Transmission Scheduling for Efficient Wireless Utilization

Xin Liu, Edwin K. P. Chong, and Ness B. Shroff
School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN 47907-1285
{xinliu, echong, shroff}@ecn.purdue.edu

Abstract— **We present an "opportunistic" transmission scheduling policy that exploits time-varying channel conditions and maximizes the system performance stochastically under a certain resource allocation fairness constraint. We establish the optimality of the scheduling scheme and also describe a practical scheduling procedure to implement our scheme. Through simulation results, we show that the scheme also works well for nonstationary scenarios and results in performance improvements of 20–150% compared with a scheduling scheme that does not take into account channel conditions. Furthermore, we note that in wireless networks, an important role of resource allocation is to balance the system performance and fairness among "good" and "bad" users. We propose three heuristic time-fraction assignment schemes, which approach the problem from different viewpoints.**

Keywords—**Scheduling, fairness, wireless, high-rate-data.**

I. Introduction

FUTURE wireless networks are designed to support wideband data communications as well as voice communications. Such a network will be the basis for a wireless information society where access to information and information services such as electronic commerce is available anytime, anywhere, and to anybody. Throughout the world, the demand for wireless communication systems has increased exponentially in the last few years. Because wireless frequency is a scarce resource, efficient frequency utilization is becoming increasingly important. Resource allocation schemes and scheduling policies are critical to achieving these goals.

In wireline networks, resource allocation schemes and scheduling policies play important roles in providing service performance guarantees, such as throughput, delay, delay-jitter, fairness, and loss rate [1]. However, resource allocation and scheduling schemes from the wireline domain do not carry over to wireless systems because wireless channels have unique characteristics not found in wireline channels, such as limited bandwidth, time-varying and location-dependent channel conditions, and channel-condition-dependent throughput.

In wireless networks, the channel conditions of mobile users are time-varying. Radio propagation can be roughly characterized by three nearly independent phenomena: path-loss variation with distance, slow log-normal shadowing, and fast multipath-fading. Path losses vary with the movement of mobile stations. Slow log-normal shadowing and fast multipath-fading are time-varying with different time scales. Thus, users perceive time-varying service quality and/or quantity because channel conditions are time-varying. For voice users, better channel conditions may result in better voice quality. For packet data service, better channel conditions (or larger signal to interference plus noise ratio (SINR)) can be used to provide higher data rates by reducing coding or spreading and/or increasing the constellation density. Previous research shows that cellular spectral efficiency (in terms of b/s/Hz/sector) can be increased by a factor of two or more if users with better links are served at higher data rates [2]. Procedures to exploit this are already in place for all the major cellular standards: adaptive modulation and coding schemes are implemented in the 3G TDMA standards, and variable spreading and coding are implemented in the 3G CDMA standards. In general, a user is served with better quality and/or at a higher bit rate when the channel condition is better.

On one hand, good scheduling schemes should be able to exploit the time-varying channel conditions of users to achieve higher utilization of wireless resources. On the other hand, the potential to exploit higher data throughputs in an opportunistic way, when channel conditions permit, introduces the tradeoff problem between wireless resource efficiency and fairness among users. Because wireless spectrum is a scarce resource, improving the efficiency of spectrum utilization is important, especially to provide high-rate-data service. Hence, we cannot expect the same throughput for all users because the users in general can have very different channel conditions. However, a scheme designed only to maximize the overall throughput could be very unfair among users, especially users with widely disparate distances from the base station. For example, allowing only users close to the base station to transmit with high transmission power may result in very high throughput, but is unfair to other users. This basic dilemma motivates our work: to improve wireless resource efficiency by exploiting time-varying channel conditions while at the same time control the level of fairness among users.

In this paper, we consider a time-slotted system in which time is the resource to be shared among the users. Associated with each user is a number between 0 and 1 representing the long-term fraction of time to be assigned to the user. This "time-fraction assignment" to users represents the fairness constraint. Given this fairness constraint, the problem is to determine which user should be scheduled to transmit at each time slot so that network performance is optimized. To solve this problem, we present an opportunistic scheduling scheme that maximizes the wireless resource utilization by exploiting time-varying channel conditions, taking into account the fairness constraint (i.e., our scheme is optimal among all schemes that satisfy the fairness constraint). The time-fraction assignment representing the fairness constraint can be obtained as a byproduct of resource allocation schemes, which we will describe in Section III-D.

Recently, the authors of [3], [4], [5] have studied wireless fair scheduling policies. They extend scheduling policies for wireline networks to wireless networks which provide various degrees of performance guarantees, including short-term and long-term fairness, as well as short-term and long-term throughput bounds. However, they model a channel as either "good" or "bad," which might be too simple to characterize realistic wireless channels, especially for data service.

In [6], [7], the authors present a scheduling scheme for the Qualcomm/HDR system which satisfies the following fairness property: if another scheduling algorithm is used to increase the throughput of a specific user by $x\%$ over what that user receives under the HDR scheduling algorithm, the sum of all the percentage decreases suffered by the throughputs of all the other users under the new algorithm will be more than $x\%$ [7]. This property is known as *proportional fairness* [8]. The HDR algorithm also exploits time-varying channel conditions while maintaining proportional fairness. However, their measure of fairness is different from ours, as is their objective function. Our structure of fairness is more flexible — the system can explicitly set the fraction of time assigned to each user. Furthermore, our scheme outperforms the HDR scheduler in terms of the overall throughput in all cases, although there is no guarantee that a single user performs better.

The paper is organized as follows. In Section II, we introduce the system model. In Section III, we present our opportunistic scheduling policy and the proof of its optimality is attached in the appendix. We also describe a practical scheduling procedure to implement our scheme. Furthermore, we give three heuristic time-fraction assignment schemes. In Section IV, we show simulation results to illustrate the performance of our scheduling policy. Conclusions and future work are presented in Section V.

II. System Model

We consider a time-slotted system—time is the resource to be shared among all users. At any given time, only one user can occupy a given channel (frequency band) in a cell; multiple users could transmit at different frequencies. We focus on the scheduling problem for one channel. Note that a channel in this context could be very large. For example, it is possible for 10 users to share 1MHz frequency band for high rate data service while in IS-136 standard, a voice channel takes 10KHz bandwidth. The time-fraction assignment scheme dictates the fraction of time that a user should transmit on the channel. The scheduling algorithm then decides which time slot should be assigned to which user, given the time-fraction assignment. This time-fraction assignment can be viewed as the fairness requirement in the system.

As explained previously, channel conditions in wireless networks are time-varying, and thus users experience time-varying performance. We use a stochastic model to capture the *time-varying* and *channel-condition-dependent* performance of each user. Specifically, let $\{U_i^k\}$ be a stochastic process associated with user i, where U_i^k is the level of performance that would be experienced by user i if it is scheduled to transmit at time k. The value of U_i^k measures the "worth" of time-slot k to the user i, and is in general a function of its channel condition. The value of U_i^k could also depend on other factors, such as user i's throughput and power consumption. Usually, the better the channel condition of user i, the larger the value of U_i^k.

