrained Optimization*. Academic Press, London, 1974.

[3] A.Jalali, R.Padovani, and R.Pankaj. Data Throughput of CDMA-HDR, a High Efficiency - High Data Rate Personal Communication Wireless System. In *Proc. of the IEEE Semiannual Vehicular Technology Conference, VTC2000-Spring,* Tokyo, Japan, May 2000.

[4] A. L. Stolyar. MaxWeight Scheduling in a Generalized Switch: State Space Collapse and Workload Minimization in Heavy Traffic. *Annals of Applied Probability*, vol.14(1), 2004, pp. 1-53.

[5] A.L. Stolyar. On the Asymptotic Optimality of the Gradient Scheduling Algorithm for Multi-User Throughput Allocation. *Operations Research*, vol.53(1), 2005, to appear.

[6] A. L. Stolyar. Maximizing Queueing Network Utility subject to Stability: Greedy Primal-Dual Algorithm. 2004, submitted.

[7] R. Chakravorty, S. Katti, I. Pratt and J. Crowcroft. Flow aggregation for enhanced TCP over wide area wireless. In *Proc. IEEE INFOCOM*, 2003.

[8] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Transactions on Automatic Control*, vol. 37, no. 12, pp. 1936 – 1948, December 1992.

[9] 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*, February 2001.

[10] S. Shakkottai and A. Stolyar, "Scheduling algorithms for a mixture of real-time and non-real-time data in HDR," in *Proceedings of 17th International Teletraffic Congress (ITC-17)*, Salvador da Bahia, Brazil, 2001, pp. 793 – 804.

[11] S. Shakkottai and A. L. Stolyar. Scheduling for Multiple Flows Sharing a Time-Varying Channel: The Exponential Rule. *Analytic Methods in Applied Probability. In Memory of Fridrih Karpelevich. Yu. M. Suhov, Editor.* American Mathematical Society Translations, Series 2, Volume 207, pp. 185-202. American Mathematical Society, Providence, RI, 2002.

[12] S. Borst and P. Whiting, "Dynamic rate control algorithms for CDMA throughput optimization," in *Proceedings of IEEE INFOCOM '01*, Anchorage, AK, April 2001.

[13] S. Borst, "User-level performance of channel-aware scheduling algorithms in wireless data networks," in *Proceedings of IEEE INFOCOM '03*, San Francisco, CA, April 2003.

[14] M. Andrews, "Instability of the proportional fair scheduling algorithm for HDR," *IEEE Transactions on Wireless Communications* 3(5), 2004.

[15] M. Andrews and L. Zhang, "Scheduling over non-stationary wireless channels with finite rate sets," In *Proceedings of IEEE INFOCOM '04*, Hong Kong, 2004.

[16] R. Agrawal, V. Subramanian. Optimality of Certain Channel Aware Scheduling Policies. In *Proc. of the 40th Annual Allerton Conference on Communication, Control, and Computing. Monticello, Illinois, USA, October 2002.*

[17] H. Kushner, P. Whiting. Asymptotic Properties of Proportional Fair Sharing Algorithms. In *Proc. of the 40th Annual Allerton Conference on Communication, Control, and Computing. Monticello, Illinois, USA, October 2002.*

[18] X.Liu, E.K.P.Chong, N.B.Shroff. Opportunistic Transmission Scheduling with Resource-Sharing Constraints in Wireless Networks. *IEEE Journal on Selected Areas in Communications*, 19(10), 2001, pp. 2053-2064.

[19] X.Liu, E.K.P.Chong, N.B.Shroff. A Framework for Opportunistic Scheduling in Wireless Networks. *Computer Networks*, vol.41, 2002, pp.451-474.

[20] P. Viswanath, D. Tse and R. Laroia. Opportunistic Beamforming using Dumb Antennas. *IEEE Transactions on Information Theory*, 48(6), 2002.