Multiuser Scheduling in a Markov-modeled Downlink Environment.

Sugumar Murugesan, Philip Schniter and Ness B. Shroff

Abstract

We address the problem of multiuser scheduling in a cellular downlink system with partial channel information. In our setting, the channel of each user is modeled by a two-state Markov chain. The scheduler indirectly estimates the channel via accumulated Automatic Repeat Request (ARQ) feedback from the scheduled users and uses this information in future scheduling decisions. This problem is a special case of the *restless multi-armed bandit* processes that have been shown to be PSPACE-hard to solve in general. By modeling the scheduling problem as a Partially Observable Markov Decision Process (POMDP), we formulate a throughput maximization problem and show that, despite the visible complexity of this problem, a simple round-robin fashioned scheduling policy optimizes the system for the special case of three or less users in the system. We study the structure of this policy for an arbitrary number of users and establish a sufficient condition for the optimality of this policy. Drawing equivalence with a genie-aided system, we derive an explicit expression for the sum capacity of the downlink.

Index Terms–Markov channel, downlink, multiuser scheduling, greedy policy, sum capacity.

I. Introduction

Opportunistic multiuser scheduling, introduced by Knopp and Humblet in [1] and defined as *allocating the resources to the user experiencing the most favorable channel conditions* has gained immense popularity among network designers. Opportunistic scheduling essentially taps the multiuser diversity in the system and has motivated several researchers (e.g., [2]–[6]) to study the performance gains obtained by opportunistic scheduling under various scenarios. For a general treatment on the subject, see [7]. While i.i.d flat fading model is popularly used by researchers in modeling time varying channels, it fails to capture the memory in the channel observed in realistic scenarios. The Gilbert Elliott model [8] that represents the channel by a two state Markov chain addresses this issue. Specifically, a user experiences error-free transmission when it observes a "good" channel, and unsuccessful transmission in a "bad" channel. Several works have been done on opportunistic scheduling in this Markov modeled channel, e.g., [9]–[13]. It is understandable that the availability of the channel state information at the scheduler is crucial for the success of the opportunistic scheduling schemes. Traditionally, when the scheduler has no channel information, pilot based channel estimation is performed and the estimates are used for scheduling decisions (e.g., [2], [6], [14]). A new line of work, see for e.g., [15], [16], attempts to exploit Automatic Repeat reQuest (ARQ) feedback to estimate the state of the Markov modeled channels. ARQ is traditionally used for error control (e.g., [17]–[20]) at the data link layer. The memory inherent in the Markov channels opens up the opportunity to exploit this ARQ feedback information in estimating the channel states.

In this paper, we consider a downlink system with the channel of each user modeled by a two state Markov chain and demonstrate that ARQ feedback can be used to make informed multiuser scheduling decisions. Specifically, we consider a Markov-modeled downlink system with an ARQ feedback provision. Using a Partially Observable Markov Decision Process (POMDP) formulation,

This work was supported by National Science Foundation grants CCR-0237037, 0721434-CNS, 0635202-CCF, and Office of Naval Research grant N00014-07-1-0209.

The authors are with the Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43210, USA (e-mail: {murugess, schniter, shroff}@ece.osu.edu)

([21]–[24]), we show that, for $N \leq 3$ users, a simple greedy policy that maximizes the current reward is optimal in terms of the sum throughput. The greedy policy can be implemented via a simple round-robin based solution that does not require the statistics of the underlying Markov chain, so that it is easily amenable for practical implementation. Then, for the general N user case, by exploiting the round-robin structure of the greedy policy, we conjecture a sufficient condition for the optimality of the greedy policy. We provide extensive simulations that suggest that the greedy policy indeed satisfies this sufficient condition and is likely to be optimal for an arbitrary number of users in the system. By establishing an equivalence with a genie-aided system, we then derive a simple expression for the sum capacity of the Markov-modeled downlink system, for the two user case.

The paper is organized as follows. The problem setup is described in Section II and followed by the proof of the optimality of the greedy policy for $N = 2$ in Section III. Section IV discusses the round-robin structure of the greedy policy. The sufficient condition for the optimality of the greedy policy for the general case of N users is derived in Section V. In the same section, we prove that the greedy policy is optimal for $N = 3$ and make a conjecture about the $N > 3$ case. In Section VI, we derive the sum capacity of the Markov-modeled downlink. Conclusions are provided in Section VII.

II. Problem Setup

A. Channel Model and the Scheduling Problem

We consider downlink transmissions with N users. For each user, there is an associated queue at the base station that accumulates packets intended for that user. We assume an infinite backlog at each queue. The channel between the base station and each user is modeled by an i.i.d two-state Markov chain. We call this the ON-OFF channel with the ON state allowing the successful transmission of a fixed length packet. Time is slotted and the channel of each user remains fixed for a slot and evolves into another state in the next slot according to the Markov chain statistics. The time slots of all users are synchronized. The two-state Markov channel is characterized by a 2×2 probability transition matrix

$$P = \begin{bmatrix} p & q \\ r & s \end{bmatrix}, \tag{1}$$

where

- p=prob(channel is ON in the current slot | channel was ON in the previous slot)
- q=$1 - p$
- r=prob(channel is ON in the current slot | channel was OFF in the previous slot)
- $s = 1 - r$.