Recently, the notion of "utility" has been used to study wireless resource management problems [9], [10]. Utility is generally defined as a measure of satisfaction that a user derives from accessing the wireless resource (in terms of bandwidth, power, etc). Along these lines, we can think of the value of U_i^k as the "utility value" of the channel to user i at time k. However, unlike the typical usage of utility, note that in this paper, we model a user's channel-utility value as a stochastic process, capturing the important feature of wireless systems that channel conditions are time-varying.

Next, we present some examples of possible performance measures. The most straightforward performance measure is the throughput (in terms of bits/sec) or the "monetary value" of the throughput (in terms of dollars/sec), where the throughput is the number of information bits successfully transmitted between the base station and the mobile user per time slot. Usually, a user's throughput is a nondecreasing function of the signal to interference plus noise ratio (SINR). Depending on the class of a user, the throughput could be a step function, an S-shape function, or a linear function of the SINR, as shown in Figure 1. Hence, different classes of users may have different throughput values even with the same channel condition. In our system model, we do not make any assumptions on the physical-layer implementation of the system. Note also that the throughput of a user could be limited by the user's interference to other cells. For example, consider a user at the edge of a cell, where the user's transmission causes significant interference to a neighboring cell. When the neighboring cell is heavily load, the user's maximum transmission power may have to be limited to avoid undue interference to the neighboring cell.

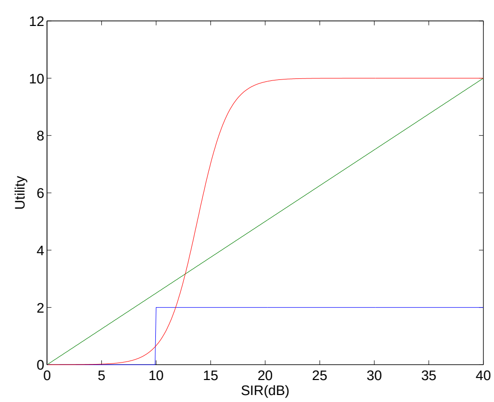

Fig. 1. Users' throughput as a function of SINR.

Besides throughput, other issues could also be important to users and different users could have different performance measures. For example, a user on a vehicle where there is no scarcity of power may be concerned only about the throughput. On the other hand, power consumption is very important to a handset user, and hence the performance of such a user could have the form:

$$\text{value of throughput} - \text{cost of power consumption.}$$

In summary, the performance value U_i^k is an abstraction used to capture the time-varying and channel-condition-dependent "worth" of a time slot to a user. The use of such a general performance model frees us from physical-layer implementation details and allows us to focus on the problem of designing scheduling policies. We assume throughout that performance values for different users are *comparable and additive*. For example, the unit of the performance values could be "dollars per time unit" for all users. In other words, if $U_1^k > U_2^k$, then assigning time slot k to user 1 rather than user 2 will lead to higher system performance. Also, if U_1^1 and U_1^2 are the performance values for user 1 at time slots 1 and 2, then the total performance over the two time slots is $U_1^1 + U_1^2$.

We consider both the uplink and the downlink of a wireless network. In both cases, the base station serves as the scheduling agent. The scheduling scheme does the following: at the beginning of a time slot, the scheduler (i.e., the base station) decides which user should be assigned the time slot based on the performance values of the users at that time slot. (We describe a particular scheduling procedure in Section III-C, including how the scheduler obtains information about the users' performance values.) For the uplink case, if a user is assigned a time slot, the user will transmit in that time slot. For the downlink case, if a user is assigned a time slot, the base station will transmit to the user in that time slot. If time slot k is assigned to user i, the system is "rewarded" with a performance value of U_i^k; i.e., user i's performance value at time slot k. *The goal of the scheduling scheme is to maximize the average system performance by exploiting the time-varying channel conditions, given the time-fraction assignment.* Basically, the scheduling policy systematically assigns a time slot to a user with a performance value that is large relative to those of the other users, while satisfying the time-fraction requirements of users (fairness constraint).

Our scheduling scheme could be implemented in TDMA/FDMA systems as well as time-slotted CDMA systems. The length of a time slot in the scheduling policy can be different from an actual time slot of a physical channel. The length of a scheduling time slot depends on how fast the channel conditions vary and how fast we want to track the variation. As mentioned in [11], it is necessary to "track" (at least slow fading) signal-level variations for better network performance.

III. Optimal Scheduling Policy

In this section, we describe the scheduling problem and our scheduling scheme. Let r_i denote the time-fraction assigned to user i, where $\sum_{i=1}^N r_i = 1$ and N is the number of users in the cell. Here, we assume that the r_i's are predetermined and serve as a prespecified fairness constraint — on average, a fraction r_i of the whole time should be scheduled to user i. Our goal is to develop a scheduling scheme that exploits the time-varying channel conditions to maximize the system performance, under the time-fraction constraints r_i, $i = 1, \cdots, N$.

Let $\vec{U}^k = (U_1^k, \cdots, U_N^k)$ be the *performance vector* at time slot k, where U_i^k is the performance value achieved by user i if time slot k is assigned to user i. We assume that U_i^k is nonnegative and bounded. We assume that $\{\vec{U}^k\}$ is stationary, so that the time index k can be dropped. Specifically, we use the notation $\vec{U} = (U_1, \cdots, U_N)$, where U_i is a random variable representing the performance value of user i at a generic time slot.

The scheduling problem is stated as follows: given $\vec{U}$, determine which user should be scheduled (in the given time slot). We define a *policy* Q to be a mapping from the performance-vector space to the index set $\{1, 2, \cdots, N\}$. Given $\vec{U}$, the policy Q determines the user to be scheduled: if $Q(\vec{U}) = i$, then user i should use the time slot, and the system receives a performance "reward" of $U_{Q(\vec{U})}$ (i.e, U_i). Hence, $E(U_{Q(\vec{U})})$ is the average system performance value associated with policy Q. Note that the policy Q is potentially "opportunistic" in the sense that it can use information on the performance vector $\vec{U}$ to decide which user to schedule.

We are interested only in policies that result in satisfaction of the time-fraction assignment constraints. Specifically, we say that a policy Q is *feasible* if $P\left\{Q(\vec{U}) = i\right\} = r_i$ for all $i = 1, \cdots, N$. Feasible policies are those that obey the given fairness constraints. We use Θ to denote the set of all feasible policies.

Our goal is to find a feasible policy Q that maximizes the average system performance while satisfying the fairness constraints. The problem can be stated formally as follows:

$$\underset{Q \in \Theta}{\text{maximize}}\, E\left(U_{Q(\vec{U})}\right). \tag{1}$$

Note that we can write

$$\begin{aligned} E\left(U_{Q(\vec{U})}\right) &= E\left(\sum_{i=1}^N U_i \mathbf{1}_{\{Q(\vec{U})=i\}}\right) \\ &= \sum_{i=1}^N E\left(U_i \mathbf{1}_{\{Q(\vec{U})=i\}}\right), \end{aligned}$$

where

$$\mathbf{1}_A = \begin{cases} 1 & \text{if } A \text{ occurs,} \\ 0 & \text{otherwise} \end{cases}$$

is the indicator function of the event A. In other words, the overall objective function is the sum of all users' average performance values (where we reap a reward of U_i only if user i is scheduled).

