Opportunistic Scheduling Using Channel Memory in Markov-modeled Wireless Networks

Dissertation

Presented in Partial Fulfillment of the Requirements for

the Degree Doctor of Philosophy in the

Graduate School of The Ohio State University

By

Sugumar Murugesan, B.E., M.S.

Graduate Program in Electrical and Computer Engineering

The Ohio State University

2010

Dissertation Committee:

Philip Schniter, Advisor

Ness B. Shroff, Co-advisor

Emre Koksal

ABSTRACT

The presence of multiple users in a network provides us with a valuable resource known as multiuser diversity. With information on the instantaneous states of the channels, multiuser diversity can be tapped by opportunistic multiuser scheduling. It is important that the channel state information is acquired in a cost-effective way so that the losses involved in this operation do not offset the gains promised by opportunistic scheduling. For various network environments of practical interest, this dissertation models the radio frequency links with memory, and studies the modalities to exploit the channel memory to simultaneously estimate channel state information, while performing opportunistic multiuser scheduling. The data transmission at any point of time is shown to be associated with two potentially contradicting objectives: opportunistic scheduling for immediate gains and channel exploration for future gains. Thus the joint scheduling problem is a dynamic program, specifically a partially observable Markov decision process that is traditionally known to be intractable or computationally expensive to implement. For various networks, we study these processes in an optimality framework and whenever possible, derive the optimal scheduling policy in closed form. In other cases, strongly founded on the optimality framework, we derive computationally inexpensive scheduling policies with near-optimal numerical performances.

By appropriately exploiting the memory in the fading channels, significant system level gains can be achieved using opportunistic scheduling, even with minimal feedback, and a considerable portion of these gains can be realized even with sub-optimal policies that are computationally inexpensive to implement – This is the central message of this dissertation.

Dedicated to my dear parents Mr. K. R. Murugesan and Mrs. Saroja Murugesan.

ACKNOWLEDGMENTS

First and foremost, my heartfelt thanks to my advisor, Prof. Philip Schniter. Phil (as I fondly call him) was a great source of inspiration and support through my good and hard times at OSU. He taught me the essence of research and the significance of the word 'impact' in research. I appreciate Phil for giving me considerable freedom in formulating and pursuing research problems of my interest, which, in turn, provided me with valuable well-rounded research experience. Thank you, Phil, for everything.

I sincerely thank my co-advisor Prof. Ness B. Shroff. His constant encouragement and thought-provoking feedback are instrumental in the creation of this dissertation. Prof. Shroff's innate ability to simplify even the most complex systems and look at the larger picture is something I hope to successfully emulate someday. I am going to miss the parties that he regularly hosts at his residence, for his students. Thank you, Prof. Shroff, for your kindness and support.

IPS lab, my home away from home, needs a special mention here. Through my Masters and PhD days at IPS, I have had the pleasure to work alongside many awesome IPS-ters — so many that I could not mention them all here. Special mention goes to Arun Sr., Praveen, Sib, Lifeng, Som, Amrita, Naveen, Justin, Arun Jr., Ahmed, Rahul, Ozan, Srikanth and Wenzhuo. These folks made my experience at IPS a memorable one.

I thank Jeri, the SITE staff and all other administrative staff in the department for their support.

This dissertation is based on work supported by the NSF CAREER grant 237037, the Office of Naval Research grant N00014-07-1-0209, NSF grants CNS-0721236, CNS-0626703, ARO W911NF-08-1-0238 and ARO W911NF-07-10376.

VITA

June 4, 1983	Born - Sivagangai, India
2004	B.E. Electronics and Communication Engineering, College of Engineering - Guindy, Anna University
2006	M.S. Electrical and Computer Engineering, The Ohio State University
2007-present	Graduate Research Associate, The Ohio State University.

FIELDS OF STUDY

Major Field: Electrical Engineering

Studies in:

Communication and Signal Processing	Prof. Philip Schniter
Communication Networks	Prof. Ness B. Shroff
Information Theory	Prof. Hesham El Gamal
Queueing Theory	Prof. Elif Uysal-Biyikoglu
Random Signal Analysis	Prof. Randolph Moses

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Recent years have witnessed a large-scale deployment of wireless networks, thanks to a surge in the demand for wireless "anytime, anywhere", high data rate services such as the wireless broadband access, multimedia services (MMS), video chat, mobile HDTV, teleconferencing, gaming and so on. Driven by this demand, spectrally efficient communication techniques have been progressively built into generations of wireless networks. These techniques are typically characterized by intelligent design paradigms across the OSI layers, such as rate adaptation, incremental redundancy ARQ, turbo coding, multiple input multiple output (MIMO) smart antennas, opportunistic multiuser scheduling, to name a few.

Among these, opportunistic multiuser scheduling aims to improve the network utilization by taking advantage of the fluctuations in the wireless channel across users, across time. Fluctuations in wireless channels are induced by a well-known phenomenon called multipath fading – when a transmitted signal traverses multiple paths (occurs when the signal reflects from obstructions like vehicles or buildings) and hence multiple copies reach the destination, they interfere constructively or destructively leading to fluctuations in channel strength. This is illustrated in Fig. 1.1. Traditionally considered a disadvantage, fading has been shown by Knopp and Hum-

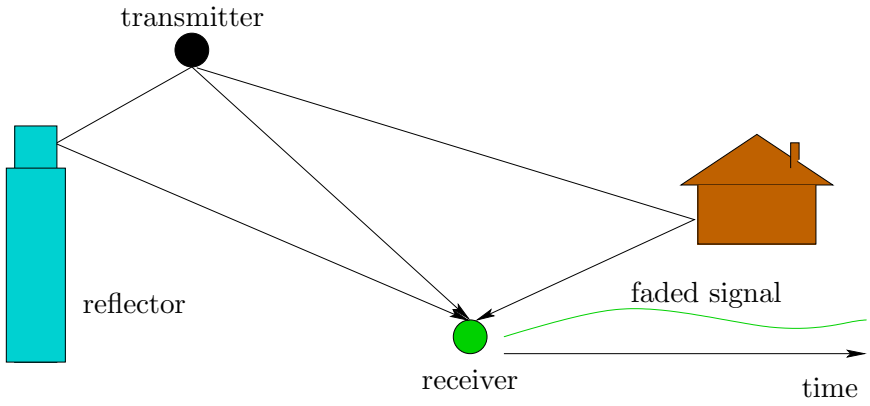

Figure 1.1: Illustration of the multipath fading phenomenon.

blet [1] to be beneficial in a wireless network, if appropriately exploited. They showed that, by scheduling transmission to the network user experiencing the best channel strength at the moment, significant system level gains can be realized. Thus fading essentially gives an opportunity for the network to ride on the peak channel condition at all times, as illustrated in Fig. 1.2. The resource that is tapped here is called *multiuser diversity* and the intelligent resource allocation is commonly referred to as *opportunistic multiuser scheduling*.

Opportunistic multiuser scheduling has since been a topic of great interest to researchers, (e.g., [2]- [7]). While a majority of the literature on the topic studied the means to exploit the multiuser diversity already present in the network, some literature (e.g., [3]) propose strategies to artificially introduce multiuser diversity in the network, when the channel fading is not large enough (happens when a significantly strong line of sight component exists leading to low scattering) and hence the amount of multiuser diversity in the network is too small to be meaningfully exploited. An

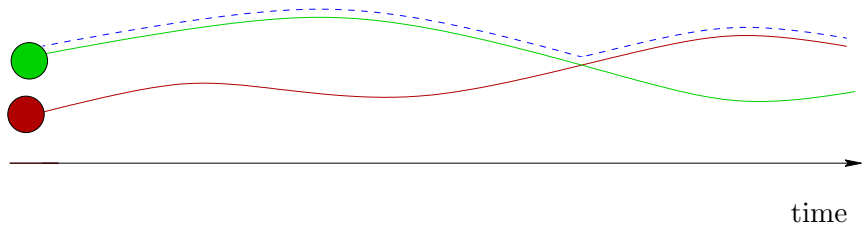

Figure 1.2: Illustration of opportunistic scheduling by 'riding the peak' of the channel strengths. Each circle indicates a user.

instance of commercial deployment of opportunistic scheduling strategy is the *proportional fair schedulers* implemented in cellular systems. Here users are prioritized, at the base station, based on the ratio of their instantaneous channel strengths to a measure of service received so far by the user. This scheduler aims to strike a balance between overall network throughput and network level fairness when the channel statistics of the users are asymmetric.

1.1 Motivation

The success of opportunistic scheduling, understandably, banks heavily on the availability of reliable information on the instantaneous channel strengths of the users, at the scheduler. A majority of the available literature, while being instrumental in enhancing our understanding of multiuser diversity and the means to exploit it, makes the following simplifying assumption: information on the channel strengths is readily available at the scheduler or the resources spent in acquiring this information is negligible. In reality, however, at regular intervals, each network user must spend

valuable resources in measuring the channel strength and reporting this information back to the scheduler. In addition, the overhead caused by this feedback creates additional strain on the reverse link that may be as scarce as the forward link. This is particularly true in recent and upcoming applications like mobile teleconferencing, video chat, gaming etc. Thus the loss of network resources associated with measuring and reporting the channel strengths to the scheduler has the potential to offset the gains associated with opportunistic scheduling [7]. It follows that the problem of acquiring channel state information is tightly coupled with the problem of exploiting multiuser diversity and it is therefore the need of the hour to design efficient joint channel information acquisition - opportunistic scheduling mechanisms.

Towards this end, we take a step back and focus on the channel modeling philosophy at the physical layer. The physical channels in the network, that suffer from fading and shadowing effects of the environment, are traditionally abstracted by Rayleigh or Rician flat fading models, depending on the strength of the line of sight component. It is assumed, in a flat fading model, that the fading coefficient evolves from one slot to another independently, i.e., without any memory. The reality, however, is different. There is a non-negligible amount of time correlation, i.e., memory in the channels, requiring the use of more realistic fading models. It has been reported [8, 9] that the first order, finite state Markov chain (Fig. 1.3) is known to abstract the fading channels with reasonable accuracy. A new line of work (e.g., [10–14]) have recognized this and adopted the first order, finite state, Markov chain to represent fading channels with memory and studied opportunistic multiuser scheduling in various networks. Despite capturing the memory in the channels, these

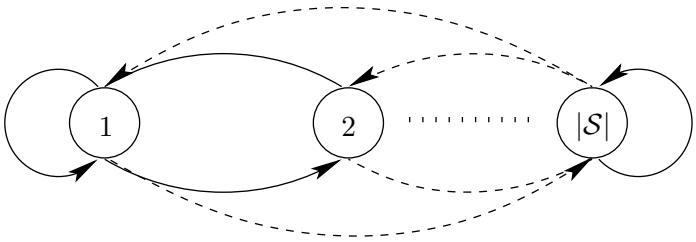

Figure 1.3: A first order, finite state Markov chain with state space $\mathcal{S}$.

works assume that the channel state information is readily available at the scheduler. Another line of work (e.g., [15, 16]), however, does not make this simplifying assumption and attempts to exploit the memory in the Markov-modeled channels to gather the channel state information, for purposes unrelated to opportunistic multiuser scheduling. These two lines of work can be combined to create a new design paradigm: model the channels with memory and use this memory to estimate the channel state information for opportunistic scheduling. This forms the central theme of this dissertation.

1.2 Contributions and Outline

We consider network models with the following common elements:

- Networks are centralized, i.e., communication happens between a central entity (the scheduler) and multiple users.

- Time is slotted and the channels between the users and the scheduler are modeled by first order, finite state Markov chains — not necessarily independent or identical across users.