We assume $p \geq r$ throughout this work. Note that this implies, for any user, the channel state is positively correlated between adjacent slots.

The base station is the central controller that controls the transmission to the users in each slot. In any time slot, the base station does not know the *exact* channel state of the users and it must schedule the transmission of the head of line packet of exactly one user. Thus, a TDMA styled scheduling is performed here. The power spent in each transmission is fixed, and a traditional ARQ based transmission is deployed. Here, at the beginning of a time slot, the head of line packet of the scheduled user is transmitted. If the packet does not go through, i.e, cannot be decoded by the user (when the channel is in the OFF state), a NACK is sent back from the user at the end of the slot, and the packet is retained at the head of the queue. If the packet goes through (when the channel is in the ON state), an ACK is sent back and the packet is removed from the queue. Note that both ACKs and NACKs are assumed to be transmitted over a dedicated error-free channel. This ARQ information, along with the label of the control interval in which it is acquired, will be used in future scheduling decisions. The performance metric that the base station aims to maximize is the sum throughput of the system. Details will be discussed in the next section.

B. Formal Problem Definition

The base station must make scheduling decisions based on only a partial observation[1] of the underlying Markov chain. This fits our problem into the theory of partially observable Markov decision processes (POMDP) (see [21] for an overview of POMDP). We now proceed to introduce the terms/entities that we use in this work, many of which are borrowed from the POMDP literature.

Control interval k: Each time slot in our problem setup will henceforth be called a control interval. The "end" of the POMDP is fixed. A control interval is indexed by k if there are k intervals (including the interval in question) until the end of the process.

Action a_k: Indicates the index of the user scheduled in control interval k and hence takes on values from $1 \dots N$.

Belief vector at the k^{th} *control interval* π_k: The i^{th} element of π_k represents the probability that the channel of user $i \in 1 \dots N$ in the k^{th} control interval is in ON state, given all the past information about the channel. Let f_k denote the ARQ feedback at the end of control interval k with $f_k = 1$ indicating an ACK and $f_k = 0$ indicating a NACK. The belief vector evolves from control interval k to $k-1$[2], $\forall k > 1$, as follows:

$$\pi_{k-1}(i) = \begin{cases} p, & \text{if } i = a_k, f_k = 1 \\ r, & \text{if } i = a_k, f_k = 0 \\ p\pi_k(i) + r(1 - \pi_k(i)), & \text{if } i \neq a_k. \end{cases} \tag{2}$$

where the first case indicates that user i is scheduled in control interval k and an ACK feedback was received. Thus, according to the Markov chain statistics in (1), $\pi_{k-1}(i) = p$. The second case is explained similarly when a NACK feedback is received. The last case indicates that user i was not scheduled for transmission in control interval k and hence the base station must estimate the belief value at the current control interval ($\pi_{k-1}(i)$) from the belief value at the previous control interval ($\pi_k(i)$) and the Markov chain statistics in (1). It has been proven in [21] that the belief vector π_k is a sufficient statistic to the scheduling decisions and ARQ information from the past. Thus the scheduling decision in any control interval can be solely based on the belief vector for that interval and not on the past ARQ or schedule information.

Scheduling Policy $\mathfrak{A}_k$: A scheduling policy $\mathfrak{A}_k$ in the control interval k is a mapping from the belief vector and the control interval index to an action as follows:

$$\mathfrak{A}_k : (\pi_k, k) \rightarrow a_k \quad \forall k \geq 1, \pi_k \in [0,1]^N.$$

Note that the scheduling policy can, in general, be time-variant.

Reward Structure: In any control interval k, a reward of 1 is accrued when the transmission is successful, i.e, when $f_k = 1$, and no reward is accrued when $f_k = 0$. Note that this reward structure is defined to be consistent with our performance metric, the sum throughput (to be discussed shortly).

Net Expected Reward in the control interval m, V_m: With the belief vector, π_m, and the scheduling policy, $\{\mathfrak{A}_k\}_{k \leq m}$, fixed, the net expected reward, V_m, is the sum of the reward, $R_m(\pi_m, a_m)$, expected in the current control interval m and $\mathrm{E}[V_{m-1}]$, the reward expected in the future control intervals. Formally,

$$V_m(\pi_m, \{\mathfrak{A}_k\}_{k \leq m}) \;=\; R_m(\pi_m, a_m) + \mathrm{E}[V_{m-1}(\pi_{m-1}, \{\mathfrak{A}_k\}_{k \leq m-1}) | \pi_m, a_m], \tag{3}$$

where the expectation is over the belief vector π_{m-1}. Since the reward in each control interval is either 1 or 0, the expected current reward can be written as

$$R_m(\pi_m, a_m) \;=\; \pi_m(a_m).$$

[1] In this case, the set of time-stamped ARQ feedback on the channels.