Recall that we assumed the sequence $\{\vec{U}^k\}$ to be stationary. This assumption does not preclude correlations across users or across time. In practice, a user's channel condition is usually time-correlated, for example, due to shadowing. Hence, a user's performance is usually also time-correlated. Furthermore, the performance of different users may also be correlated. For example, when the intercell interference is high, most users' performance values simultaneously decrease.

In the following, we first present our opportunistic scheduling policy given a time-fraction assignment. We attach the proof of its optimality in the appendix. Then we explain how to estimate the parameters used in the policy. Finally, we describe a procedure to implement our scheduling policy by tuning the parameter values based on measurements.

A. Opportunistic Scheduling Policy

A.1 2-user Case

For the purpose of illustration, we start with the 2-user case. Suppose that user 1 and user 2 have time-fraction assignments

r_1 and r_2, respectively, and $r_1 + r_2 = 1$. We wish to find an opportunistic policy that solves (1).

Define $y(v) = P\{U_1 + v \geq U_2\}$, where $v \in \mathbb{R}$. Because $y(v)$ is the distribution function of the random variable $(U_2 - U_1)$, $y(v)$ is a right-continuous monotonically increasing function of v with $y(\infty) = 1$ and $y(-\infty) = 0$. Hence, there exists a v^* (which may not be unique) such that for any $\epsilon > 0$,

$$y(v^* - \epsilon) \leq r_1 \leq y(v^*),$$

where r_1 is the time-fraction assignment of user 1.

We consider the scheduling policy under two conditions.

1. $y(v^*) = r_1$: The opportunistic scheduling policy in this case is given by

$$Q^*(\vec{U}) = \begin{cases} 1 & \text{if } U_1 + v^* \geq U_2, \\ 2 & \text{otherwise.} \end{cases}$$

2. $y(v^*) > r_1$: Let $y^-(v) = P\{U_1 + v > U_2\}$, which is a left-continuous monotonically increasing function of v. So

$$y^-(v^*) \leq r_1 < y(v^*);$$

i.e., $y(v^*) - y^-(v^*) = P\{U_1 + v^* = U_2\} > 0$. Let $p = (r_1 - y^-(v^*))/(y(v^*) - y^-(v^*))$. Note that $0 \leq p \leq 1$. The opportunistic scheduling policy is then given by

$$Q^*(\vec{U}) = \begin{cases} 1 & \text{if } U_1 + v^* > U_2 \\ 1 & \text{with prob. } p \text{ if } U_1 + v^* = U_2 \\ 2 & \text{with prob. } 1 - p \text{ if } U_1 + v^* = U_2 \\ 2 & \text{if } U_1 + v^* < U_2 \end{cases}$$

It is clear that the policy $Q^*(\vec{U})$ defined above is feasible:

$$P\{Q^*(\vec{U}) = 1\} = P\{U_1 + v^* > U_2\} + P\{U_1 + v^* = U_2\}p = r_1.$$

The policy can be described as follows. The space spanned by U_1 and U_2 is divided into two halves by the line $U_1 + v^* = U_2$. Above the line (i.e., $U_2 > U_1 + v^*$), we always schedule user 2 to transmit. Under the line (i.e., $U_1 + v^* > U_2$), we always schedule user 1 to transmit. If the probability of the line is positive, some randomization is needed if we fall on the line—with probability p, we schedule user 1 and with probability $1-p$, we schedule user 2, where $p = (r_1 - y^-(v^*))/(y(v^*) - y^-(v^*))$ is determined by the time-fraction assignment constraint.

A.2 General Case

Now we extend the policy from the previous section to the N-user case. Define

$$y_i(\vec{v}) = P\{U_i + v_i \geq \max_{j \neq i}(U_j + v_j)\}, \quad \text{for } i = 1, \cdots, N,$$

where $\vec{v} = (v_1, \cdots, v_N)$. Note that $y_i(\vec{v})$ is a monotonically-increasing right-continuous function of v_i and a monotonically-decreasing left-continuous function of v_j, $j \neq i$. There exists a $\vec{v^*}$ that satisfies $P\{Q^*(\vec{U}) = i\} = r_i$, where the opportunistic policy is

$$Q^*(\vec{U}) = \operatorname*{argmax}_i (U_i + v_i^*). \tag{2}$$

In the argmax above, we break ties probabilistically by picking a user i among those that achieve the maximum above with probability

$$\frac{r_i - P\{U_i + v_i^* > \max_{j \neq i}(U_j + v_j^*)\}}{P\{U_i + v_i^* = \max_{j \neq i}(U_j + v_j^*)\}}. \tag{3}$$

Note that $\vec{v}^*$ is not unique. There are N components but only $N-1$ independent constraint equations: $P\{Q^*(\vec{U}) = i\} = r_i$, for $i = 1, \cdots, N-1$, and $P\{Q^*(\vec{U}) = N\} = 1 - \sum_{i=1}^{N-1} r_i$ is a linear combination of the first $N-1$ equations, we can simply set $v_N = 0$.

The policy Q^* is a solution to the problem defined in Eq. (1); i.e., it maximizes the average system performance under the fairness constraint. The parameter $\vec{v}^*$ is the "offset" used to satisfy the time-fraction assignment constraint. Under this constraint, the scheduling policy schedules the "relatively-best" user to transmit. User i is the "relatively-best" user if $U_i + v_i^* \geq U_j + v_j^*$ for all j. In a special case where $v_j^* = 0$ for all j, the scheduling policy reduces to $Q^*(\vec{U}) = \operatorname{argmax}_i U_i$; i.e., always schedule the user with the largest performance value to transmit.

B. *Parameter Estimation*

Basically, the opportunistic scheduling policy is given by

$$Q^*(\vec{U}) = \operatorname*{argmax}_i (U_i + v_i^*),$$

where the v_i^*s are parameters determined by the distribution of $\vec{U}$. In practice, this distribution is unknown, and hence we need to estimate the parameters v_i^*, $i = 1, \cdots, N$. Figure 2 shows a block diagram of a practical scheduling procedure that incorporates on-line estimation of these parameters.

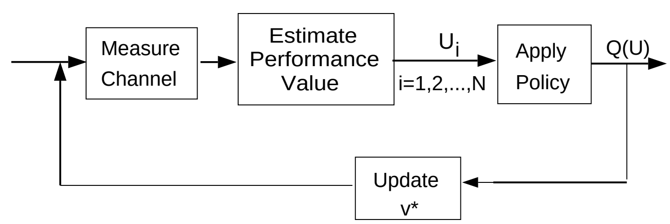

Fig. 2. Block diagram of the scheduling policy with on-line parameter estimation

In this section, we focus on the block that implements the on-line estimation of the parameters v_i^*, $i = 1, \cdots, N$, labeled "Update v^*" in Figure 2. We use a standard stochastic approximation algorithm to estimate $\vec{v}^*$ (the vector of the v_i^*s).