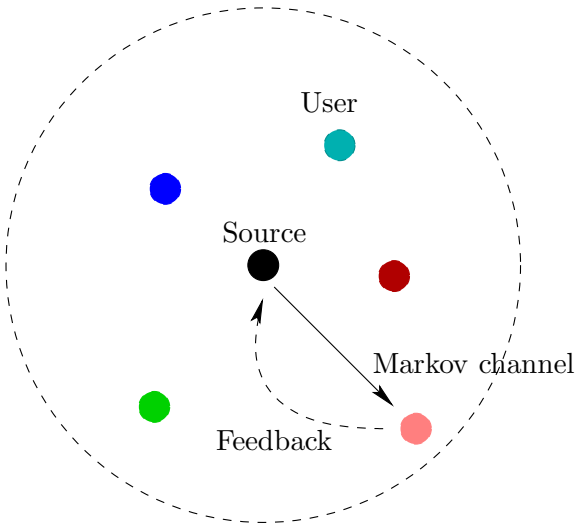

Figure 1.4: An Illustration of the general one-to-many scheduling model.

- The scheduler in each slot must schedule transmission to one of the users, without actual knowledge on the instantaneous channel state of the users and based solely on the belief values of the channels. The overall goal of the scheduler is to maximize the system sum-throughput.

- The scheduled user, at the end of the transmission slot, sends accurate feedback on the channel state in that slot to the scheduler. This end-of-slot feedback, under specific network environments, could be visualized as the Automatic Repeat reQuest (ARQ) feedback that is prevalent in communication protocols.

- The scheduler uses the feedback from the scheduled user, along with the memory inherent in the Markovian channels, to create belief values of the channels, which are, in turn, used for future scheduling decisions.

Thanks to the last element in the model, in any slot, scheduling transmission to a user is associated with the following two (potentially contradicting) objectives:

- **Exploitation**: Schedule the user with the best (perceived) channel condition at the moment — this corresponds to immediate gains in throughput.

- **Exploration**: Schedule a user and thus probe a user's channel to gain better overall understanding of the network channels, and hence better opportunistic scheduling, in the future. This, however, may require a compromise on the immediate gains in throughput.

We capture this trade-off by modeling the scheduling problem as a Partially Observable Markov Decision Process (POMDP) ([17]- [22]). In POMDPs, a system controller must take an action based on partial observations of the underlying system state. After each action, the controller accrues an immediate reward that is a function of the underlying state and the action taken. The system then evolves to the next state, probabilistically dependent on the current state and action. A simple illustration of a POMDP is provided in Fig. 1.5.

It must be noted that POMDPs are traditionally known to be analytically intractable and computationally expensive to solve. Although various 'one-size-fits-all' exact and approximate numerical solutions are available in the literature, they do not usually provide insights into the problem at hand. Taking note of these, we adopt the following approach in our study of opportunistic scheduling in various networks:

- Study the scheduling problem in an optimality framework that results in identifying crucial structural properties of the optimal scheduling policy.

- Analytically characterize the optimal scheduler whenever possible.

- In other cases, strongly founded on the optimality framework, derive near-optimal scheduling policies that are computationally inexpensive to implement.

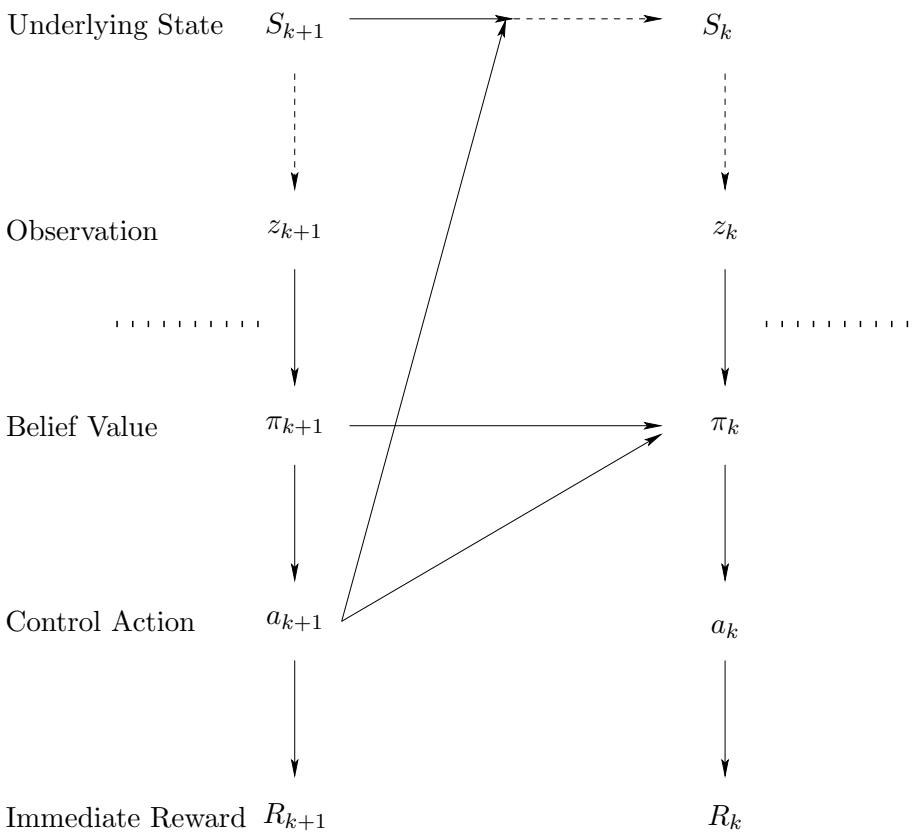

Figure 1.5: Illustration of a partially observable Markov decision process. Dotted arrows indicate probabilistic connections

Specifically, in Chapter 2, we consider opportunistic scheduling using 1-bit channel feedback in broadcast networks modeled by Markovian channels. Here, we address throughput/energy aware scheduling where, during every scheduling slot, the scheduler either *transmits* data to *all* users or stays *idle*. While a *transmit* decision corresponds to throughput gain, *idle* decision corresponds to energy savings. If one considers the set of all broadcast users as a single user and the *idle* decision as transmitting to a virtual user with time-invariant channel state, then this scheduling problem is essentially opportunistic *multiuser* scheduling in the traditional sense. By formulating the scheduling problem as a POMDP over an infinite horizon with discounted reward, we obtain the following main results: For particular ranges of the system parameters, we show that the optimal scheduling policy is greedy or partially greedy, depending on the case. For the general scenario, using an indirect approach, we perform an optimality analysis of the scheduling problem and derive a threshold scheduling policy that is easy to implement and near-optimal.

In Chapter 3, we study opportunistic scheduling using ARQ feedback in cellular downlink modeled by Markovian channels. We consider a general setup where the ARQ feedback from the users is assumed to be delayed randomly. We show that, despite the complicated dynamics between the channel information acquisition and scheduling mechanisms, a simple greedy policy is optimal when the number of downlink users is two. For higher number of users, we show that the greedy policy is strictly suboptimal and that it has near-optimal performance. We then study the structure of the greedy policy and show that it can be implemented via a simple algorithm that does not require the statistics of the underlying Markov chain nor the statistics of the feedback delay. By establishing an equivalence between the downlink

and a genie-aided system, we perform a fundamental capacity region analysis of the downlink.

In Chapter 4, we study opportunistic scheduling in cellular downlink when the channel state feedback is instantaneous (i.e., end of slot) and the channel is modeled by three-states. The purpose of this model is to study the effect of increasing the Markovian channel state space on the scheduling problem studied in Chapter 3. It turns out, many of the elegant structural results identified in Chapter 3 vanishes even when the size of the state space is increased by one. We then study the structural properties of the greedy policy and derive simple algorithms for its implementation.

In Chapter 5, we study opportunistic scheduling in multi-cellular downlink assuming the ARQ feedback is instantaneous. We focus on a two-cellular system since our analysis can be readily extended to the multi-cellular systems. We address the scheduling problem by following a two layered approach: the well established 'cell breathing' based inter-cell interference (ICI) control mechanism is adopted and assumed to be in place. On top of this layer we optimize ARQ based scheduling across the cells. We consider two scenarios: when the cooperation between the cells is asymmetric and when it is symmetric. Under asymmetric cooperation, the optimal scheduling policy has a greedy flavor and is simple to implement. Under symmetric cooperation, however, since a direct optimality analysis appears difficult, we formulate the scheduling problem as a more general variant of the restless multiarmed bandit processes [23] and study it from the perspective of Whittles indexability. Whittles indexability is an important condition that is known to predispose the Whittles index policy towards optimality in various RMAB processes. By linking the indexability

analysis to the broadcast scheduling problem studied in Chapter 2, we propose an index policy that is easy to implement and has near-optimal performance.

We summarize our work along with a discussion on future directions for this research topic in Chapter 6.

CHAPTER 2

OPPORTUNISTIC SCHEDULING USING 1-BIT FEEDBACK IN BROADCAST NETWORKS

2.1 Background

In this chapter we study energy aware, joint channel estimation - opportunistic scheduling in broadcast networks with Markov-modeled channels. We first give a brief background on broadcast networks and related literature. In broadcast networks, a designated source node (scheduler) attempts to transmit a packet to all users in the network. An integral component of mobile ad-hoc and sensor networks [24], broadcast plays a crucial role in a variety of protocols that provide basic functionality to higher layer services (e.g., [25]). In sensor networks, broadcast is used for coordinated and distributed computing (e.g., [26]). Thanks to the limited life of the mobile node batteries and a limited ability to replenish these batteries, energy aware transmission scheduling in broadcast networks is an important design consideration. This is particularly true in sensor networks where nodes are often deployed in hard to access or hostile environments. A large volume of work (e.g., [27]- [33]) is available for energy efficient communication in wireless networks - broadcast and otherwise. The reader is directed to [34] for an excellent exposition on the topic. Much of these works, while

providing valuable insights into energy efficient network design, are lacking in one of two ways: the physical channel considerations are disregarded and the problem is studied exclusively at the upper layers or, if the physical channel is indeed included in the design, the instantaneous channel state is assumed to be readily available at the scheduler. We address both of these issues in this chapter. The detailed problem setup is described next.

2.2 Problem Setup

2.2.1 Channel Model

We consider an N user broadcast. As mentioned in Chapter 1, the channel between the base station and each broadcast user is modeled by an $i.i.d$ two-state Markov chain (GE model, [35]). Assuming packetized data transmissions, each state corresponds to the degree of decodability of the packet sent through the channel. State 1 (ON) corresponds to full decodability, while state 0 (OFF) corresponds to zero decodability. Time is slotted and the channel of each user remains fixed for a slot and moves into another state in the next slot following the state transition probability of the Markov chain. The time slots of all users are synchronized. The two-state Markov channel is characterized by a 2×2 probability transition matrix

$$P = \begin{bmatrix} p & 1-p \\ r & 1-r \end{bmatrix}, \tag{2.1}$$

where

$$p := \text{prob(channel is in ON state in the current slot}|\text{ channel was in ON state in the previous slot)}$$

$$r := \text{prob(channel is in ON state in the current slot}|\text{ channel was in OFF state in the previous slot).}$$

Note that the Markov channel states can be interpreted as a quantized representation of the underlying channel strength lying on a continuum. Since, in realistic scenarios, the channel strength can be expected to evolve gradually over time (positive correlation), we assume $p > r$ throughout this chapter.