[2] Note that the control intervals are in decreasing order consistent with the POMDP theory.

Performance Metric- the Sum Throughput, η_{sum}: For a given scheduling policy, $\{\mathfrak{A}_k\}_{k\geq 1}$, the sum throughput is given by

$$\eta_{\text{sum}}(\{\mathfrak{A}_k\}_{k\geq 1}) \;=\; \lim_{m\to\infty} \frac{V_m(\pi_{\text{ss}}, \{\mathfrak{A}_k\}_{k\geq 1})}{m}, \tag{4}$$

where $\pi_{\text{ss}}(i), i \in 1 \ldots N$ is the steady state probability that the channel of user i is ON in the underlying Markov chain.

Optimal Scheduling Policy, $\{\mathfrak{A}_k^*\}_{k\geq 1}$:

$$\{\mathfrak{A}_k^*\}_{k\geq 1} \;=\; \arg \max_{\{\mathfrak{A}_k\}_{k\geq 1}} \eta_{\text{sum}}(\{\mathfrak{A}_k\}_{k\geq 1}). \tag{5}$$

III. Optimal Scheduling Policy for Two Users

Consider the following policy:

$$\begin{aligned} \widehat{\mathfrak{A}}_k : (\pi_k, k) \to a_k \;&=\; \arg\max_{a_k} R_k(\pi_k, a_k) \\ &=\; \arg\max_i \pi_k(i) \quad \forall k \geq 1, \pi_k \in [0,1]^N. \end{aligned}$$

Since the above given policy attempts to maximize the expected current reward, without any regard to the expected future reward, it follows an approach that is fundamentally *greedy* in nature. For this reason, we henceforth call $\{\widehat{\mathfrak{A}}_k\}_{k\geq 1}$ the greedy policy.

Proposition 1: The sum throughput, η_{sum}, of the system is maximized by the greedy policy $\{\widehat{\mathfrak{A}}_k\}_{k\geq 1}$ for the case when $N = 2$, i.e.,

$$\mathfrak{A}_k^*|_{N=2} \;=\; \widehat{\mathfrak{A}}_k \quad \forall k \geq 1.$$

Proof: From subsection II-B, to prove the optimality of the greedy policy, it is sufficient to prove that

$$\{\widehat{\mathfrak{A}}_k\}_{k\leq m} \;=\; \arg \max_{\{\mathfrak{A}_k\}_{k\leq m}} V_m(\pi_m, \{\mathfrak{A}_k\}_{k\leq m}) \quad \forall m \geq 1, \pi_m \in [0,1]^2. \tag{6}$$

We first prove the following statement:

(P) If, for a fixed $m > 1$,

$$\{\widehat{\mathfrak{A}}_k\}_{k\leq m-1} \;=\; \arg \max_{\{\mathfrak{A}_k\}_{k\leq m-1}} V_{m-1}(\pi_{m-1}, \{\mathfrak{A}_k\}_{k\leq m-1}) \quad \forall \pi_{m-1} \in [0,1]^2,$$

then

$$\{\widehat{\mathfrak{A}}_k\}_{k\leq m} \;=\; \arg \max_{\{\mathfrak{A}_k\}_{k\leq m}} V_m(\pi_m, \{\mathfrak{A}_k\}_{k\leq m}) \quad \forall \pi_m \in [0,1]^2.$$

The proof of (P) involves expanding the net expected reward V_m as a sum of the rewards expected in each of the future control intervals with $\{\mathfrak{A}_k\}_{k\leq m-1} = \{\widehat{\mathfrak{A}}_k\}_{k\leq m-1}$. We then establish that the reward expected to be accrued in any future control interval is independent of the current scheduling decision a_m (this result is summarized in Corollary 2). This proves (P) which in turn is used to establish (6) using an induction argument. The complete proof is omitted for conciseness and the reader is referred to [28] for details. ■

Corollary 2: The future reward expected to be accrued in any control interval $k \leq m-1$ is independent of the scheduling decision a_m, as long as the greedy policy is implemented in control interval k. Formally,

$$\mathrm{E}_{\pi_k|\pi_m, a_m=1} R_k(\pi_k, \hat{a}_k) \;=\; \mathrm{E}_{\pi_k|\pi_m, a_m=2} R_k(\pi_k, \hat{a}_k), \forall k \leq m-1$$

with $\hat{a}_k$ indicating the use of greedy policy in control interval k.

The above observation will be pivotal in obtaining a simple, closed form expression for the sum capacity of the Markov-modeled downlink in Section VI.

It has to be mentioned that a parallel work, [25], by Qing Zhao et al., addresses a similar problem in a cognitive radio setting where a single user attempts to opportunistically access one of the several radio channels. Due to the fundamental difference in the application areas targeted, the overlap between our paper and [25] is limited to the result on the optimality of the greedy policy when $N = 2$ users. The proof technique they have used involves evaluating the net expected reward by averaging over the channel states of both the users in the current control interval. Whereas, in our case, we averaged over the belief vector. Moreover, we explicitly established that the reward expected to be accrued in *any* future control interval is independent of the current scheduling decision. This is unlike [25], where the independence result is obtained only for the net future reward.