We first explain intuitively the idea of the stochastic approximation algorithm used in this paper. For a systematic and rigorous study of stochastic approximation algorithms, see [13], [14]. Suppose we want to solve the root-finding problem $f(x^*) = 0$, where f is a continuous function with one root x^* (both x^* and $f(x^*)$ are vectors of the same dimension). If we can evaluate the value of $f(x)$ at any x, then we can use the iterative algorithm

$$x^{k+1} = x^k - a^k f(x^k),$$

which will converge to x^* as long as the step size a^k is appropriately chosen; e.g., $a^k = 1/k$. Suppose that we cannot obtain exactly the value of $f(x^k)$ at x^k, but instead we only have a noisy observation g^k of $f(x^k)$ at x^k; i.e., $g^k = f(x^k) + e^k$ where e^k is the observation error (noise). In this case, it is well-known that if $E(e^k) = 0$ (i.e., the mean of the observation error is zero), then the algorithm

$$x^{k+1} = x^k - a^k g^k,$$

converges to x^* with probability 1 under appropriate conditions on a^k and f (see, e.g., [13], [14]).

In this paper, we use a stochastic approximation algorithm to estimate $\vec{v}^*$. For this, note that we can write $\vec{v}^*$ as a root of the equation $f(\vec{v}^*) = 0$, where the ith component of $f(\vec{v}^*)$ is given by

$$f_i(\vec{v}^*) = P\{Q^*(\vec{U}) = i\} - r_i, \qquad i = 1, \cdots, N$$

and

$$Q^*(\vec{U}) = \underset{i}{\operatorname{argmax}}(U_i + v_i^*).$$

We use a stochastic approximation algorithm to generate a sequence of iterates $\vec{v}^1, \vec{v}^2, \cdots$ that represent estimates of $\vec{v}^*$. Each $\vec{v}$ defines a policy Q^k given by

$$Q^k(\vec{U}) = \underset{i}{\operatorname{argmax}}(U_i + v_i^k).$$

To construct the stochastic approximation algorithm, we need an estimate g^k of $f(\vec{v}^k)$. Note that although we cannot obtain $f(\vec{v}^k)$ directly, we have a noisy observation of its components:

$$g_i^k = \mathbf{1}_{\{Q^k(\vec{U})=i\}} - r_i, \qquad i = 1, \cdots, N.$$

The observation error in this case is

$$e_i^k = g_i^k - f_i(\vec{v}^k) = \mathbf{1}_{\{Q^k(\vec{U})=i\}} - P\{Q^k(\vec{U}) = i\},$$

and thus we have $E(e_i^k) = 0$. Hence, we can use a stochastic approximation algorithm of the form

$$v_i^{k+1} = v_i^k - a^k \left(\mathbf{1}_{\{Q^k(\vec{U})=i\}} - r_i \right),$$

where $a^k = 1/k$. For the initial condition, we can set v_i^1 to be 0, or some estimate based on the measurement history. For the above algorithm, following the standard proof of [13], we can show that $\{v_i^k\}$ converges to v_i^* with probability 1. Furthermore, to accelerate and to reduce the range of the fluctuation the convergence of the stochastic approximation algorithm, we can use the standard technique of averaging (see, e.g., [14]):

$$\bar{v_i}^k = \left(1 - \frac{1}{k}\right) \bar{v_i}^{k-1} + \frac{1}{k} v_i^k.$$

Furthermore, simulations show that with the stochastic approximation algorithm, v_i^k converges to $\vec{v}^*$ relatively quickly.

When the U_i's are not continuous random variables, there may be "ties" in the argmax of Q^* in Eq. (2). Specifically, ties occur when $P\{U_i + v_i^* = \max_{j \neq i}(U_j + v_j^*)\} > 0$ for some i. In this case, v_i^k will still converge to v_i^*. However, we should break ties probabilistically by picking a user i among those that achieve the maximum with the probability shown in Eq. (3). In practice, we do not need to estimate this probability in Eq. (3) because the tie-break can be handled automatically by the adaptive nature of the stochastic approximation algorithm. To see this, imagine v_i^k fluctuating around v_i^* within a small range; when v_i^k is too large, we have $P\{Q = i\} > r_i$ and hence v_i^k will be dragged down. Our simulation results show that the stochastic approximation algorithm works well in both the continuous and "tie-break" cases — the system performance obtained with the stochastic approximation scheme is very close to that of the true optimal value while maintaining the fairness requirements.

C. Implementation Considerations

So far, we have described our scheduling policy, proved its optimality (in the appendix), and addressed the problem of estimating the parameter values needed for the policy. In this section, we explore some implementation considerations for our scheduling policy.

In our scheduling policy, the base station needs to obtain information of each user's performance value at a given time slot to make the scheduling decision.

The performance value of a user can be estimated either by the user or by the base station, based on the channel condition and/or measurements from previous transmissions. For the downlink case, a user could measure the received signal power level (from the user's base station) and the interference power level. The user could then calculate the performance value of the time slot based on the channel condition and other factors (such as power consumption). For example, suppose a user's performance is defined as its throughput, which is an S-shape function of the SINR, as shown in Figure 1. Based on the estimated SINR, the user can then obtain its performance value. For the uplink case, the base station could estimate the user's channel condition based on the received signal from the user. Assuming the base station knows the form of the performance value for each user (i.e., how the performance value depends on the SINR and/or other factors), the performance value could then be calculated by the base station.

If the performance value is estimated by the user, this information needs to be sent to the base station, which can be accomplished in several ways. For example, each user could maintain a small signaling channel with the base station. Alternatively, the required information could be piggybacked over the user's acknowledgment packets.

As mentioned before, the length of a time slot in our scheduling policy can be different from an actual time slot of the physical channel. The length of a scheduling time slot depends on how fast the channel condition varies and how fast we want to track the variation. The usual tradeoff between accuracy and signaling overhead exists here. Specifically, more frequent updating provides more accurate tracking of varying channel conditions, but incurs higher signaling costs. In practice, to decrease signaling costs, a user can update its information only when the change in the performance value is larger than a certain threshold. Furthermore, it is not necessary for all users to update at the same time.

In the following we summarize our scheduling procedure, which incorporates the on-line parameter estimation algorithm described in the last section. As mentioned before, the initial value of $\vec{v}^*$ can be set to $\vec{0}$ or some estimate based on history information. At each time slot $k = 1, 2, \cdots$, the system performs the following steps:

1. Estimate U_i^k;
• Uplink: the base station estimates each user's channel condition and calculates the values of U_i^k, $i = 1, \cdots, N$;
• Downlink: user i measures its channel condition, calculates U_i^k, and informs the base station;
2. The base station decides which user should be scheduled to transmit in the time slot based on the scheduling policy:

$$Q^k(\vec{U}^k) = \underset{i}{\mathrm{argmax}}(U_i^k + v_i^k);$$

3. The scheduled transmission takes place;
• Uplink: the base station broadcasts the ID of the selected user and the selected user transmits in the time slot.
• Downlink: the base station transmits to the selected user;
4. The base station updates the parameter vector $\vec{v}^{k+1}$ via

$$v_i^{k+1} = v_i^k - a^k \left(\mathbf{1}_{\{Q^k(\vec{U}^k)=i\}} - r_i \right);$$

For the stationary case, we set $a^k = 1/k$. For the nonstationary case, we set a^k to a small constant to track system variations. Note that the computation burden above is $O(N)$ per time slot, where N is the number of users sharing the channel (usually in the order of tens), which suggests that the procedure is easy to implement in practice.