2.2.2 Scheduling Problem

In each time slot, the scheduler makes one of the following two decisions: (1) *transmit* (broadcast) a packet to the users, or (2) stay *idle*. While a broadcast transmission is associated with a throughput gain (and a concurrent energy loss), an *idle* decision corresponds to energy savings (and a concurrent loss in throughput). Our reward structure reflects this trade-off – Upon *transmit* decision, the scheduler accrues a reward of 1 for each user that successfully decodes the broadcast packet. If an *idle* decision is made, a reward of W (reward for passivity - corresponding to energy savings) is accrued at the scheduler. The exact reward structure will be described in the next subsection. The packets to be broadcast to the users are stored in an infinite queue at the scheduler. Upon *transmit* decision, the scheduler broadcasts the head of line packet to the users and drops it from the queue. At the end of the slot, each user attempts to decode the packet and sends back bit 1 (decoding success) or bit 0 (decoding failure) to the scheduler, over an error-free feedback channel. By the

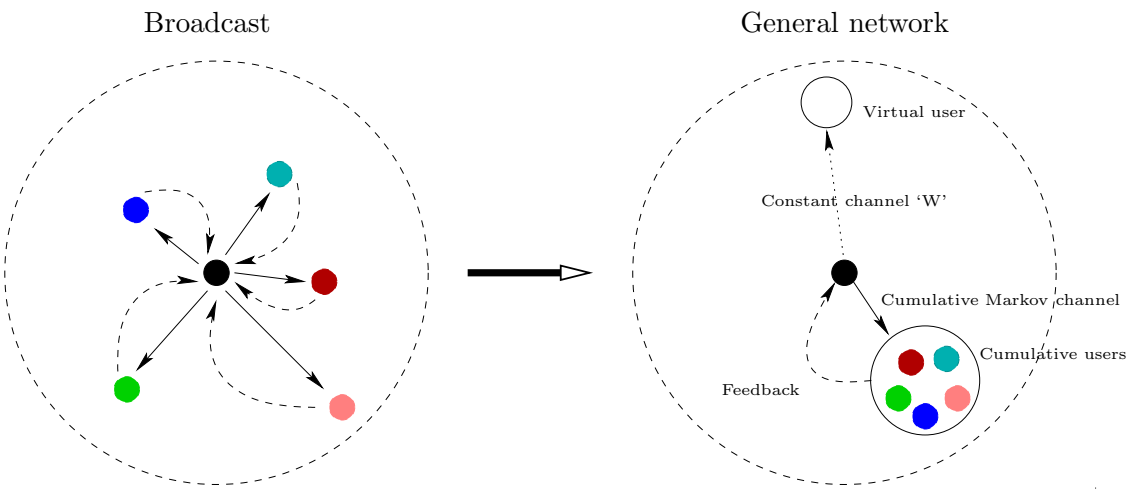

Figure 2.1: Illustration showing the broadcast scheduling model as a special case of the general one-to-many scheduling model.

definition of the two-state Markov channel defined earlier, this 1-bit feedback from an user at the end of a slot gives the state of the channel of that user in that slot. The scheduler collects this state feedback from all the users and creates a belief value of the channel state of the users in the next slot, using the Markov channel statistics. These belief values are used by the scheduler to make *transmit/idle* decisions in future slots. The scheduling problem is thus a dynamic program [36], more specifically a partially observable Markov decision process [17]. Also, note that, we can consider the *idle* decision as scheduling transmission to a virtual user with constant reward (W), and the *transmit* decision as scheduling to a single cumulative user made up of all the broadcast users. Thus the broadcast scheduling problem fits into the general scheduling model described in Chapter 1. This is illustrated in Fig. 2.1.

We now formally define the problem below.

2.2.3 Formal Problem Definition

Horizon: The number of consecutive time slots over which the scheduling decisions are made is the horizon. Throughout this chapter, we focus on the infinite horizon scenario.

Belief vector: Let $\pi = (\pi_1, \ldots, \pi_N) \in [0,1]^N$ be the vector of belief values in the current slot with π_i denoting the belief value of the channel of user $i \in \{1, \ldots, N\}$. It is well known [18] that the belief values are sufficient statistics to any information about the channels in the past slots, in our case, the scheduling decisions and the 1-bit feedbacks from the past. Thus the scheduling decision in any slot can be solely based on the belief values for that slot and (instead of the past schedule or feedback information).

Action: Let $a \in \{0,1\}$ indicate the action (scheduling decision) taken in the current slot. Let $a = 1$ correspond to the *transmit* decision and $a = 0$ correspond to the *idle* decision.

1-bit feedback: Upon the *transmit* decision, at the end of the slot, each user i in the broadcast determines if the reception was successful and sends back a 1-bit feedback f_i (1 for success and 0 for failure). Feedback f_i is one-to-one mapped to the state of the channel of user i in the corresponding slot.

Expected immediate reward: In each slot, if a *transmit* decision is made, the scheduler accrues a reward of 1 for each user that successfully decodes the broadcast packet. If an *idle* decision is made, a reward of W (reward for passivity - corresponding to energy savings) is accrued at the scheduler. Thus the expected immediate reward

accrued by the scheduler as a function of the belief vector and action is given by

$$R(\pi, a) \;=\; \begin{cases} \sum_i \pi_i, & \text{if } a = 1 \\ W, & \text{if } a = 0. \end{cases} \tag{2.2}$$

Stationary scheduling policy: A stationary scheduling policy $\mathfrak{A}$ is a stationary mapping from the belief vector π to an action as follows:

$$\mathfrak{A} : \pi \rightarrow a \in \{0, 1\}.$$

Expected total discounted reward under $\mathfrak{A}$: Under a policy $\mathfrak{A}$, for initial belief vector π, the expected (infinite horizon) total discounted reward is given by

$$V_{\mathfrak{A}}(\pi) \;=\; R(\pi, a) + \beta E\big[V_{\mathfrak{A}}(\pi^+)\big] \tag{2.3}$$

where $a = \mathfrak{A}(\pi)$ and $\beta \in [0, 1)$ is the discount factor that determines the relative weight between the immediate and the future rewards. The expectation is over the belief vector in the next slot, i.e., π^+, which in turn is a function of the scheduling decision a and (upon the *transmit* decision) the 1-bit feedback from the users. We now proceed to explicitly express the total discounted reward under $\mathfrak{A}$. For notational simplicity, we first define a few quantities. Let $\Pi_0 \doteq (r, r, \ldots, r, r)$, $\Pi_1 \doteq (r, r, \ldots, r, p)$, $\Pi_2 \doteq (r, r, \ldots, p, r), \ldots, \Pi_{2^N-1} \doteq (p, p, \ldots, p, p)$. Let $P_0(\pi) \doteq (1-\pi_1)(1-\pi_2)\ldots(1-\pi_{N-1})(1-\pi_N)$, $P_1(\pi) \doteq (1-\pi_1)(1-\pi_2)\ldots(1-\pi_{N-1})(\pi_N)$, $P_2(\pi) \doteq (1-\pi_1)(1-\pi_2)\ldots(\pi_{N-1})(1-\pi_N), \ldots, P_{2^N-1}(\pi) \doteq \pi_1\pi_2\ldots\pi_{N-1}\pi_N$. Define the operator $T(.)$ as the evolution of the belief value of a Markov channel to the next slot under the *idle* decision. Thus, if $x \in [0, 1]$ is the belief value, then $T(x) = xp + (1-x)r$. The total discounted reward under $\mathfrak{A}$ is now explicitly given by

$$V_{\mathfrak{A}}(\pi) \;=\; \begin{cases} \sum_i \pi_i + \beta \sum_{j=0}^{2^N-1} P_j(\pi) V_{\mathfrak{A}}(\Pi_j), & \text{if } a = \mathfrak{A}(\pi) = 1 \\ W + \beta V_{\mathfrak{A}}(T(\pi)), & \text{if } a = \mathfrak{A}(\pi) = 0. \end{cases} \tag{2.4}$$

Optimal scheduling policy: For a given belief vector π, the *optimal* total discounted reward (henceforth, simply the *total discounted reward*), $V(\pi)$, is given by the Bellman equation [36]

$$V(\pi) = \max\{\sum_i \pi_i + \beta \sum_{j=0}^{2^N-1} P_j(\pi) V(\Pi_j), W + \beta V(T(\pi))\}. \quad (2.5)$$

By standard dynamic programming theory [36], a stationary policy $\mathfrak{A}^*$ is optimal if and only if the total discounted reward under $\mathfrak{A}^*$, i.e., $V_{\mathfrak{A}^*}(\pi)$, satisfies the Bellman equation in (2.5) for every $\pi \in [0,1]^N$, i.e., $\mathfrak{A}^*$ is optimal if and only if

$$V_{\mathfrak{A}^*}(\pi) = \max\{\sum_i \pi_i + \beta \sum_{j=0}^{2^N-1} P_j(\pi) V_{\mathfrak{A}^*}(\Pi_j), W + \beta V_{\mathfrak{A}^*}(T(\pi))\}. \quad (2.6)$$

2.3 Optimal Scheduling Policy - Partial Characterization and Thresholdability Properties

2.3.1 Partial Characterization of the Optimal Scheduling Policy

Define $V^a(\pi)$ as the expected total discounted reward upon the *transmit* (active) decision in the current slot and optimal decisions in all future slots and $V^p(\pi)$ as the expected total discounted reward upon *idle* (passive) decision in the current slot and optimal decisions in all future slots, i.e.,

$$\begin{aligned} V^a(\pi) &= \sum_i \pi_i + \beta \sum_{j=0}^{2^N-1} P_j(\pi) V(\Pi_j) \\ V^p(\pi) &= W + \beta V(T(\pi)). \end{aligned} \quad (2.7)$$

Let $\mathcal{A}$ and $\mathcal{P}$ be the regions in the state space, $[0,1]^N$, where it is optimal to *transmit* and *idle*, respectively. Formally,

$$\pi \in \begin{cases} \mathcal{A}, & \text{if } V^a(\pi) \geq V^p(\pi) \\ \mathcal{P}, & \text{if } V^a(\pi) < V^p(\pi). \end{cases} \quad (2.8)$$

We now report our result on the optimal scheduling policy when the reward for passivity $W \notin (Nr, Np)$.

Proposition 1. *When $W \notin (Nr, Np)$, the optimal scheduling policy is greedy, i.e.,*

$$\pi \in \begin{cases} \mathcal{A}, & \text{if } \sum_i \pi_i \geq W \\ \mathcal{P}, & \text{if } \sum_i \pi_i < W \end{cases}.$$

Proof. Consider the greedy policy, $\hat{\mathfrak{A}} : (\pi, W) \rightarrow \arg\max_{a \in \{1,0\}}(R(\pi, a))$ where $R(\pi, 1) = \sum_i \pi_i$ and $R(\pi, 0) = W$. In order to prove the optimality of the greedy policy, it is sufficient to prove that the total discounted reward achieved by the greedy policy satisfies the Bellman equation [36], i.e.,

$$V_{\hat{\mathfrak{A}}}(\pi) \quad = \quad \max\{\sum_i \pi_i + \beta \sum_{j=0}^{2^N-1} P_j(\pi) V_{\hat{\mathfrak{A}}}(\Pi_j), W + \beta V_{\hat{\mathfrak{A}}}(T(\pi))\}. \tag{2.9}$$

Since the scheduling problem is a *discounted reward* dynamic program, the infinite horizon reward can be interpreted as a limit on the finite horizon reward [36]. Thus the first quantity in the $\max\{.,.\}$ operator in (2.9) can be interpreted as the limiting value (as horizon $\rightarrow \infty$) on the total discounted reward over a finite horizon when the *transmit* decision is made in the current slot and the greedy policy is implemented in all future slots until the horizon. Likewise, the second quantity corresponds to the limit on the total discounted reward when the *idle* decision is made in the current slot and the greedy policy is implemented in all future slots.

Now, note that, with π being the current belief vector, in any future slot, independent of whether a *transmit* or *idle* decision was made in the current slot, the belief vector lies in the following set:

$$\{T^u(\pi)|_{u=\{1,2,\ldots\}}, T^v(\Pi_0)|_{v\in\{0,1,\ldots\}}, \ldots, T^v(\Pi_{2^N-1})|_{v\in\{0,1,\ldots\}}\}. \tag{2.10}$$

For any vector, $(x_1, \ldots, x_N)$, in the preceding set, $Nr \leq \sum_i x_i \leq Np$ since $T(x) = xp + (1-x)r \in [r, p]$.