IV. Structure of the Greedy Policy for N Users

We now take a closer look at the structure of the greedy policy for the general case of N users. We begin by defining the following quantity.

Scheduling order vector, O_k: The ordered arrangement of the index of the users in decreasing order of $\pi_k(i)$, i.e.,

$$\begin{aligned} O_k(1) &= \arg\max_i \pi_k(i) \\ &\vdots \\ O_k(N) &= \arg\min_i \pi_k(i). \end{aligned}$$

Thus, under the greedy policy in k, $a_k = O_k(1)$.

We now discuss the evolution of O_k to O_{k-1}. Consider any two users $i \neq a_k$ and $j \neq a_k$. Thus from (2), $\pi_{k-1}(i) = p\pi_k(i) + r(1 - \pi_k(i)) = (p - r)\pi_k(i) + r$. Similarly $\pi_{k-1}(j) = (p - r)\pi_k(j) + r$. Thus $\pi_{k-1}(i) \geq \pi_{k-1}(j)$ if $\pi_k(i) \geq \pi_k(j), \forall i \neq a_k, j \neq a_k$. Consider the user scheduled in control interval k, i.e., user a_k. If the ARQ feedback $f_k = 1$, then $\pi_{k-1}(a_k) = p$. Since for any user $i \neq a_k$, $\pi_{k-1}(i) = p\pi_k(i) + r(1 - \pi_k(i))$ and since $p \geq r$, we have, $\pi_{k-1}(a_k) \geq \pi_{k-1}(i), \forall i \neq a_k$. Similarly when $f_k = 0$, $\pi_{k-1}(a_k) = r \leq \pi_{k-1}(i), \forall i \neq a_k$. From the preceding observations,

$$O_{k-1} = \begin{cases} [a_k \ \{O_k - a_k\}], & \text{if } f_k = 1 \\ [\{O_k - a_k\} \ a_k], & \text{if } f_k = 0, \end{cases} \tag{7}$$

where $\{O_k - a_k\}$ is the schedule order vector O_k with the element valued a_k removed. For instance $\{[x \ y \ z] - y\}$=$[x \ z]$.

As a special case, when the greedy policy is employed in control interval k, i.e., when $a_k = O_k(1)$,

$$O_{k-1} = \begin{cases} O_k, & \text{if } f_k = 1 \\ [O_k(2) \ O_k(3) \ \ldots O_k(N) \ O_k(1)], & \text{if } f_k = 0. \end{cases} \tag{8}$$

We are now in a position to make the following important observation:

Let the greedy policy be implemented from control interval m. Let the schedule order vector, O_m, be available to the base station. The scheduling algorithm is implemented as follows: *Schedule the user positioned at the top of the schedule order vector (i.e., $a_m = O_m(1)$). If an ACK is received, schedule the same user again in the next control interval. Otherwise, schedule the next user in the schedule order vector O_m. Repeat the same procedure in all the future control intervals. If the bottom of the schedule*

order vector is reached, repeat from the top. Formally, the algorithm is implemented in the following simple steps

- Step 1: Initialize the control interval index $k \leftarrow m$ and the position of the scheduled user in the schedule order vector as $i \leftarrow 1$.
- Step 2: Schedule user $O_m(i)$ in control interval k, i.e., $a_k = O_m(i)$.
- Step 3: If $f_k = 0$ and $i < N$, then $i \leftarrow i + 1$. If $f_k = 0$ and $i = N$, then $i \leftarrow 1$.
- Step 4: $k \leftarrow k - 1$. If $k > 0$, then repeat steps 2-4.

Thus the scheduling algorithm, under the greedy policy, boils down to a simple round-robin algorithm with a change in scheduling decision stimulated by a NACK feedback. The schedule order vector, O_m, provides the order of this round-robin approach. There is no need to evaluate the belief vector in every control interval and hence the Markov transition matrix information is not required. This structure makes the greedy policy particularly attractive from an implementation point of view. Motivated by this development, we proceed to examine the optimality of the greedy policy in a general N user setting.

V. On the Optimality of the Greedy Policy for N Users

A. Sufficient Condition for the Optimality of the Greedy Policy

Consider a control interval $m > 1$ with belief vector π_m and action a_m. Let the users be indexed in the order of their belief values in control interval m, i.e, $O_m = [1 \dots N]$. Assuming $\{\mathfrak{A}_k\}_{k \le m-1} = \{\widehat{\mathfrak{A}}_k\}_{k \le m-1}$ and recalling the definition of state vector S_k from Section III, we rewrite the net expected reward from (3) as follows

$$V_m(\pi_m, \{a_m, \{\widehat{\mathfrak{A}}_k\}_{k \le m-1}\}) \;\; = \;\; \pi_m(a_m) + \sum_{S_m} P_{S_m|\pi_m}(S_m|\pi_m) \hat{V}_{m-1}(S_m, O_{m-1}),$$