D. Time-Fraction Assignment

In the previous section, our opportunistic scheduling scheme assumed a given time-fraction assignment. The time-fraction assignment $r_1, \ldots, r_N$ represents a prespecified allocation requirement. In the following, we describe three time-fraction assignment schemes, which approach the problem from different viewpoints.

User bidding: Suppose m_i is the amount of money that user i is willing to pay per unit time to access the wireless resources. The network then assigns time-fractions to users in proportion to their willingness to pay:

$$r_i = \frac{m_i}{\sum_{j=1}^N m_j}.$$

Hence, the more a user is willing to pay, the higher the fraction of the resources assigned to the user.

Fair sharing: If there are N users in the system, each user is assigned $r_i = 1/N$; i.e., each user receives the same share of resources. This scheme provides fair resource sharing assuming users are homogeneous. We can extend this scheme to the multi-class case as follows. Suppose there are L classes of users, where each class has l_i active users, and an associated weight w_i reflecting the importance and/or resource requirement of this class. Then the time-fraction assignment for a user in class i is:

$$r_i = \frac{w_i}{\sum_{j=1}^L w_j l_j}.$$

Bias sharing: The philosophy here is to allocate resources to users in proportion to their expected performance values. The corresponding time-fraction assignment is given by

$$r_i = \frac{E(U_i)}{\sum_{j=1}^N E(U_j)}.$$

This scheme clearly favors users with high performance values, but at the same time does not totally ignore the requirement of users with poor performance values.

IV. Simulation Results

In this section, we present numerical results from computer simulations of our scheduling scheme. Our scheduling policy exploits time-varying channel conditions—the policy dynamically decides which user should be scheduled to transmit in a time slot based on users' current performance values. For the purpose of simulations, we assume that the time-fraction assignment is done using *fair sharing*, i.e., the total resources are evenly divided among the users. The well known *round-robin* scheme is a policy that shares the resource (time in this case) in this manner, but does not exploit channel conditions. To evaluate the performance gain of our dynamic and opportunistic assignment of transmissions, we compare the performance of our policy with that of the round-robin scheme.

Our simulation environment is described in the following. We consider a multi-cell system consisting of a center cell surrounded by hexagonal cells of the same size. The base station is at the center of each cell and simple omni-directional antennas are used by mobiles and base stations. We focus on the performance of the downlink of the center cell because downlink communication is more important for data services. The frequency reuse factor is 3 and co-channel interference from the six first-ring neighboring cells is taken into account. We assume that each cell has a fixed number of frequency bands. Usually there are tens of users in each cell sharing different frequency bands. We focus on one frequency band, which is shared by 25 users in the central cell. The scheduling policy decides which user should transmit in this frequency band at each time slot. The users have exponentially distributed "on" and "off" periods.

We model user mobility as follows. The velocities of mobile users are independent random variables uniformly distributed between the minimum (2km/h) and the maximum velocity (100km/h). The directions of mobile users are independent random variables uniformly distributed between 0 and 2π. A mobile user chooses its velocity when it becomes active and the velocity is fixed during that on-period. The direction of a mobile user changes periodically. When a user becomes active, its location is uniformly distributed in the cell. If a user moves out of the border, we assume that it reappears at a point that is symmetric to the exiting point about the center base station.

As mentioned earlier, for our simulation experiments, we use the *fair sharing* time-fraction assignment scheme. When the number of active users N changes, i.e., when an active user becomes inactive or vice versa, we update the time-fraction assignment r_i for all active users. In other words, if N is the number of active users sharing the channel in the central cell, then we set $r_i = 1/N$, for all $i = 1, \cdots, N$.

The channel gains of the users are mutually independent random processes determined by the sum of two terms: one due to path (distance) loss and the other to shadowing. To be conservative, we assume, in the simulation, that the effects of fast multipath fading are averaged out via interleaving, diversity, etc. because current standards have not specified faster than frame-rate adaptation. However, we should note that fast fading is considered in the Qualcomm/HDR proposal [6]. In fact, if fast fading could be tracked, our scheme should provide even higher performance improvements than shown here.

We adopt the path-loss model (Lee's model) and the slow lognormal shadowing model in [15]. Specifically, the channel gain $g(k)$ (in dB) at time slot k between an arbitrary user at a distance d from a base station is given by:

$$g(k) = l_p(k) + s(k).$$

where $l_p(k)$ and $s(k)$ are terms representing path-loss and shadowing, respectively. The path loss $l_p(k)$ (dB) is obtained as

$$l_p(k) = K + 38.4\log_{10}(d(k)) - \alpha_0,$$

where α_0 is a correction factor used to account for different base station and mobile station (MS) antenna heights, transmit powers, and antenna gains, and $K = 103.41$ is a constant in the simulation assuming that the transmission power of a base station is fixed at 10W.

Shadowing is the result of the transmitted signal passing through or reflecting off some random number of objects such as buildings, hills, and trees. The shadowing term $s(k)$ is usually modeled as a zero-mean stationary Gaussian process with autocorrelation function given by

$$E(s(k)s(k+m)) = \sigma_o^2 \xi_d^{vT/D},$$

where ξ_d is the correlation between two points separated by a spatial distance D (meters), and v is velocity of the mobile user. In our simulation, we use a value of $\sigma_o = 4.3$ dB, corresponding to a correlation of 0.3 at a distance of 10 meters, as reported by Gudmundson [16].

The parameters of the simulation and their values are summarized in Table I.

In the following, we describe the simulation procedure in detail. At the beginning of the simulation, we set $\vec{v}^1 = \vec{0}$. We maintain an ordered list of users in the system. Let N be the number of active users. Each active user has a time-fraction assignment of $r_i = 1/N$. At each time slot $k = 1, 2, \cdots$, the following steps are simulated:

1. If user i is active, we generate U_i^k. In our simulation, a user's performance is a function of its SINR, as shown in Figure 3. Each user measures the received power level from the central base station, and the interference power level received from neighboring cells. Based on these measurements, the user calculates the SINR, and thus the corresponding performance value as a function of SINR. Figure 3 shows the forms of the performance values used by different users. To avoid crowding the figure, we only show the performance functions of 8 users. The performance values of users 1 and 2 are step-functions of their SINR, and user 2 has a higher threshold than user 1. The performance values of users 3–4 are linear functions of their SINR (in dB), with different slopes. Users 5–8 have performance values that are S-shape functions of their SINR, with different parameters. Totally, there are 4 users with step-functions, 6 users with linear functions, and 15 users with S-shape functions in the simulation.