Consider the case $W \leq Nr$. From the preceding discussion, the sum of belief values in any future slot is at least as high as W and hence the greedy policy would choose to *transmit* in all future time slots independent of whether the scheduler decides to *transmit* or *idle* in the current slot. Now, since the underlying Markov channel behavior is unchanged by the scheduling decisions, the future discounted reward, under the greedy policy in all future slots, is the same after *transmit* or *idle* decision in the current slot. Likewise, when $W \geq Np$, in any future slot, the sum of the belief values is no more than W. Thus, the greedy policy would choose to stay *idle* in all future slots independent of the current scheduling decision. This equates the future discounted rewards after *transmit* and *idle* decisions in the current slot, if the greedy policy is implemented in all future slots. Thus $\beta \sum_{j=0}^{2^N-1} P_j(\pi) V_{\hat{\mathfrak{A}}}(\Pi_j) = \beta V_{\hat{\mathfrak{A}}}(T(\pi))$ when $W \notin (Nr, Np)$. The condition for optimality of the greedy policy, when $W \notin (Nr, Np)$, is now rewritten from (2.9) as

$$V_{\hat{\mathfrak{A}}}(\pi) \;=\; \begin{cases} \sum_i \pi_i + \beta \sum_{j=0}^{2^N-1} P_j(\pi) V_{\hat{\mathfrak{A}}}(\Pi_j), & \text{if } \sum_i \pi_i \geq W \\ W + \beta V_{\hat{\mathfrak{A}}}(T(\pi)), & \text{if } \sum_i \pi_i < W. \end{cases} \tag{2.11}$$

This is indeed satisfied by the definition of the greedy policy, thus establishing the proposition. □

We now introduce some preparatory results on the structure of the optimal reward functions.

Lemma 1. *The total discounted reward $V(\pi_1, \ldots, \pi_N)$ is component-wise convex over the N-dimensional state space, i.e., for any $i \in \{1, \ldots, N\}$ and $(\pi_1, \ldots, \pi_{i-1}, \pi_{i+1}, \ldots, \pi_N)$*

fixed,

$$\alpha V(\pi_1, \ldots, \pi_{i-1}, x_i, \pi_{i+1}, \ldots, \pi_N) + (1-\alpha) V(\pi_1, \ldots, \pi_{i-1}, y_i, \pi_{i+1}, \ldots, \pi_N)$$
$$\geq V(\pi_1, \ldots, \pi_{i-1}, \alpha x_i + (1-\alpha) y_i, \pi_{i+1}, \ldots, \pi_N)$$

for any $\alpha \in [0,1]$ *and* $x_i, y_i \in [0,1]$.

Proof. Reinterpreting the infinite horizon total discounted reward [36] as the limit on the finite horizon reward, we have

$$V(\pi) = \lim_{t \to \infty} V_t(\pi) \tag{2.12}$$

with

$$\begin{aligned} V_t(\pi) &= \max\{V_t^a(\pi), V_t^p(\pi)\} \\ V_t^a(\pi) &= \sum_i \pi_i + \beta \sum_{j=0}^{2^N-1} P_j(\pi) V_{t-1}(\Pi_j) \\ V_t^p(\pi) &= W + \beta V_{t-1}(T(\pi)). \end{aligned} \tag{2.13}$$

We have used the convention of decreasing time index up to the horizon at $t = 1$. Note that V_t^a and V_t^p are defined along the lines of V^a and V^p in (2.7). The terminal reward, i.e., the value function at the horizon, is given by

$$V_1(\pi) = \max\{\sum_i \pi_i, W\}. \tag{2.14}$$

Note, from (2.13), that $V_t^a(\pi)$ is linear in π_i with $\pi_j, \forall j \neq i$, fixed.

Now, assume the following condition holds (induction hypothesis (H_0)): *For* $t \geq 2$, $V_{t-1}(\pi)$ *is component-wise convex.*

With $V_t^p(\pi) = W + \beta V_{t-1}(T(\pi))$, it follows that, for $i \in \{1, \ldots, N\}$,

$$\begin{aligned}
\frac{dV_t^p(\pi)}{d\pi_i} &= \beta \frac{dV_{t-1}(T(\pi_1), \ldots, T(\pi_i), \ldots, T(\pi_N))}{dT(\pi_i)} \frac{dT(\pi_i)}{d\pi_i} \\
&= \beta(p-r) \frac{dV_{t-1}(T(\pi_1), \ldots, T(\pi_{i-1}), \pi_i, T(\pi_{i+1}), \ldots, T(\pi_N))}{d\pi_i} |_{\pi_i(p-r)+r}
\end{aligned} \tag{2.15}$$

where we have used $T(x) = x(p-r) + r$. Differentiating again with respect to π_i,

$$\begin{aligned}
\frac{d^2V_t^p(\pi)}{d\pi_i^2} &= \beta(p-r)^2 \frac{d^2V_{t-1}(T(\pi_1), \ldots, T(\pi_{i-1}), \pi_i, T(\pi_{i+1}), \ldots, T(\pi_N))}{d\pi_i^2} |_{\pi_i(p-r)+r} \\
&\geq 0
\end{aligned} \tag{2.16}$$

since V_{t-1} is component-wise convex (hypothesis (H_0)). Thus V_t^p is component-wise convex.

With the optimal reward given by $V_t(\pi) = \max(V_t^a(\pi), V_t^p(\pi))$ and since V_t^a is component-wise linear and V_t^p is component-wise convex, we have V_t is component-wise convex. Note that the terminal reward, $V_1(\pi) = \max\{\sum_i \pi_i, W\}$, is linear in π and hence can be considered to be component-wise convex. Thus, using induction, $V_t(\pi)$, for any $t \in \{1, 2, \ldots\}$ and hence $V(\pi)$ (from 2.12) is component-wise convex. This establishes the lemma. □

We now compare the future discounted reward corresponding to the *transmit* decision in the current slot with that of the *idle* decision in the current slot. Intuition suggests that probing the channels at the end of the slot (associated with the *transmit* decision) results in a higher future reward than when probing is not performed (*idle*). We formally establish this using the component-wise convexity property of V below.

Lemma 2. *For any belief vector π, the future discounted reward after a transmit decision in the current slot is at least as high as the future discounted reward after an idle decision, i.e., $\beta \sum_{j=0}^{2^N-1} P_j(\pi) V(\Pi_j) \geq \beta V(T(\pi))$.*

Proof. Consider the future discounted reward after *transmit* decision;

$$\begin{aligned}
&\beta \sum_{j=0}^{2^N-1} P_j(\pi) V(\Pi_j) \\
&= \beta\Big(P_0(\pi)V(\Pi_0) + \ldots + P_{2^N-1}(\pi)V(\Pi_{2^N-1}))\Big) \\
&= \beta\Big((1-\pi_1)(1-\pi_2)\ldots(1-\pi_{N-1})(1-\pi_N)V(r,r,\ldots,r,r,r) \\
&\quad +(1-\pi_1)(1-\pi_2)\ldots(1-\pi_{N-1})(\pi_N)V(r,r,\ldots,r,r,p) \\
&\quad +(1-\pi_1)(1-\pi_2)\ldots(\pi_{N-1})(1-\pi_N)V(r,r,\ldots,r,p,r) \\
&\quad +\ldots+\pi_1\pi_2\ldots\pi_N V(p,p,\ldots,p,p,p)\Big) \\
&\geq \beta\Big((1-\pi_1)(1-\pi_2)\ldots(1-\pi_{N-1})V(r,r,\ldots,r,r,T(\pi_N)) \\
&\quad +(1-\pi_1)(1-\pi_2)\ldots(\pi_{N-1})V(r,r,\ldots,r,p,T(\pi_N)) \\
&\quad +\ldots+\pi_1\pi_2\ldots\pi_{N-1}V(p,p,\ldots,p,p,T(\pi_N))\Big) \\
&\geq \ldots \geq \beta V(T(\pi_1),\ldots,T(\pi_N)) \\
&= \beta V(T(\pi)) \qquad (2.17)
\end{aligned}$$

which gives the future discounted reward after *idle* decision. Note that we have used the component-wise convexity property of V (Lemma 1) in (2.17). □

From the preceding lemma, we readily conclude that, if the immediate reward corresponding to *transmit* decision is at least as high as the immediate reward corresponding to an *idle* decision, then it is optimal to *transmit*, thus giving a partial greedy flavor to the optimal scheduling policy. We formalize this below.

Proposition 2. *For any W, the optimal policy has the following partial structure*

$$\pi \in \mathcal{A}, \text{ if } \sum_i \pi_i \geq W \tag{2.18}$$

It is worth noting that the energy loss per broadcast action (*transmit*) and hence W is independent of the number of broadcast users, N. Thus as N increases, the throughput gain (*transmit*) progressively outweighs the energy savings (*idle*). It follows that the optimal policy would increasingly choose to *transmit* than *idle*, with increasing broadcast size. This intuition is supported by the result in Proposition 2.

2.3.2 Thresholdability Properties of the Optimal Policy in the Two User Broadcast

We now proceed to establish structural properties of the value functions V, V^a and V^p and hence the optimal scheduling policy in the two-user broadcast.

Lemma 3. *With $\pi_2 = \pi_1 k + c$, the reward functions $V^a(\pi_1, \pi_2 = \pi_1 k + c)$, $V^p(\pi_1, \pi_2 = \pi_1 k + c)$ and $V(\pi_1, \pi_2 = \pi_1 k + c)$ are convex and increasing in π_1 for $k \geq 0$. For $k < 0$, $V^a(\pi_1, \pi_2 = \pi_1 k + c)$ is concave in π_1 while $V^p(\pi_1, \pi_2 = \pi_1 k + c)$ and $V(\pi_1, \pi_2 = \pi_1 k + c)$ are piecewise concave in π_1. When $k = -1$, V^p and V attain their maximum at $\pi_1 = \pi_2$, i.e.,*

$$\arg\max_{\pi_1} V^p(\pi_1, \pi_2 = -\pi_1 + c) = \frac{c}{2}$$
$$\arg\max_{\pi_1} V(\pi_1, \pi_2 = -\pi_1 + c) = \frac{c}{2}$$

The quantities $\gamma \doteq V(p,p) + V(r,r) - 2V(p,r)$ and $\alpha \doteq V(p,r) - V(r,r)$ are non-negative.

Proof. The proof is tedious and is therefore moved to the appendix. The proof proceeds by carefully reinterpreting the infinite horizon problem as a limit on the finite

horizon problem. We then study the reward functions over the two dimensional state space by sweeping over $\pi_1 \in [0,1]$ with π_2 along specific *directions/axes* given by $\pi_2 = \pi_1 k + c$ for $k, c \in \mathbb{R}$. The lemma is then established using backward induction. □

We now identify a crucial structure in the evolution of the state pair under consecutive idle decisions.