where $\hat{V}_{m-1}$ is the expected future reward conditioned on the state vector in control interval m. The *hat* on this quantity emphasizes the use of the greedy policy in all $k \le m-1$. $P_{S_m|\pi_m}(S_m|\pi_m)$ is the conditional probability of the current state vector S_m given the belief vector π_m. Note that the schedule order vector O_{m-1} is only a function of O_m and the state $S_m(a_m)$, thus maintaining consistency with the amount of information available for scheduling decision in the actual problem setup. We now proceed to compare the net expected reward when $a_m = n$ and $a_m = n+1$ where $n \in \{1 \dots N-1\}$. Let Y and X be random binary vectors of lengths $n-1$ and $N-n-1$ (empty when the length is non-positive) respectively. Then,

$$\begin{aligned}
&V_m(\pi_m, \{a_m = n, \{\widehat{\mathfrak{A}}_k\}_{k \le m-1}\}) \\
&= \;\; \pi_m(n) + \sum_{Y,X} P_{S_m|\pi_m}([Y\ 0\ 0\ X]|\pi_m) \hat{V}_{m-1}\big([Y\ 0\ 0\ X], [\{O_m - n\}\ n]\big) \\
&\quad + \sum_{Y,X} P_{S_m|\pi_m}([Y\ 0\ 1\ X]|\pi_m) \hat{V}_{m-1}\big([Y\ 0\ 1\ X], [\{O_m - n\}\ n]\big) \\
&\quad + \sum_{Y,X} P_{S_m|\pi_m}([Y\ 1\ 0\ X]|\pi_m) \hat{V}_{m-1}\big([Y\ 1\ 0\ X], [n\ \{O_m - n\}]\big) \\
&\quad + \sum_{Y,X} P_{S_m|\pi_m}([Y\ 1\ 1\ X]|\pi_m) \hat{V}_{m-1}\big([Y\ 1\ 1\ X], [n\ \{O_m - n\}]\big),
\end{aligned} \tag{9}$$

where $O_m \to O_{m-1}$ evolution follows (7). Since the Markov channel statistics are identical across the users, we have the following symmetry property: for any $k \ge 1$,

$$\hat{V}_k(S_{k+1}, O_k) \;\; = \;\; \hat{V}_k(\tilde{S}_{k+1}, \tilde{O}_k) \;\; \text{if} \;\; S_{k+1}(O_k(i)) = \tilde{S}_{k+1}(\tilde{O}_k(i)) \;\; \forall\, i \in \{1 \dots N\}. \tag{10}$$

Expanding $V_m(\pi_m, \{a_m = n+1, \{\widehat{\mathfrak{A}}_k\}_{k \le m-1}\})$ along the lines of (9), and using the preceding symmetry property, with further mathematical simplification ([28]), we can evaluate the difference in the net

expected reward as follows,

$$\begin{aligned} & V_m(\pi_m, \{a_m = n, \widehat{\mathfrak{A}}_{k,k\leq m-1}\}) - V_m(\pi_m, \{a_m = n+1, \widehat{\mathfrak{A}}_{k,k\leq m-1}\}) \\ = \; & \Big(\pi_m(n) - \pi_m(n+1)\Big) \times \\ & \Big(1 - \sum_{Y,X} \Big[\hat{V}_{m-1}\big([Y\ 1\ X\ 0], [1 \ldots N]\big) - \hat{V}_{m-1}\big([1\ Y\ 0\ X], [1 \ldots N]\big)\Big] \times \\ & P_{S_m|\pi_m}\big([S_m(1) \ldots S_m(n-1)] = Y|\pi_m\big) P_{S_m|\pi_m}\big([S_m(n+2) \ldots S_m(N)] = X|\pi_m\big)\Big). \end{aligned} \quad (11)$$

Proposition 3: A sufficient condition for the optimality of the greedy policy is given as follows,

$$\Big[\hat{V}_{m-1}\big([Y\ 1\ X\ 0], [1 \ldots N]\big) - \hat{V}_{m-1}\big([1\ Y\ 0\ X], [1 \ldots N]\big)\Big] \leq 1, \quad (12)$$

$\forall\ m > 1$, $n \in \{1 \ldots N-1\}$, Y, X being random binary vectors of length $n-1$ and $N-n-1$ and $\{\mathfrak{A}_k\}_{k\leq m-1} = \{\widehat{\mathfrak{A}}_k\}_{k\leq m-1}$.

Proof: The proof uses (11) along with an induction argument. The reader is referred to [28] for details. ■

B. Optimality of the Greedy Policy with $N = 3$ Users

Proposition 4: When $N = 3$ users, the greedy policy satisfies the sufficient condition in Proposition 3 and hence is optimal.