Cell radius	1000m
Propagation environments	North American suburb
Frequency	1845 MHz
Distance Exponent (β)	3.84
Height of base station antenna	38.4m
Height of MS antenna	1.5m
base station transmission power	10W
base station antenna gain	6dB above dipole gain
MS antenna gain	0dB above dipole gain
Shadowing standard deviation	4.3dB
Shadowing correlation distance	10m
Background noise power	120dBm
Minimum speed	2km/h
Maximum speed	100km/h
Period for direction change	127 steps
Sample step size	10ms
Mean duration of on-period	5000 steps (50s)
Mean duration of off-period	2500 steps (25s)
Stochastic approximation step	0.01

TABLE I
SIMULATION PARAMETERS AND VALUES

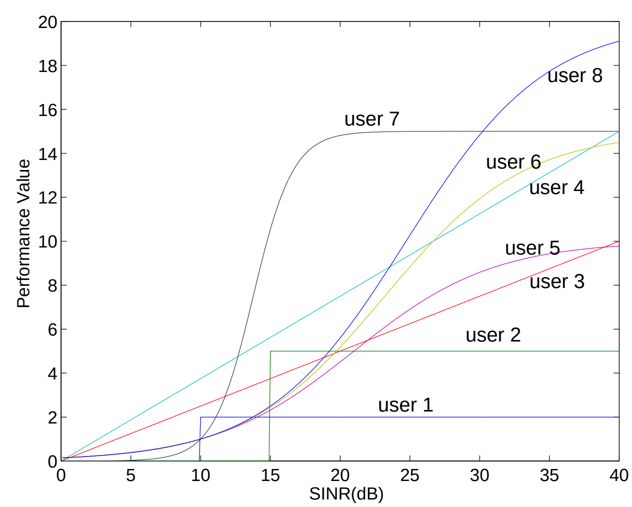

Fig. 3. Users' performance values as a function of SINR.

2. Active users transmit their values of U_i^k to the base station through a signaling channel.

3. Based on the vector of performance values $\vec{U}^k$, the base station decides which user to schedule in the time slot:

$$Q^k(\vec{U}^k) = \underset{i \in A}{\operatorname{argmax}}(U_i^k + v_i^k),$$

where A is the index set of all active users.

4. If user $j = Q(\vec{U}^k)$ is the selected user, then the base station transmits to user j in the time slot k. The system receives a performance "reward" equal to the performance value U_j^k.
5. In the round-robin scheduling scheme, we set J to be the index of the next active user in our ordered list of users, and let user J transmit. The system receives a performance "reward" equal to the performance value U_J^k of user J.
6. The base station updates $\vec{v}^{k+1}$ for all active users as follows:

$$v_i^{k+1} = v_i^k - a^k \left(\mathbf{1}_{\{Q^k(\vec{U}^k)=i\}} - r_i \right).$$

Because we are simulating a non-stationary system, we set $a^k = 0.01$ to track changes in the system. In general, the larger the value of a^k, the faster that v_i^k tracks v_i^*, but at the same time the larger the fluctuation of v_i^k around the value of v_i^*.

The system performs the above procedure for every time slot. Whenever the number of active users changes, the base station updates the time-fraction assignment requirements according to the fair sharing scheme, and the value of $\vec{v}^k$ at that time is used as the initial value of the on-line parameter estimation procedure in the new system state.

Figure 4 shows the results of our simulation experiment. In the figure, the x-axis represents the users' IDs. For each user, we compare the average performance in our opportunistic scheduling policy (the first bar) with that of the round-robin (RR) policy (the second bar). We can see that in every case, our opportunistic policy significantly outperforms the round-robbin policy, with gains of 20% to 150%. The amount of improvement varies from user to user because different users have different performance functions. The third (right-most) bar in the figure is the ratio of the total number of slots assigned to each user in our opportunistic scheme to that of the round-robin scheme, which was set equal to that required by the time-fraction assignment constraint. For all users, the third bar is virtually identical to 1. Hence, our scheduling scheme satisfies the time-fraction assignment constraint, which indicates that our stochastic approximation algorithm works well in the simulation experiment even in the nonstationary case.

Next, we show results for a simulation experiment designed to evaluate the impact of our on-line parameter estimation procedure and the sensitivity of our opportunistic scheme to estimation errors on parameters. We generate four time-correlated Gaussian processes, representing the performance-value sequences for four users. We display in Table II the means and standard deviations σ of the Gaussian processes for the four users. Each user has an exponentially distributed "on" and "off" periods. Because we know the distributions of the stochastic processes, we can calculate the exact value of $\vec{v}^*$ needed for the opportunistic scheduling policy. Hence, we can implement the exact (optimal) scheduling policy Q^*. In Table II, we show the average performance for each user obtained by applying policy Q^*, labeled as T_o.

Then we simulated our scheduling procedure where we use our on-line parameter-estimation algorithm to estimate $\vec{v}^*$ based on measurements. The average performance obtained is shown as T_e in Table II. As a benchmark comparison, we also simulated the average performance of the round-robin policy, shown as T_r in Table II.

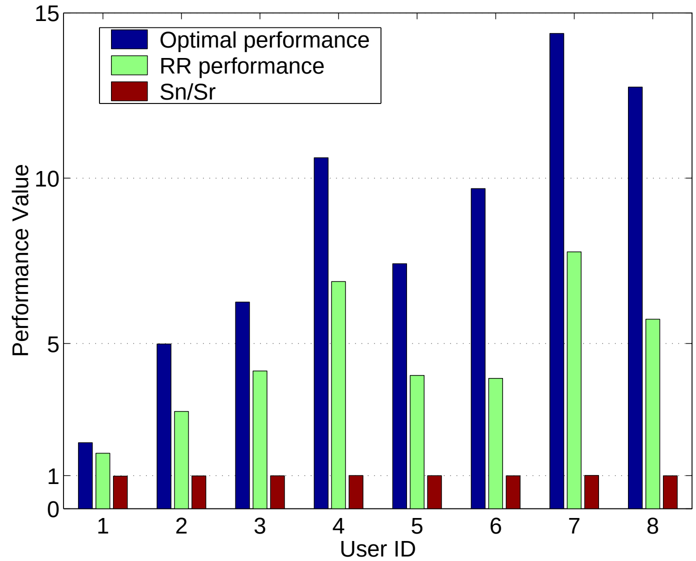

Fig. 4. Comparison of the opportunistic scheduling policy with the round-robin scheme. In the figure, S_n is the number of time slots assigned to user i in the optimal scheduling policy and S_r is the number of slots assigned to user i in the round-robin scheme.

ID	mean	σ	T_r	T_o	T_e	time ratio
1	10	10.8	10.001	20.03	19.90	1.001
2	10	6.9	9.998	18.12	18.10	0.999
3	10	4.0	10.000	14.75	14.68	1.001
4	10	2.5	10.001	12.44	12.40	0.998

TABLE II

COMPARISON OF THE AVERAGE PERFORMANCE OF THE SCHEDULING POLICY Q^* (WITH KNOWN $\vec{v}^*$), Q^k (WITH ESTIMATED $\vec{v}^*$), AND ROUND ROBIN.