Lemma 4. *Under consecutive idle decisions, the state pair* $(\pi_1, \pi_2) \in [0,1]^2$ *progressively evolves towards the steady state* (π_{ss}, π_{ss}) *and falls on the line segment between* (π_1, π_2) *and* (π_{ss}, π_{ss})*. Mathematically, with* $T^k(.)$ *denoting the* $k-$*step state evolution operator under* k *consecutive idle decisions,*

$$(T^{k+1}(\pi_1), T^{k+1}(\pi_2)) \quad \in \quad \mathcal{L}\big((T^k(\pi_1), T^k(\pi_2)), (\pi_{ss}, \pi_{ss})\big) \tag{2.19}$$

where $\mathcal{L}(x, y)$ *is the line segment between* x *and* y*.*

Proof. We first show that, for $k \geq 0$, $T^{k+1}(\pi_1, \pi_2)$ falls on the line segment between $(T^k(\pi_1), T^k(\pi_2))$ and (π_{ss}, π_{ss}). The vector from $(T^k(\pi_1), T^k(\pi_2))$ to $(T^{k+1}(\pi_1), T^{k+1}(\pi_2))$ is represented by

$$\begin{aligned}
&(T^{k+1}(\pi_1) - T^k(\pi_1), T^{k+1}(\pi_2) - T^k(\pi_2)) \\
&= (T^k(\pi_1)(p-r) + r - T^k(\pi_1), T^k(\pi_2)(p-r) + r - T^k(\pi_2)) \\
&= (r - T^k(\pi_1)(1-(p-r)), r - T^k(\pi_2)(1-(p-r))) \\
&= (1-(p-r))\big(\pi_{ss} - T^k(\pi_1), \pi_{ss} - T^k(\pi_2)\big)
\end{aligned} \tag{2.20}$$

where $(\pi_{ss} - T^k(\pi_1), \pi_{ss} - T^k(\pi_2))$ represents the vector from $(T^k(\pi_1), T^k(\pi_2))$ to (π_{ss}, π_{ss}). Thus, since $(1-(p-r)) \geq 0$, $T^{k+1}(\pi_1, \pi_2)$ falls on the line segment between

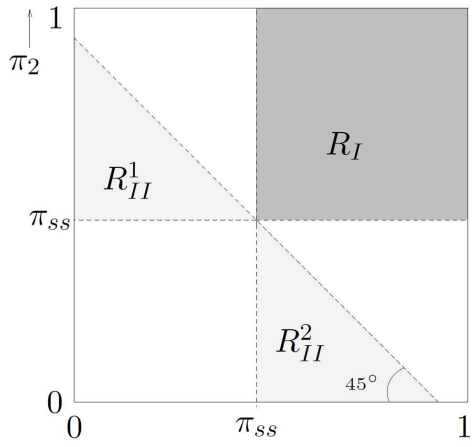

Figure 2.2: Illustration of the regions R_I, R_{II}^1 and R_{II}^2.

$(T^k(\pi_1), T^k(\pi_2))$ and (π_{ss}, π_{ss}), for any $k \geq 0$. Also, for progressively increasing k, $(T^k(\pi_1), T^k(\pi_2))$ progressively moves towards the steady state. This establishes the lemma. □

Using the structural results established so far, we now proceed to show that the optimal scheduling policy in the two-user broadcast is provably *thresholdable*[1] in specific regions of the two dimensional state space. We first classify the broadcast into two types based on the optimal scheduling decision at steady state, as below:

- Type I: If $(\pi_{ss}, \pi_{ss}) \in \mathcal{A}$
- Type II: If $(\pi_{ss}, \pi_{ss}) \in \mathcal{P}$

Referring to Fig. 2.2, let R_I denote the region $\{(\pi_1, \pi_2); \pi_1 \in [\pi_{ss}, 1], \pi_2 \in [\pi_{ss}, 1]\}$. Let R_{II} denote the union of the regions $R_{II}^1 \doteq \{(\pi_1, \pi_2); \pi_1 \in [0, \pi_{ss}], \pi_2 \in [\pi_{ss}, 2\pi_{ss} - \pi_1]\}$ and $R_{II}^2 \doteq \{(\pi_1, \pi_2); \pi_2 \in [0, \pi_{ss}], \pi_1 \in [\pi_{ss}, 2\pi_{ss} - \pi_2]\}$.

[1]The definition of *thresholdable* will soon be revealed within context.

We now record our result on the thresholdability property of the optimal scheduling policy in the Type-I two-user broadcast.

Proposition 3. *If the two-user broadcast is Type I, i.e.,* $V^a(\pi_{ss}, \pi_{ss}) \geq V^p(\pi_{ss}, \pi_{ss})$, *then*

(1) $R_I \subset \mathcal{A}$

(2) $V^a(\pi_{ss}, \pi_{ss}) = V^p(\pi_{ss}, \pi_{ss}) \Rightarrow R_{II} \subset \mathcal{P}$

(3) $V^a(\pi_{ss}, \pi_{ss}) > V^p(\pi_{ss}, \pi_{ss}) \Rightarrow$ *(thresholdability property) In the region* R_{II}^1, *if for* $k \in [-1, 0]$, $\exists$ *a* π_1^* *and* $\pi_2^* = \pi_1^* k + \pi_{ss}(1-k)$ *such that* $V^a(\pi_1^*, \pi_2^*) = V^p(\pi_1^*, \pi_2^*)$, *then*

$$(\pi_1, \pi_2 = \pi_1 k + \pi_{ss}(1-k)) \in \begin{cases} \mathcal{A}, & \text{if } \pi_1 \in (\pi_1^*, \pi_{ss}] \\ \mathcal{P}, & \text{if } \pi_1 \in [0, \pi_1^*] \end{cases}$$

If $\nexists$ *such a* (π_1^*, π_2^*), *then*

$$(\pi_1, \pi_2 = \pi_1 k + \pi_{ss}(1-k)) \in \mathcal{A} \ \forall \ \pi_1 \in [0, \pi_{ss}].$$

Similarly, in the region R_{II}^2, *if for* $k \in [-1, 0]$, $\exists$ *a* $\pi_2^* \in [0, \pi_{ss}]$ *and* $\pi_1^* = \pi_2^* k + \pi_{ss}(1-k)$ *such that* $V^a(\pi_1^*, \pi_2^*) = V^p(\pi_1^*, \pi_2^*)$, *then*

$$(\pi_1 = \pi_2 k + \pi_{ss}(1-k), \pi_2) \in \begin{cases} \mathcal{A}, & \text{if } \pi_2 \in (\pi_1^*, \pi_{ss}] \\ \mathcal{P}, & \text{if } \pi_2 \in [0, \pi_1^*] \end{cases}$$

If $\nexists$ *such a* (π_1^*, π_2^*), *then*

$$(\pi_1 = \pi_2 k + \pi_{ss}(1-k), \pi_2) \in \mathcal{A} \ \forall \ \pi_2 \in [0, \pi_{ss}].$$

Proof. The proposition is established using Lemma 3 and Lemma 4. The proof is tedious and hence moved to the appendix. □

Call the set of points (π_1^*, π_2^*) in R_{II} given by the third part of Proposition 3 as the threshold boundary. We now characterize this threshold boundary in region R_{II}.

Corollary 1. *Within region R_{II}, the threshold boundary is given by the upper segment of the hyperbola*

$$V^a(\pi_1, \pi_2) = W + \beta V^a(T(\pi_1), T(\pi_2))$$

where

$$\begin{aligned} V^a(x_1, x_2) &= x_1 + x_2 + \beta\big[(1-x_1)(1-x_2)V(r,r) + (1-x_1)(x_2)V(r,p) \\ &\quad + x_1(1-x_2)V(p,r) + x_1 x_2 V(p,p), \end{aligned}$$

$T(x) = x(p-r)+r$, and "upper segment" indicates the segment of the hyperbola that lies in the first quadrant around the asymptotes.

Proof. A point (π_1^*, π_2^*) in region R_{II} that falls on the threshold boundary, by definition, satisfies $V^a(\pi_1^*, \pi_2^*) = V^p(\pi_1^*, \pi_2^*)$ and the thresholdability property reported in Proposition 3. Thus, since $(T(\pi_1^*), T(\pi_2^*)) \in \mathcal{L}((\pi_1^*, \pi_2^*), (\pi_{ss}, \pi_{ss}))$, we have from the thresholdability property in Proposition 3, $(T(\pi_1^*), T(\pi_2^*)) \in \mathcal{A}$. Therefore,

$$\begin{aligned} V^p(\pi_1^*, \pi_2^*) &= W + \beta V(T(\pi_1^*), T(\pi_2^*)) \\ &= W + \beta V^a(T(\pi_1^*), T(\pi_2^*)) \end{aligned} \tag{2.21}$$

Substituting V^p in the equation $V^a(\pi_1^*, \pi_2^*) = V^p(\pi_1^*, \pi_2^*)$, the threshold boundary is given by $V^a(\pi_1, \pi_2) = W + \beta V^a(T(\pi_1), T(\pi_2))$. With algebraic manipulation, and using the expression for V^a from 2.7, the threshold boundary equation can be written

as $(\pi_1^* + \pi_2^*)A + \pi_1^*\pi_2^*B = C$, which is a hyperbola with

$$\begin{aligned} A &= 1 + \beta\alpha - \beta(p-r)\big(1 + \beta(r\gamma + \alpha)\big) \\ B &= (1 - \beta(p-r)^2)\beta\gamma \\ C &= W + \beta(2r(1+\beta\alpha) + r^2\gamma\beta). \end{aligned} \tag{2.22}$$

The asymptotes of the hyperbola are given by $\pi_1 = \frac{-A}{B}$ and $\pi_2 = \frac{-A}{B}$. The slope of the hyperbola with respect to π_1 is given by $\frac{d\pi_2}{d\pi_1} = \frac{-A^2 - BC}{(A+\pi_1 B)^2}$. Since $\gamma, \alpha \geq 0$ (lemma 5) and $(1 - \beta(p-r)^2) \geq 0$, we have $B \geq 0$ and $C \geq 0$. Thus the hyperbola has a negative slope with respect to π_1 and hence lies in the first and third quadrants around $(\frac{-A}{B}, \frac{-A}{B})$. We now proceed to show that the threshold boundary in R_{II} is given by the upper segment (first quadrant) of the hyperbola. Consider the following inequality involving the asymptote $\frac{-A}{B}$.

$$\begin{aligned} \frac{-A}{B} &< \pi_{ss} \\ \Leftrightarrow -\frac{1 + \beta\alpha - \beta(p-r)\big(1+\beta(r\gamma+\alpha)\big)}{(1-\beta(p-r)^2)\beta\gamma} &< \frac{r}{1-(p-r)} \\ \Leftrightarrow -(1 + \beta\alpha - \beta(p-r)(1+\beta(r\gamma+\alpha)))(1-(p-r)) &< r(1-\beta(p-r)^2)\beta\gamma \\ \Leftrightarrow -(1+\beta\alpha)(1-(p-r)) + \beta(p-r)(1+\beta(r\gamma+\alpha)) & \\ -\beta(p-r)^2(1+\beta\alpha) - r\beta\gamma &< 0 \\ \Leftarrow (1-(p-r))\Big[-(1-\beta(p-r)) - r\beta\gamma - \beta\alpha(1-(p-r))\Big] &< 0 \end{aligned} \tag{2.23}$$

The last statement is indeed true. Thus $\frac{-A}{B} < \pi_{ss}$ and hence the lower segment of the hyperbola that lies in the third quadrant centered at $(\frac{-A}{B}, \frac{-A}{B})$ does not intersect the region R_{II}. Therefore, the threshold boundary in R_{II} is given by the upper segment of the hyperbola. This completes the proof. □

An illustration of the threshold boundary in R_{II} is provided in Fig. 2.3(a).