Proof: With $N = 3$, $n \in \{1, 2\}$. The sufficient condition in Proposition 3 becomes, $\forall\ m > 1$,

$$\begin{aligned} \Big[\hat{V}_{m-1}\big([1\ X\ 0], [1\ 2\ 3]\big) - \hat{V}_{m-1}\big([1\ 0\ X], [1\ 2\ 3]\big)\Big] \leq 1 \text{ when } n = 1 \\ \Big[\hat{V}_{m-1}\big([Y\ 1\ 0], [1\ 2\ 3]\big) - \hat{V}_{m-1}\big([1\ Y\ 0], [1\ 2\ 3]\big)\Big] \leq 1 \text{ when } n = 2 \end{aligned} \quad (13)$$

where X, Y are binary numbers and $\{\mathfrak{A}_k\}_{k\leq m-1} = \{\widehat{\mathfrak{A}}_k\}_{k\leq m-1}$. The proof of (13) is available in [28]. ■

Due to the complex relationship between the scheduling decision in a control interval and the reward expected in the future intervals, an analysis of the optimality of the greedy policy for the general N-user case appears difficult. But with support from simulation results, we conjecture that the sufficient condition in Proposition 3 is satisfied for any value of N, thus suggesting the optimality of the greedy policy for any N. Fig. 1 plots the values of $\hat{V}_{m-1}\big([Y\ 1\ X\ 0], [1 \ldots N]\big) - \hat{V}_{m-1}\big([1\ Y\ 0\ X], [1 \ldots N]\big)$ for various values of $n \in \{1 \ldots N-1\}$ when $N = 5$, $P = \begin{bmatrix} 0.6324 & 0.3676 \\ .0975 & .9025 \end{bmatrix}$. In each of the four subplots, Y and X are allowed to take on every possible binary word of length $n-1$ and $N-n-1$, respectively.

VI. Sum Capacity of the Markov-modeled Downlink

With the greedy policy established as the sum-throughput maximizing scheduling policy for $N = 2$ users, the sum capacity of the system is given by the sum throughput under the greedy policy. We now give the following result.

Proposition 5: When $N = 2$, the sum capacity of the given Markov-modeled downlink equals that of a genie-aided Markov-modeled downlink where, at the end of every control interval, the base station learns the state of the channels in that control interval. The sum capacity is given as

$$C_{sum} = p_s p + (1 - p_s) p_s \text{ with } \; p_s = \frac{r}{1 - (p - r)},$$

where p_s is the probability that the channel of user 1 (or 2) is ON in steady state. The greedy policy achieves the sum capacity of both systems.

Proof: We begin with the sum throughput of the greedy policy in the genie-aided system. With $\pi_{\rm ss}(1) = \pi_{\rm ss}(2) = p_s$ (the steady state ON probability of the Markov channels), it can be proved ([28]) that,

$$\eta_{\rm sum}^{\rm genie}(\{\widehat{\mathfrak{A}}_k\}_{k\geq 1}) \;\; = \;\; p_s p + (1-p_s)p_s, \tag{14}$$

with $p_s = \frac{r}{1-(p-r)}$. We now proceed to prove that the sum throughput of the greedy policy in the original system equals that of the greedy policy in the genie-aided system. Consider the scheduling problem for the original system in control interval k under the greedy policy. When the user scheduled in the previous control interval a_{k+1} sends back an ACK, the scheduling decision is retained in the current interval, i.e., $a_k = a_{k+1}$. Otherwise, the other user is scheduled in k. This procedure is evident from the structure of the greedy policy discussed in Section IV. We can interpret this decision logic as follows:

When at least one of the users had an ON channel in the previous control interval, that user[3] *is identified for scheduling in the current control interval k, leading to an expected current reward $R_k = p$. Reward $R_k = r$ is accrued only when both the channels were in OFF state.*

From this observation we see that, under the greedy policy, no improvement in sum throughput can be achieved even if the channel states of both the users in control interval $k+1$ were available for the scheduling decision in control interval k. This establishes the equivalence between the original system and the genie-aided system in terms of the sum throughput achieved by the greedy policy. We have already proved the sum throughput optimality of the greedy policy in the original system when $N = 2$, in Section III. Thus the sum capacity of the original system is given by (14).

We now proceed to prove that (14) is the sum capacity of the genie-aided system as well by examining the sum throughput optimality of the greedy policy in the genie-aided system. For any control interval m, we rewrite the net expected reward from 3 for the genie aided system below.

$$V_m^{\rm genie}(\pi_m, \{\mathfrak{A}_k\}_{k\leq m}) \;\; = \;\; R_m(\pi_m, a_m) + \mathrm{E}[V_{m-1}^{\rm genie}(\pi_{m-1}, \{\mathfrak{A}_k\}_{k\leq m-1})|\pi_m, a_m].$$

Note that since the current channel state of both the users ($S_m(1)$ and $S_m(2)$) are available at the base station at the end of the control interval m, the belief vector π_{m-1} and hence the expected future reward $\mathrm{E}[V_{m-1}^{\rm genie}]$ are independent of the scheduling decision a_m. Therefore, in any control interval, the net expected reward is maximized by the greedy policy. This establishes the sum throughput optimality of the greedy policy in the genie-aided system as well. The proposition thus follows. ■

Insights on the result in Proposition 5 can be obtained by examining the fundamental trade-off involved in the scheduling decisions in the Markov-modeled downlink. Transmission to a scheduled user and eventually obtaining ARQ feedback from that user accomplishes the following two objectives:

- data transmission in the current slot, which influences the current reward R_k.
- probing the channel of a user for future scheduling decisions, which influences the expected reward in future control intervals.