We can see in Table II that the performance of both the optimal policy Q^* and our policy Q^k with estimated parameters are very similar, and are significantly higher than that of the round-robbin policy. This suggests that our on-line parameter estimation scheme works well and that any estimation errors present do not significantly degrade the performance of the policy relative to the optimal policy. Moreover, the performance gains appear to be related to the standard deviation: the higher the standard deviation, the larger the performance gain. Note that in the round-robbin scheme, the performance levels for all users are all approximately equal to the mean of the Gaussian processes (which is the mean performance value). This is to be expected because the round-robbin scheme allocates an equal fraction of time slots to each user, regardless of the channel conditions. Our opportunistic approach takes advantage of favorable transmission conditions, thereby leading to average performance values that are far above the mean of the Gaussian processes.

Table II also shows the ratio of the time-fractions obtained by our policy Q^k to that of the time-fractions for the round-robbin policy (which, as pointed out before, are equal to the prespecified values). As we can see, our scheme satisfies the time-fraction constraints very well.

V. Conclusion and Future Work

Cellular wireless networks present users with the promise of "anywhere, anytime" communication and enable them to enjoy the freedom of tetherless and ubiquitous services. Throughout the world, the demand for wireless communication systems has increased exponentially in the last few years. Moreover, people have begun to expect many advanced networking functions from wireless networks, such as multimedia applications, multicasting, and guaranteed quality of service. Hence, the technologies that have made cellular telephony successful are no longer sufficient to meet future demands.

In this paper, we present an opportunistic transmission-scheduling policy. Given a time-fraction assignment requirement, the scheduling policy maximizes the average network performance. In our model, each user's performance value is a stochastic process, reflecting the time-varying performance that results from randomly varying channel conditions. The users' performance-value processes can be arbitrarily correlated, both in time and across users. We establish the optimality of our opportunistic scheduling policy. We also provide a scheduling procedure that includes an on-line parameter-estimation algorithm to estimate parameter values used in the scheduling policy. Our scheduling algorithm has a low computational burden, which is important for on-line implementation. Via simulation, we illustrate the performance of our scheduling policy, showing significant performance gains over the round-robbin policy. Our simulation results also show that our scheme works well for the case of nonstationary performance-value sequences.

Now, we are working on a modified version of the scheduling scheme presented in this paper. Current scheduling scheme provides "long-term" fairness; i.e., the expectation of the number of time slots assigned to a user is guaranteed. However, with such a scheme, it is possible that a user could be starved for a long time (say, more than a few seconds), which may be undesirable for certain users. Usually, a user may also have the demand for "short-term" fairness; i.e., the user expects that the amount of service obtained within a time window be close to its fair share. The more stringent the "short-term" requirement, the less the chance to exploit channel "opportunistically". We combine the "opportunistic" scheduling with "short-term" fairness requirement. Basically, when a user has been starved, we increase the probability that the user being chosen to transmit in the future. When the user receives more resource than its share, we decrease the probability that that the user being chosen to transmit in the future.

Resource allocation and scheduling schemes are important in wireless networks, especially to provide high-rate data and seamless service for future wireless networks. There are many interesting problems in this area that remain to be resolved. These include the need for a general fairness criterion tailored to wireless networks; dealing with explicit delay requirement for certain users, and time-fraction assignment schemes that balances between system performance and fairness among users.

References

[1] H. Zhang, "Service disciplines for guaranteed performance service in packet-switching networks," *Proceedings of the IEEE*, vol. 83, no. 10, pp. 1374–1396, October 1995.

[2] S. Nanda, K. Balachandran, and S. Kumar, "Adaptation techniques in wireless packet data services," *IEEE Communications Magazine*, vol. 38, pp. 54–64, January 2000.

[3] S. Lu, V. Bharghavan, and R. Srikant, "Fair scheduling in wireless packet networks," *IEEE/ACM Transactions on Networking*, vol. 7, no. 4, pp. 473–489, August 1999.

[4] T. Nandagopal, S. Lu, and V. Bharghavan, "A unified architecture for the design and evaluation of wireless fair queueing algorithms," in *ACM Mobicom '99*, August 1999.

[5] T. S. E. Ng, I. Stoica, and H. Zhang, "Packet fair queueing algorithms for wireless networks with location-dependent errors," in *IEEE INFOCOM'98*, New York, 1998, vol. 3.

[6] P. Bender, P. Black, M. Grob, R. Padovani, N. Sindhushayana, and A. Viterbi, "CDMA/HDR: a bandwidth-efficient high-speed wireless data service for nomadic users," *IEEE Communications Magazine*, pp. 70–77, July 2000.

[7] A. Jalali, R. Padovani, and R. Pankaj, "Data throughput of CDMA-HDR a high efficiency-high data rate personal communication wireless system," in *IEEE VTC*, 2000.

[8] F. Kelly, "Charging and rate control for elastic traffic," *European Transactions on Telecommunications,*, vol. 8, pp. 33–37, 1997.

[9] G. Bianchi and A. T. Campbell, "A programmable MAC framework for utility-based adaptive quality of service support," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 2, pp. 244–255, February 2000.

[10] C. U. Saraydar, N. B. Mandayam, and D. J. Goodman, "Pareto efficiency of pricing-based power control in wireless data networks," in *1999 IEEE Wireless Communications and Networking Conference*, 1999, vol. 1.

[11] J. Zander, "Trends in resource management future wireless networks," in *Future Telecommunications Forum, FTF 99*, Beijing, China, December 1999.

[12] X. Liu, E. K. P. Chong, and N. B. Shroff, "Transmission scheduling for efficient wireless network utilization," 2000, http://shay.ecn.purdue.edu/∼xinliu/paper/.

[13] H. Kushner and G. Yin, *Stochastic Approximation Algorithms and Applications*, Springer-Verlag New York, 1997.

[14] I. J. Wang, E. K. P. Chong, and S. R. Kulkarni, "Weighted averaging and stochastic approximation," *Math. Control Signals Systems*, vol. 1, no. 10, pp. 41–60, 1997.

[15] G. Stuber, *Principles of Mobile Communication*, Kluwer Academic Publishers, 1996.

[16] M. Gudmundson, "Correlation model for shadow fading in mobile radio systems," *Electronics Letters*, vol. 27, pp. 2145–2146, November 1991.

Appendix

I. Proof of Optimality

Given the time-fraction assignments $r_1, \cdots, r_N$, the scheduling policy (2) is an "opportunistic" solution of (1). In the following, we prove the optimality of our scheduling policy.

To make the proof easy to understand and provide insight into the scheduling policy, we consider the special case where there are only two users and $P\{U_1 + v^* = U_2\} = 0$. For this special case, the opportunistic scheduling policy is given by $Q^*(\vec{U}) = \mathrm{argmax}(U_1 + v^*, U_2)$, which is illustrated in Figure 5. Above the line $U_1 + v^* = U_2$, we have $Q^*(\vec{U}) = 2$, while below the line, we have $Q^*(\vec{U}) = 1$. The probability measure of the line is 0 in this case. We show that $E\left(U_{Q^*(\vec{U})}\right) \geq E\left(U_{Q'(\vec{U})}\right)$ for any feasible policy Q' (recall that a feasible policy is a policy that satisfies the time-fraction assignment constraint).