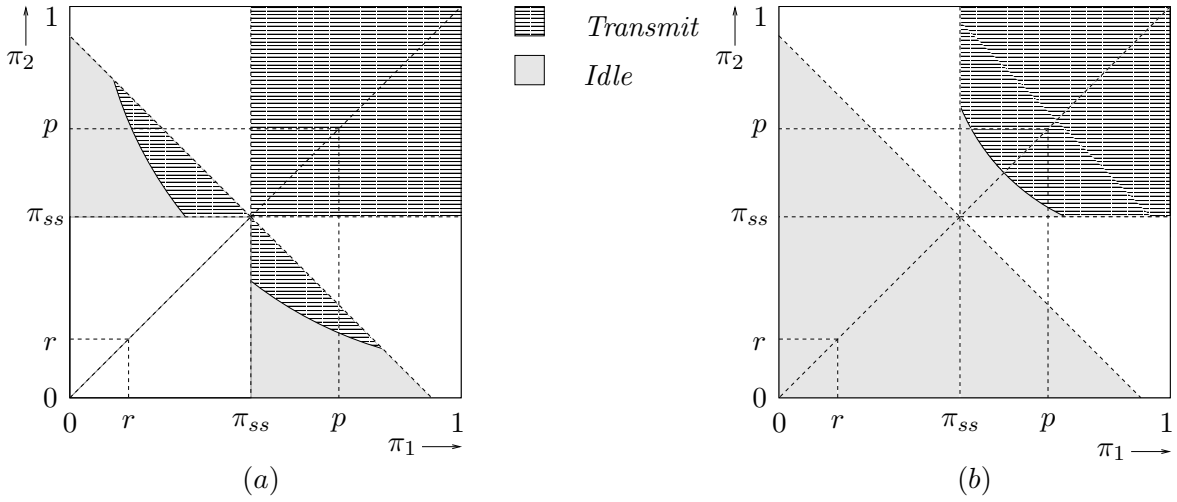

Figure 2.3: Illustration of the threshold boundaries when the broadcast is (a) Type I, (b) Type II.

We now record our result on the thresholdability of the optimal policy when the two-user broadcast is Type II.

Proposition 4. *If the two-user broadcast is Type II, then*

(1) $(\pi_1, \pi_2) \in \mathcal{P}$, $\forall \pi_1 + \pi_2 \leq 2\pi_{ss}$

(2) (Thresholdability property) In the region R_I*, if for* $k \geq 0$, $\exists$ *a* π_1^* *and* $\pi_2^* = \pi_1^* k + \pi_{ss}(1-k)$ *such that* $V^a(\pi_1^*, \pi_2^*) = V^p(\pi_1^*, \pi_2^*)$*, then*

$$(\pi_1, \pi_1^* k + \pi_{ss}(1-k)) \in \begin{cases} \mathcal{A}, & \text{if } \pi_1 \in [\pi_1^*, 1] \\ \mathcal{P}, & \text{if } \pi_1 \in [\pi_{ss}, \pi_1^*) \end{cases}$$

If $\nexists$ *such a* (π_1^*, π_2^*)*, then*

$$(\pi_1, \pi_2 = \pi_1 k + \pi_{ss}(1-k)) \in \mathcal{P} \ \forall \ \pi_1 \in [\pi_{ss}, 1].$$

Proof. The proposition is established along similar lines to Proposition 3. The reader is referred to the appendix for details. □

We proceed to characterize the threshold boundary in region R_I below.

Corollary 2. *Within region R_I, the threshold boundary is given by the upper segment of the hyperbola*

$$V^a(\pi_1, \pi_2) = \frac{W}{1-\beta}$$

where

$$\begin{aligned} V^a(x_1, x_2) &= x_1 + x_2 + \beta\big[(1-x_1)(1-x_2)V(r,r) + (1-x_1)(x_2)V(r,p) \\ &\quad + x_1(1-x_2)V(p,r) + x_1 x_2 V(p,p). \end{aligned}$$

Proof. A point (π_1^*, π_2^*) in region R_I that falls on the threshold boundary, by definition, satisfies $V^a(\pi_1^*, \pi_2^*) = V^p(\pi_1^*, \pi_2^*)$ and the thresholdability property of Proposition 4. We can write (π_1^*, π_2^*) as $(\pi_1^*, \pi_1^* k + \pi_{ss}(1-k))$ for some $k \geq 0$. Then, from the thresholdability result of Proposition 4, since $\pi_{ss} \leq T(\pi_1^*) \leq \pi_1^*$ (Lemma 4), $(T(\pi_1^*), T(\pi_2^*)) \in \mathcal{P}$. Therefore, the total reward corresponding to inactivate decision at (π_1^*, π_2^*) is given by

$$\begin{aligned} V^p(\pi_1^*, \pi_2^*) &= W + \beta V^p(T(\pi_1^*), T(\pi_2^*)) \\ &= W + \beta \frac{W}{1-\beta} \\ &= \frac{W}{1-\beta} \end{aligned} \tag{2.24}$$

where $V^p(T(\pi_1^*), T(\pi_2^*)) = \frac{W}{1-\beta}$ comes from $(T^l(\pi_1^*), T^l(\pi_2^*)) \in \mathcal{P}$ for $l \geq 1$. Substituting V^p in the equation $V^a(\pi_1^*, \pi_2^*) = V^p(\pi_1^*, \pi_2^*)$ the threshold boundary is given by $V^a(\pi_1, \pi_2) = \frac{W}{1-\beta}$ where V^a is given in (2.7). With algebraic manipulation, the threshold boundary equation can be written in the form of a hyperbola:

$(\pi_1^* + \pi_2^*)A + \pi_1^*\pi_2^* B = C$ where

$$\begin{aligned} A &= 1 + \beta\alpha \\ B &= \beta\gamma \\ C &= \frac{W}{1-\beta} - \beta V(r,r). \end{aligned} \tag{2.25}$$

with asymptotes given by $\pi_1 = \frac{-A}{B}$ and $\pi_2 = \frac{-A}{B}$. Since $\gamma, \alpha \geq 0$ (Lemma 3), we have $A \geq 0$ and $B \geq 0$. Note that, from Proposition 4, $(T^l(r), T^l(r)) \in \mathcal{P}\ \forall l \in \{0, 1, \ldots\}$, since $r \leq T^l(r) \leq \pi_{ss}$ and hence $2T^l(r) \leq 2\pi_{ss}$. Thus $V(r,r) = \frac{W}{1-\beta}$ and hence $C = \frac{W}{1-\beta} - \beta V(r,r) = \frac{W}{1-\beta}(1-\beta) \geq 0$.

It follows that the first derivative of the hyperbola with respect to π_1 given by $\frac{-A^2-BC}{(A+\pi_1 B)^2} \leq 0$. Thus the hyperbola has a negative slope and hence lies in the first and third quadrants around $(\frac{-A}{B}, \frac{-A}{B})$. Since $\frac{-A}{B} < 0$, the lower segment of the hyperbola (third quadrant) does not intersect the region R_I. Thus the threshold boundary in R_I is given by the upper segment of the hyperbola. This completes the proof. □

An illustration of the threshold boundary in R_I is provided in Fig. 2.3(b).

2.4 Threshold Scheduling Policy

We now proceed to use the structural results of the optimal scheduling policy, derived in the preceding section, to develop a threshold scheduling policy. We first consider the two-user broadcast and conjecture the following:

Conjecture 1. *The thresholdability property of the optimal scheduling policy reported in Proposition 3 and Proposition 4, in regions R_{II} and R_I, respectively, extends to the entire state space $[0,1]^2$.*

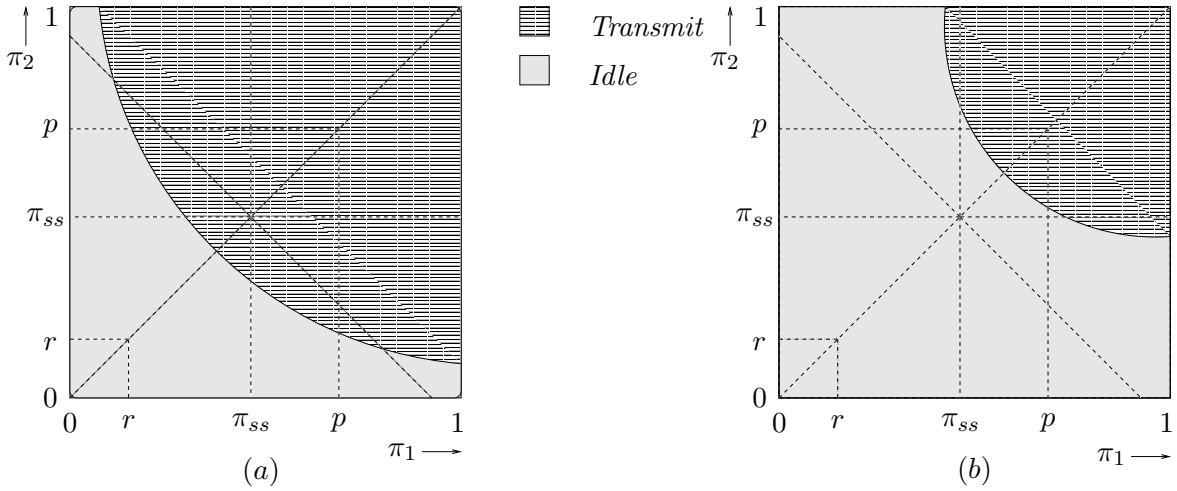

Figure 2.4: Illustration of the extrapolation of the threshold boundaries to the entire two-dimensional state space, when the broadcast is (a) Type I, (b) Type II.

Under Conjecture 1, the extrapolated threshold boundary spanning the entire state space is given by Corollary 1 and Corollary 2 for Type I and Type II systems, respectively. Fig. 2.4 illustrates this extrapolation. It is worth comparing it to Fig. 2.3.

Now, for a given belief vector (π_1, π_2), with the system type identified and the value functions $V(\Pi(0)) \ldots V(\Pi(3))$ evaluated, the threshold policy is implemented by the following two steps:

- Evaluate

$$k^* \quad = \quad \arg\max_{k \in R} \begin{cases} V^a(k\pi_1, k\pi_2) = W + \beta V^a(T(k\pi_1), T(k\pi_2)), & \text{if Type I} \\ V^a(k\pi_1, k\pi_2) = \frac{W}{1-\beta}, & \text{if Type II.} \end{cases}$$

  Note that, for both system types, the corresponding equations solved to evaluate k^* are two-dimensional polynomials. This can be verified by examining the expression for V^a in (eq 5).

- The threshold policy is given by

$$\text{Transmit, if } k^* \leq 1$$

$$\text{Idle, if } k^* > 1$$

Note that the first step identifies the point on the extrapolated threshold boundary along the direction of the belief vector (π_1, π_2). The $\max_{k \in R}$ follows from the result (Corollary 1 and Corollary 2) that the threshold boundary is given by the *upper segment* of the hyperbola. The second step determines the location of the belief vector with respect to the extrapolated threshold boundary.

We now extend, heuristically, the preceding threshold policy to the N-user broadcast. First, we generalize the two-user broadcast classification to the N-user case: The N-user broadcast is defined as Type I if $V^a(\pi_{ss}, \ldots, \pi_{ss}) \geq V^p(\pi_{ss}, \ldots, \pi_{ss})$ and Type II otherwise. This definition can be simplified with the following arguments: Denoting $(\pi_{ss} \ldots \pi_{ss})$ simply by π_{ss}, we have $V^p(\pi_{ss}) = W + \beta \max(V^p(\pi_{ss}), V^a(\pi_{ss}))$ since $T(\pi_{ss}) = \pi_{ss}$. Thus, $V^p(\pi_{ss}) \geq \frac{W}{1-\beta}$. Now, if $V^a(\pi_{ss}) < \frac{W}{1-\beta}$, then $V^a(\pi_{ss}) < V^p(\pi_{ss})$. Thus $\pi_{ss} \in \mathcal{P}$ and the system is Type II. Consider the case $V^a(\pi_{ss}) \geq \frac{W}{1-\beta}$. We have $V^a(\pi_{ss}) \geq W + \beta V^a(\pi_{ss})$. Note that $V^p(\pi_{ss}) = W + \beta V^a(\pi_{ss})$ (since, $V^p(\pi_{ss}) = W + \beta V^a(\pi_{ss})$ results in $V^p(\pi_{ss}) \geq \frac{W}{1-\beta}$, while $V^p(\pi_{ss}) = W + \beta V^p(\pi_{ss})$ results in $V^p(\pi_{ss}) = \frac{W}{1-\beta}$). From the preceding arguments, we have $V^a(\pi_{ss}) \geq V^p(\pi_{ss})$. Thus, $V^a(\pi_{ss}) \geq \frac{W}{1-\beta} \Rightarrow \pi_{ss} \in \mathcal{A}$ and the system is Type I. We summarize the simplified definition below:

$$\text{Type} \quad = \quad \begin{cases} I, & \text{if } V^a(\pi_{ss}) \geq \frac{W}{1-\beta} \\ II, & \text{if } V^a(\pi_{ss}) < \frac{W}{1-\beta}. \end{cases} \tag{2.26}$$

With $W \in (Nr, Np)$,[2] the threshold scheduling policy for the N-user broadcast is implemented in the following steps:

[2]Note that if $W \notin (Nr, Np)$ the optimal policy is greedy as proved in Section 2.3.1.