The optimal schedule strikes a balance between these two objectives (that need not always contradict each other).

From the discussion in the proof of Proposition 5, we see that, in the original system, the choice of the user whose channel is probed becomes irrelevant as far as the future reward is concerned[4]. That explains the result in Corollary 2 and why the greedy policy is optimal in the sum throughput sense. Considering the genie-aided system, since the channel state information of both the users are freely

[3]User a_{k+1} is given higher priority if both channels were ON.

[4]As long as one of the users is probed.

available to the scheduler at the end of the control interval, there is no need to probe the channel of any user to help with future scheduling decisions. This makes the greedy policy the sum throughput optimal for the genie-aided system as well.

VII. Conclusion

We have addressed the problem of scheduling using ARQ feedback in a Markov-modeled downlink. Using POMDP formulation, we have shown that, for $N \leq 3$ users, a simple greedy policy that maximizes the current reward is optimal in terms of sum throughput. We have shown that the greedy policy can be implemented by a simple round-robin based algorithm does not require the statistics of the underlying Markov chain, thus establishing the attractiveness of the greedy policy from a practical point of view. We have also derived a sufficient condition for the optimality of the greedy policy in the general N user case. We conjectured that the greedy policy satisfies this condition and hence is optimal for any number of users in the system. By establishing an equivalence with a genie-aided system, a simple expression for the sum capacity of the Markov-modeled downlink system has been derived when $N = 2$. Before we conclude, note that our problem is a special case of the *restless multi-armed bandit* problem [26]. This problem has been shown to be PSPACE-hard to solve in general [27]. Thus our result on the optimality of the greedy policy may be of importance in understanding the properties of the optimal policy in the general restless multi-armed bandit processes.

References

[1] R. Knopp and P. A. Humblet, "Information capacity and power control in single cell multiuser communications," *Proc. IEEE International Conference on Communications,* (Seattle, WA), pp. 331-335, June 1995.

[2] P. Viswanath, D. Tse, and R. Laroia, "Opportunistic beamforming using dumb antennas," *IEEE Transactions on Information Theory,* vol. 48, no. 6, pp. 1277-1294, Jun. 2002.

[3] R. W. Heath, M. Airy, and A. J. Paulraj, "Multiuser diversity for MIMO wireless systems with linear receivers," *Proc. Asilomar Conf. Signals, Systems, and Computers,* (Pacific Grove, CA), pp. 1194-1199, Nov. 2001.

[4] A. Gyasi-Agyei, "Multiuser diversity based opportunistic scheduling for wireless data networks," *IEEE Communications Letters,* vol. 9, issue 7, pp. 670-672, Jul. 2005.

[5] J. Chung, C. S. Hwang, K. Kim, and Y. K. Kim, "A random beamforming technique in MIMO systems exploiting multiuser diversity," *IEEE Journal on Selected Areas in Communications,* vol. 21, pp. 848-855, Jun. 2003.

[6] S. Murugesan, E. Uysal-Biyikoglu and P. Schniter, "Optimization of Training and Scheduling in the Non-Coherent MIMO Multiple-Access Channel," *IEEE Journal on Selected Areas in Communications,* vol. 25, no. 7, pp. 1446-1456, Sep. 2007.

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

[8] E. Gilbert, "Capacity of a burst-noise channel," *Bell Systems Technical Journal,* vol. 39, pp. 1253-1266, 1960.

[9] 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, Aug. 1999.

[10] T. Nandagopal, S. Lu, and V. Bharghavan, "A unified architecture for the design and evaluation of wireless fair queueing algorithms," *Proc. ACM Mobicom,* Aug. 1999.

[11] T. Ng, I. Stoica, and H. Zhang, "Packet fair queueing algorithms for wireless networks with location-dependent errors," *Proc IEEE INFOCOM,* (New York), vol. 3, 1998.

[12] S. Shakkottai and R. Srikant, "Scheduling real-time traffic with deadlines over a wireless channel," *Proc. ACM Workshop on Wireless and Mobile Multimedia,* (Seattle, WA), Aug. 1999.

[13] Y. Cao and V. Li, "Scheduling algorithms in broadband wireless networks," *Proc. IEEE,* vol. 89, no. 1, pp. 76-87, Jan. 2001.

[14] Y. C. Liang and R. Zhang, "Multiuser MIMO Systems with Random Transmit Beamforming," *International Journal of Wireless Information Networks,* vol. 12, no.4, pp. 235-247, Dec. 2005.

[15] M. Zorzi and R. Rao, "Error control and energy consumption in communications for nomadic computing," *IEEE Transactions on Computers,* vol. 46, pp. 279-289, Mar. 1997.

[16] L. A. Johnston and V. Krishnamurthy, "Opportunistic File Transfer over a Fading Channel: A POMDP Search Theory Formulation with Optimal Threshold Policies," *IEEE Transactions on Wireless Communications,* vol. 5, no. 2, Feb. 2006.