Consider a feasible policy Q' that is different from policy Q^* in regions A and B as shown in Figure 5, where A, B, C, and D are events given by:

$$\begin{aligned} A &= \{Q^*(\vec{U}) = 2, Q'(\vec{U}) = 1\}; \\ C &= \{Q^*(\vec{U}) = Q'(\vec{U}) = 2\}; \\ B &= \{Q^*(\vec{U}) = 1, Q'(\vec{U}) = 2\}; \\ D &= \{Q^*(\vec{U}) = Q'(\vec{U}) = 1\}. \end{aligned}$$

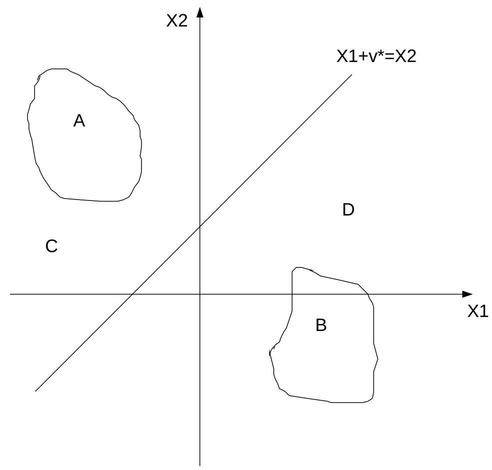

Fig. 5. Illustration of policies Q^* and Q'; $Q^*(\vec{U}) = 1$ in B and D, and $Q'(\vec{U}) = 1$ in A and D; $Q^*(\vec{U}) = 2$ in A and C, and $Q'(\vec{U}) = 1$ in B and C.

It is obvious that $P(A) = P(B)$. If $P(A) = 0$, then $Q'(\vec{U})$ is equal to $Q^*(\vec{U})$ with probability 1 and $E\left(U_{Q'(\vec{U})}\right) = E\left(U_{Q^*(\vec{U})}\right)$ because U_i is bounded. We next show that if $P(A) > 0$, then $E\left(U_{Q^*(\vec{U})}\right) > E\left(U_{Q'(\vec{U})}\right)$. Indeed,

$$\begin{aligned} & E\left(U_{Q^*(\vec{U})}\right) \\ &= \int_D U_1 \, \mathrm{d}P + \int_B U_1 \, \mathrm{d}P + \int_A U_2 \, \mathrm{d}P + \int_C U_2 \, \mathrm{d}P \\ & E\left(U_{Q'(\vec{U})}\right) \\ &= \int_D U_1 \, \mathrm{d}P + \int_B U_2 \, \mathrm{d}P + \int_A U_1 \, \mathrm{d}P + \int_C U_2 \, \mathrm{d}P. \end{aligned}$$

Hence,

$$\begin{aligned} & E\left(U_{Q^*(\vec{U})}\right) - E\left(U_{Q'(\vec{U})}\right) \\ &= \int_B (U_1 - U_2) \, \mathrm{d}P + \int_A (U_2 - U_1) \, \mathrm{d}P. \end{aligned}$$

On A, $U_2 > U_1 + v^*$, so $U_2 - U_1 > v^*$; and on B, $U_2 < U_1 + v^*$, so $U_1 - U_2 > -v^*$. So

$$\begin{aligned} & E\left(U_{Q^*(\vec{U})}\right) - E\left(U_{Q'(\vec{U})}\right) \\ &> \int_B -v^* \, \mathrm{d}P + \int_A v^* \, \mathrm{d}P \\ &= 0. \end{aligned}$$

Hence, $E\left(U_{Q^*(\vec{U})}\right) \geq E\left(U_{Q'(\vec{U})}\right)$ for any feasible policy Q'. The key insight in the above argument is that any policy Q' that schedules user 1 instead of user 2 in any region above the line will lose out to policy Q^* because the performance over that region is inferior for user 1 relative to user 2.

Next, we prove the scheduling policy is optimal *in general*; i.e., for the case of multiple users. Let $Q^*(\vec{U})$ be the opportunistic policy in Eq. (2). We show that $E\left(U_{Q^*(\vec{U})}\right) \geq E\left(U_{Q'(\vec{U})}\right)$ for any feasible policy Q'. A similar argument to the one used above applies here.

Let C_{ij} denote the event:

$$C_{ij} = \{Q^*(\vec{U}) = i, Q'(\vec{U}) = j\}$$

for $i = 1, \cdots, N$ and $j = 1, \cdots, N$. Because both Q^* and Q' are feasible policies, we have

$$\begin{aligned} B_j &= \bigcup_{i=1}^{N} C_{ij} = \{Q'(\vec{U}) = j\} \quad \Rightarrow \quad P(B_j) = r_j; \\ A_i &= \bigcup_{j=1}^{N} C_{ij} = \{Q^*(\vec{U}) = i\} \quad \Rightarrow \quad P(A_i) = r_i. \end{aligned}$$

Thus,

$$\begin{aligned} & E\left(U_{Q^*(\vec{U})}\right) - E\left(U_{Q'(\vec{U})}\right) \\ = & \sum_{i=1}^{N}\sum_{j=1}^{N}\int_{C_{ij}} U_{Q^*(\vec{U})} \, \mathrm{d}P - \sum_{i=1}^{N}\sum_{j=1}^{N}\int_{C_{ij}} U_{Q'(\vec{U})} \, \mathrm{d}P \\ = & \sum_{i=1}^{N}\sum_{j=1}^{N}\int_{C_{ij}} U_{Q^*(\vec{U})} - U_{Q'(\vec{U})} \, \mathrm{d}P \\ = & \sum_{i=1}^{N}\sum_{j=1}^{N}\int_{C_{ij}} U_i - U_j \, \mathrm{d}P. \end{aligned}$$

On C_{ij}, we have $Q^*(\vec{U}) = i$ and $Q'(\vec{U}) = j$. Because $Q^*(\vec{U}) = i$, then $U_i - v_i^* \geq U_j - v_j^*$ by the construction of the opportunistic policy Q^*, and so $U_i - U_j \geq v_i^* - v_j^*$. Therefore,

$$\begin{aligned} & E\left(U_{Q^*(\vec{U})}\right) - E\left(U_{Q'(\vec{U})}\right) \\ \geq & \sum_{i=1}^{N}\sum_{j=1}^{N}\int_{C_{ij}} v_i^* - v_j^* \, \mathrm{d}P \\ = & \sum_{i=1}^{N} v_i^* \sum_{j=1}^{N}\int_{C_{ij}} \mathrm{d}P - \sum_{j=1}^{N} v_j^* \sum_{i=1}^{N}\int_{C_{ij}} \mathrm{d}P \\ = & \sum_{i=1}^{N} v_i^* \int_{A_i} \mathrm{d}P - \sum_{j=1}^{N} v_j^* \int_{B_j} \mathrm{d}P \\ = & \sum_{i=1}^{N} v_i^* r_i - \sum_{j=1}^{N} v_j^* r_j \\ = & \ 0 \end{aligned}$$

Hence, $E\left(U_{Q^*(\vec{U})}\right) - E\left(U_{Q'(\vec{U})}\right) \geq 0$ for any feasible policy Q', which completes the proof.