Step 0: Initialization

- *Evaluate the quantities $V(\Pi_0), \ldots, V(\Pi_{2^N-1})$:*
  Note that, by the inherent symmetry in the underlying Markov channel statistics of the users, we have the following property (P): For any permutation Ψ on belief vector x, we have $V(x) = V(\Psi(x))$. Thus the 2^N quantities, $V(\Pi_0), \ldots, V(\Pi_{2^N-1})$, can be obtained by evaluating only the following $N+1$ quantities: $\{V(\Pi_{(2^i-1)}\}$, $i \in \{0, 1, 2, \ldots, N\}$. Interpreting the infinite horizon discounted rewards as limits on the finite horizon rewards, as we did in the proof of Lemma 3, evaluate $V(\Pi_0) = \lim_{t\to\infty} V_t(\Pi_0)$, $\ldots$, $V(\Pi_{2^N-1}) = \lim_{t\to\infty} V_t(\Pi_{2^N-1})$, using an appropriate measure of convergence. The finite horizon reward $V_t(x)$ is given by the finite horizon Bellman equation [36]

$$V_t(x) \;=\; \max\Big(\sum_i x_i + \beta \sum_{j=0}^{2^N-1} P_j(x) V_{t-1}(\Pi_j), W + \beta V_{t-1}(T(x))\Big). \tag{2.27}$$

- *Identify the system type:*
  With $\pi_{ss} = (\pi_{ss}, \ldots, \pi_{ss})$, the system type is identified using the simplified rule (2.26), reproduced below:

$$\text{Type} \;=\; \begin{cases} I, & \text{if } V^a(\pi_{ss}) \geq \frac{W}{1-\beta} \\ II, & \text{if } V^a(\pi_{ss}) < \frac{W}{1-\beta}. \end{cases}$$

  where $V^a(\pi_{ss})$ is evaluated using (5), simplified using property (P) as

$$V^a(\pi_{ss}) \;=\; N\pi_{ss} + \beta \sum_{j=0}^{N} NC_j (1-\pi_{ss})^{(N-j)} \pi_{ss}^j V(\Pi(2^j-1)), \tag{2.28}$$

  with $V(\Pi(0)) \ldots V(\Pi(2^N-1))$ evaluated earlier.

Step 1: Threshold scheduling policy on belief vector $\pi = (\pi_1, \ldots, \pi_N)$

- If $\sum_i \pi_i > W$, *transmit* (follows from Proposition 2). Skip to Step 2
- Otherwise, evaluate k^* by solving a N-dimensional polynomial in k, as below:

$$k^* = \arg\max_{k \in R} \begin{cases} V^a(k\pi) = W + \beta V^a(T(k\pi)), & \text{if Type I} \\ V^a(k\pi) = \frac{W}{1-\beta}, & \text{if Type II} \end{cases} \tag{2.29}$$

  where

$$V^a(x) = \sum_i x_i + \beta \sum_{j=0}^{2^N-1} P_j(x) V(\Pi(j)), \tag{2.30}$$

  with $V(\Pi(0)) \ldots V(\Pi(2^N - 1))$ evaluated in Step 0

- The threshold policy is given by

$$\text{Transmit, if } k^* \leq 1$$
$$\text{Idle, if } k^* > 1$$

Step 2: State evolution

- If *transmit* decision was made, at the end of the slot, collect the 1-bit feedback, $f_1, \ldots, f_N$, from the broadcast users and update the belief values as below.

$$\pi_i \leftarrow \begin{cases} p, & \text{if } f_i = 1 \\ r, & \text{if } f_i = 0 \end{cases} \tag{2.31}$$

- If *idle* decision was made, update the belief vector as $\pi \leftarrow T(\pi)$
- Repeat Step 1 in the next slot.

Remark: The convexity of the threshold boundary renders optimality properties to the threshold policy in the following sense. Consider two belief vectors π^g and π^r such that $\sum \pi^g = \sum \pi^r$. Thus, if the broadcast state is in any of these two states, the immediate rewards upon *transmit* decisions are the same. Indeed, the

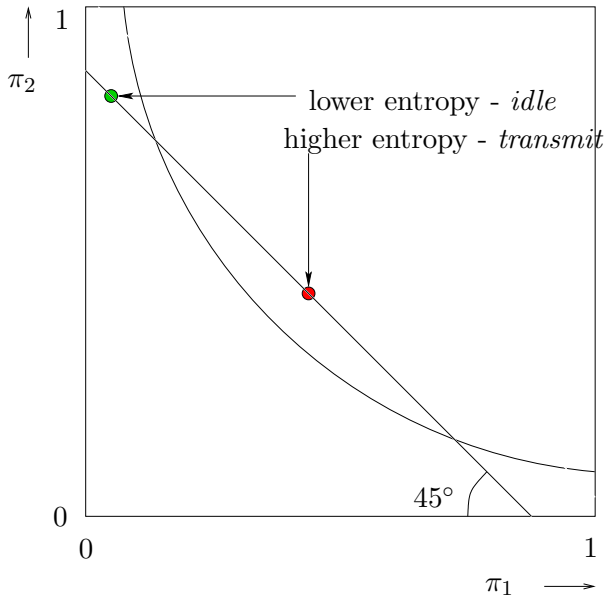

Figure 2.5: Illustration of the connection between the threshold scheduling decision and the entropy of the broadcast system state.

immediate rewards upon *idle* decisions are equal to W in both states. Let the entropy of the broadcast in state π^g be lower than when the broadcast is in state π^r, i.e., $\sum_i H(\pi^g(i)) < \sum_i H(\pi^g(i))$, where $H(x)$ indicates the entropy [37] of a channel with belief x. It is intuitive to see that if it is optimal to *transmit* at state π^g, then it is optimal to *transmit* at state π^r. This is because, with the 'exploitation' end of the trade-off equalized between π^g and π^r (since $\sum \pi^g = \sum \pi^r$), the exploration end of the tradeoff is more pronounced in π^r due to its higher entropy, essentially making it optimal to *transmit*, i.e., explore at π^r, if it is optimal to explore at π^g. Since the threshold boundary in the proposed threshold policy is convex, if the threshold decision at π^g is to *transmit*, then the threshold decision at π^r is also to *transmit*. An illustration of π^g, π^r along with the convex threshold boundary is provided in

Fig. 2.5. Thus the threshold policy, thanks to the optimality framework in which it is derived, exhibits an implementation structure similar to that of the optimal policy.

2.5 Numerical Results and Discussion

We now proceed to illustrate, via numerical experiments, that the proposed threshold policy has near-optimal performance. We first study the finite horizon performance of the proposed policy in Fig. 2.6. The discounted reward of the proposed policy over a finite horizon m (denoted by $V_{\text{policy}}(m)$) is plotted alongside the total discounted reward corresponding to the optimal policy[3] over horizon m (denoted by $V(m)$). Note that $V_{\text{policy}}(m)$ is indistinguishable from $V(m)$. Recall that the threshold policy was derived by extrapolating the structural properties of the optimal policy from the two-user broadcast to the general N-user broadcast. The superior performance of the threshold policy, indicated by Fig. 2.6 (and subsequent numerical results), suggests that the structural properties indeed can be generalized and hence justifies our approach.

In the rest of this analysis, we will focus on the infinite horizon performance of the proposed policy, compared with various system level performance limits. Note that the infinite horizon reward can be approximated by evaluating finite horizon rewards over a 'sufficiently' large horizon. From exhaustive simulations, we observed that the reward functions achieve reasonable convergence around $m = 7$ (also seen in Fig. 2.6). We therefore approximate the infinite horizon rewards by the rewards evaluated at $m = 7$ in the rest of this analysis. In Table 2.1, we report the % suboptimality of the proposed policy. The quantity $\%\text{subopt}:=\frac{V-V_{\text{policy}}}{V} \times 100\%$ quantifies the degree of

[3]The optimal policy is implemented using the finite horizon Bellman equation in (38)

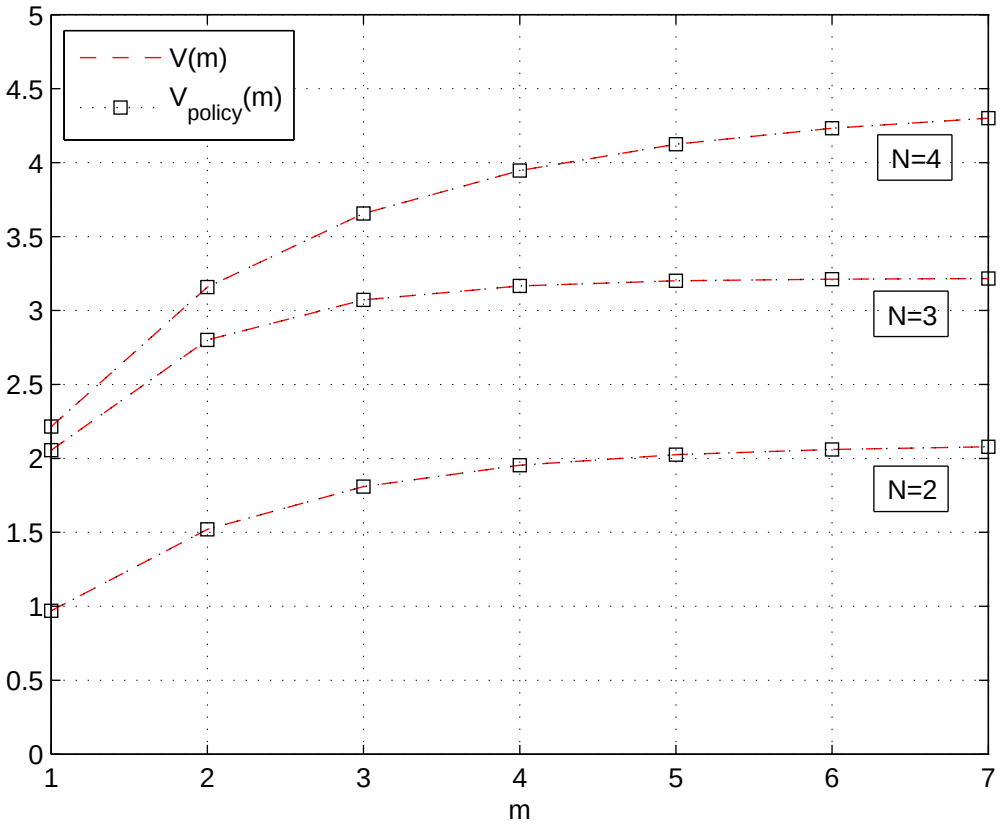

Figure 2.6: Finite horizon values of $V(m)$, $V_{\text{policy}}(m)$ for various number of broadcast users.