[17] S. Lin, D. Costello, and M. Miller, "Automatic-repeat-request error control schemes," *IEEE Communications Magazine,* vol. 22, pp. 5-17, Dec. 1984.

[18] D. L. Lu and J. F. Chang, "Performance of ARQ protocols in nonindependent channel errors," *IEEE Transactions on Communications,* vol. 41, pp. 721-730, May 1993.

[19] M. Zorzi, R. R. Rao, and L. B. Milstein, "ARQ error control on fading mobile radio channels," *IEEE Transactions on Vehicular Technology,* vol. 46, pp. 445-455, May 1997.

[20] Y. J. Cho and C. K. Un, "Performance analysis of ARQ error controls under Markovian block error pattern," *IEEE Transactions on Communications,* vol. 42, pp. 2051-2061, Feb.-Apr. 1994.

[21] R. D. Smallwood and E. J. Sondik, "The Optimal Control of Partially Observable Markov Processes Over a Finite Horizon," *Operations Research,* Sep. 1973.

[22] S. Christian Albright, "Structural Results for Partially Observable Markov Decision Processes," *Operations Research,* vol. 27, no. 5, pp. 1041-1053, Sep.-Oct. 1979.

[23] C. C. White and W. Scherer, "Solution procedures for partially observed Markov decision processes," *Operations Research,* pp. 791-797, 1985.

[24] G. E. Monahan, "A survey of partially observable Markov decision processes: Theory, Models, and Algorithms," *Management Science,* vol. 28, no. 1, pp. 1-16, Jan. 1982.

[25] Q. Zhao, B. Krishnamachari, and K. Liu, "On Myopic Sensing for Multi-Channel Opportunistic Access," submitted to *IEEE Transactions on Wireless Communications*, November, 2007. (`http://www.ece.ucdavis.edu/~qzhao/RecentPublication.html`).

[26] P. Whittle, "Restless Bandits: Activity Allocation in a Changing World," *Journal of Applied Probability,* vol. 25, pp. 287-298, 1988.

[27] C. H. Papadimitriou, J. N. Tsitsiklis, "The complexity of optimal queueing network control," *Mathematics of Operations Research,* vol. 24, no. 2, pp. 293305, May 1999.

[28] S. Murugesan, P. Schniter and N. B. Shroff, "Markov Modeled Downlink: Opportunistic Multiuser Scheduling and Stability Region," *Technical Report, Dept. of ECE, The Ohio State University,* Spring, 2008. (`http://www.ece.osu.edu/~schniter/pubs_by_topic.html`).

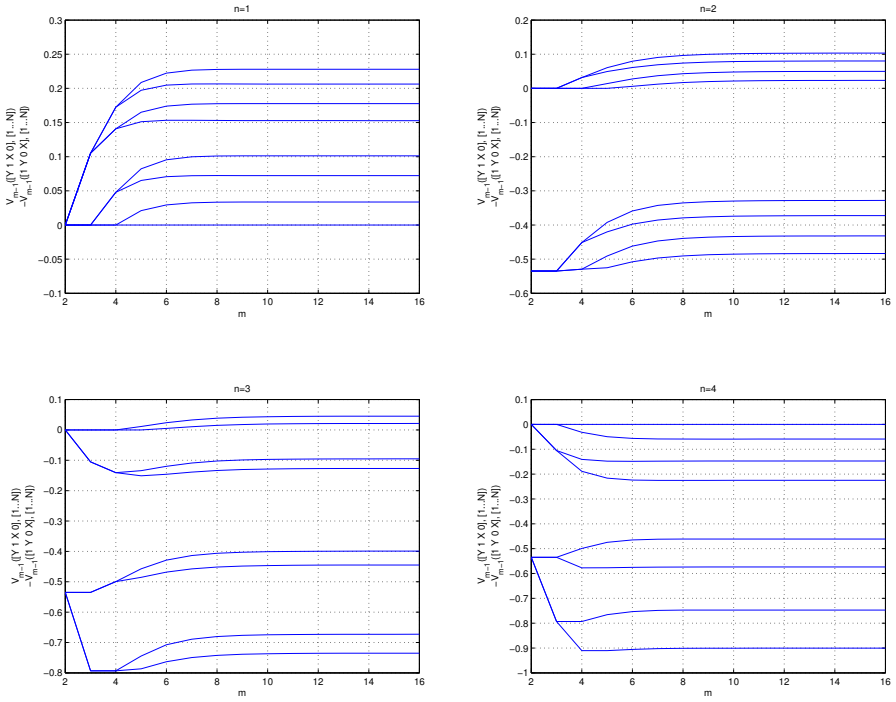

Fig. 1. Illustration showing $\hat{V}_{m-1}\big([Y\ 1\ X\ 0],[1\dots N]\big) - \hat{V}_{m-1}\big([1\ Y\ 0\ X],[1\dots N]\big) \leq 1\ \forall n \in \{1\dots N-1\}$ with $N = 5$ users.