W	p	r	β	V	V_{policy}	%subopt
3.2323	0.8147	0.7380	0.2762	4.4651	4.4651	0 %
3.8261	0.9575	0.9239	0.2946	5.4227	5.4227	0 %
1.6813	0.4218	0.3862	0.6753	5.6400	5.6399	0.0018 %
0.9212	0.2769	0.0128	0.2583	2.3176	2.3176	0 %
1.5327	0.4387	0.1674	0.6593	4.4950	4.4950	0 %
2.2140	0.6491	0.4750	0.5886	5.3344	5.3253	0.1706 %
1.9074	0.6868	0.1260	0.4211	4.0446	4.0446	0 %
1.1852	0.4868	0.2122	0.4681	3.8936	3.8935	0.0026 %
1.8291	0.6443	0.2439	0.6869	6.5860	6.5726	0.2035 %
1.7714	0.6225	0.3654	0.3246	2.7002	2.7002	0 %

Table 2.1: Illustration of the near optimal performance of the proposed threshold policy. Total reward values are truncated to four decimal places. Each row corresponds to a fixed set of randomly generated system parameters and initial belief values. Number of broadcast users = 4.

suboptimality of the proposed policy. Each row in Table 2.1 corresponds to randomly generated system parameters with $N = 4$. The near optimal performance of the proposed policy is once again evident from Table 2.1.

In Table 2.2, we study the gains achieved by using 1-bit feedback from the users. The quantity V_{genie} corresponds to the total discounted reward under optimal scheduling in the *genie-aided* system defined as follows: at the end of each slot — independent of whether a transmit or idle decision was made in that slot — the scheduler learns about the channel states of all the users in that slot. The quantity V_{nofb} is the total discounted reward when the scheduler rejects the feedback information from the scheduled users and schedules solely based on the knowledge of the system level parameters, i.e., N, W, β and the statistics of the Markov channels, i.e., p and r. Thus

N	W	p	r	β	V_{genie}	V_{policy}	V_{nofb}	%fbgain
2	0.8139	0.4456	0.2880	0.6256	2.2523	2.2446	2.0923	95.1679 %
2	0.4801	0.2630	0.1720	0.6135	1.2585	1.2570	1.2017	97.4352 %
2	0.7949	0.6477	0.2921	0.5282	1.9907	1.9788	1.8996	86.9424 %
3	1.5819	0.5469	0.5236	0.7789	6.8312	6.7480	6.0094	89.8665 %
3	2.2272	0.8003	0.1135	0.4531	4.2901	4.2875	4.0562	98.8572 %
3	1.3724	0.5085	0.2597	0.6906	4.5968	4.5653	4.1031	93.6145 %
4	1.8299	0.5085	0.2597	0.6906	6.0284	6.0083	5.4709	96.3937 %
4	2.3315	0.7513	0.1916	0.5036	4.9462	4.9347	4.6579	96.0039 %
4	0.7165	0.4709	0.1085	0.7066	2.8552	2.8015	2.2272	91.4621 %
5	1.1834	0.6948	0.2203	0.7701	8.4060	8.4045	7.6546	99.8030 %
5	2.0542	0.4898	0.2182	0.5878	5.3549	5.3510	4.8626	99.2000 %
5	0.3981	0.1190	0.0593	0.7758	3.4376	3.3988	1.4755	98.0243 %

Table 2.2: Illustration of the gain associated with 1-bit feedback. Each row corresponds to a fixed set of randomly generated system parameters and initial belief values. Reward values are truncated to four decimal places.

with horizon $m = 7$, $V_{\text{nofb}} = \max\{W, N\pi_{ss}\}\frac{1-\beta^7}{1-\beta}$, where the steady state probability of the Markov channels, $\pi_{ss} = \frac{r}{1-(p-r)}$. The gain corresponding to the 1-bit feedback from each user, at the end of slots when *transmit* decision was made, is now quantified by the quantity %fbgain$=\frac{V_{\text{policy}}-V_{\text{nofb}}}{V_{\text{genie}}-V_{\text{nofb}}} \times 100\%$. The high value of %fbgain reported in Table 2.2, for various randomly generated system parameters, underlines the significance of using the 1-bit feedback as well as the near-optimal performance of the proposed policy.

In Fig. 2.7(a), for increasing values of the discount factor, β, we plot the optimal discounted reward alongside V_{policy} and V_{rand} - the total discounted reward under random scheduling, i.e., in each slot the scheduler randomly decides to transmit or idle with equal probabilities, without any regard to the belief values on the user channels. Note that as β increases, the effect of the future discounted reward, and hence the

significance of the channel feedback, on the total discounted reward increases. Since the random policy throws away the channel feedback information, the gap between optimal reward V (similarly V_{policy}) and V_{rand} is expected to increase with increasing β, as observed in Fig. 2.7(a). In Fig. 2.7(b), we plot V, V_{policy} and V_{rand} for increasing system 'memory'. We define system memory as $(p - r)$. In the plot, the system memory is varied from 0 to 1 by assuming $r = 1 - p$ and varying p from 0.5 to 1. Similar to our discussion in Fig. 2.7(a), as memory increases, the significance of the channel feedback on the performance of a policy increases. Thus with increasing system memory the gap between V (similarly V_{policy}) and V_{rand} increases, as observed in Fig. 2.7(b).

Remark on Complexity: The proposed threshold policy is also computationally inexpensive to implement, having polynomial complexity in the number of broadcast users. Contrast this with the complexity of the optimal POMDP solutions: for finite horizon POMDPs, the optimal solution is, in general, PSPACE-hard to compute [38], whereas infinite horizon POMDPs are, in general, *undecidable* [39].

2.6 Summary

The 'exploitation vs exploration' trade-off vastly simplifies for special cases of broadcast parameters, with 'greedy type' policies turning out to be optimal. For the general broadcast, the trade-off is not as simple. We therefore approached the problem indirectly by first studying scheduling in the *two-user* broadcast. We established structural properties of the optimal policy in the two-user broadcast and, based on these structural properties, proposed a threshold scheduling policy for the general broadcast. Extensive numerical results suggest near-optimal performance of

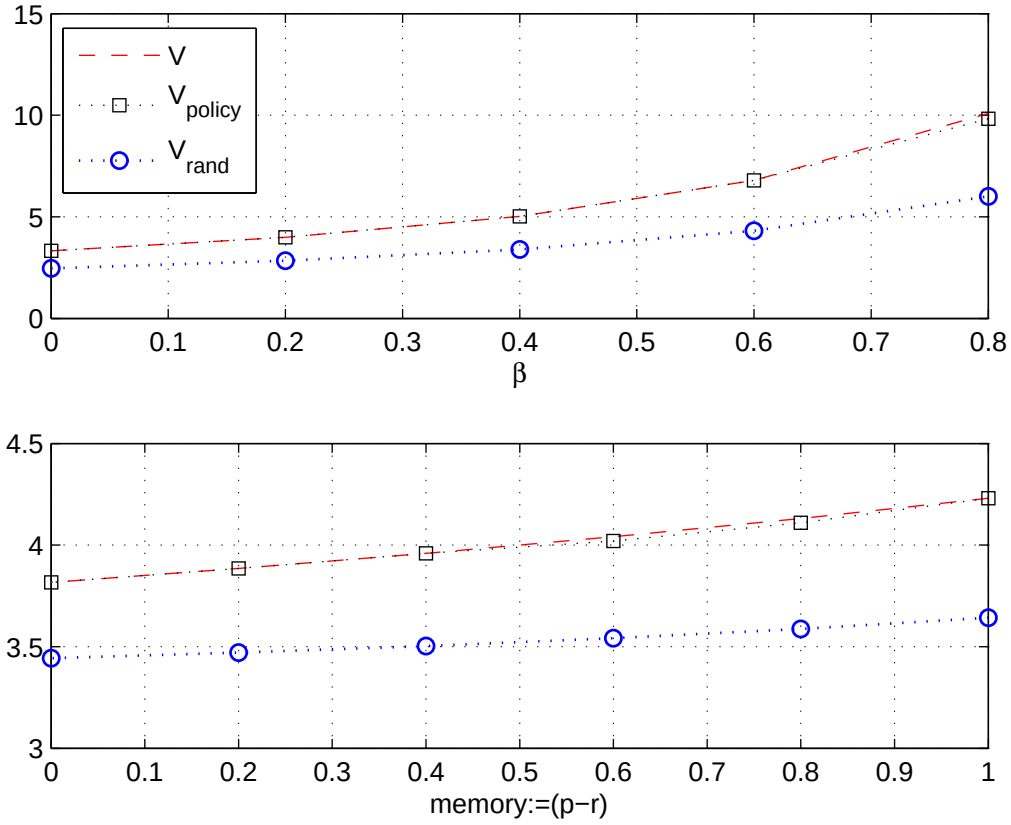

Figure 2.7: V, V_{policy} and V_{rand} versus (a) discount factor β, (b) system memory $(p-r)$. Same set of system parameters used within each subplot.

the proposed threshold policy. In addition, the proposed threshold policy is easy to implement, having complexity polynomial in the number of broadcast users. Numerical results further suggest that significant system level gains are associated with exploiting channel memory for opportunistic scheduling, even with minimal feedback (delayed, and obtained only during *transmit* slots).

CHAPTER 3

OPPORTUNISTIC SCHEDULING USING RANDOMLY DELAYED ARQ FEEDBACK IN CELLULAR DOWNLINK

3.1 Background

In the preceding chapter, we observed that, in a broadcast network with Markov modeled channels, opportunistic scheduling using 1-bit feedback from the users provides significant system level gains. While it is reasonable to assume that this 1-bit feedback is almost 'cost-free' in most networks and applications, this may not be the case in certain upcoming applications when both the forward and the reverse channels are equally in high demand. Fortunately, there is a strong consensus among the networking community [40] that the future wireless standards will increasingly support Automatic Repeat reQuest (ARQ) based error control (e.g., [41–44]) at the data link layer. From the scheduling point of view, the ARQ feedback is effectively the 1-bit feedback considered in the previous chapter, and is available free of cost.

In this chapter, we study joint channel estimation - opportunistic scheduling in a downlink system with an in-built ARQ feedback mechanism. We consider the general case when the ARQ feedback arrives at the scheduler with a *random delay* that is $i.i.d$ across users and time. It must be noted that a related work [45] studies opportunistic

spectrum access in a cognitive radio setting — a setup mathematically equivalent to our scheduling problem when the ARQ feedback is *instantaneous* — and showed that a simple greedy scheduling policy is optimal. In our setup, we consider the general problem when the ARQ feedback is randomly delayed. The delay in the feedback channel is an important consideration that cannot be overlooked in many realistic scenarios: one such instance being when the feedback signals suffer from significant propagation delay.

Despite the random delay, the ARQ feedback can be used for opportunistic scheduling to achieve significant performance gains. A sample of this gain is illustrated in Fig. 3.1 for a specific set of system parameters to be defined in the next section. Fig. 3.1 plots the sum (over all the downlink users) rate of successful transmission of packets over a length of m slots under optimal opportunistic scheduling when the scheduler has: (a) randomly delayed channel state information (CSI) from *all* the downlink users, (b) randomly delayed CSI from the scheduled user, i.e., randomly delayed ARQ feedback, and (c) no CSI, i.e., random scheduling. We make two observations from the figure: (1) The use of delayed ARQ feedback for opportunistic scheduling can achieve a performance close to opportunistic scheduling using delayed, perfect, CSI from all users, and (2) a 49% gain (when $m = 7$) in the sum rate is associated with opportunistic scheduling using delayed ARQ relative to random scheduling. These observations motivate our approach: exploit multiuser diversity in Markov-modeled downlink channels using the already existing (albeit delayed) ARQ feedback mechanisms. We describe the problem setup next.

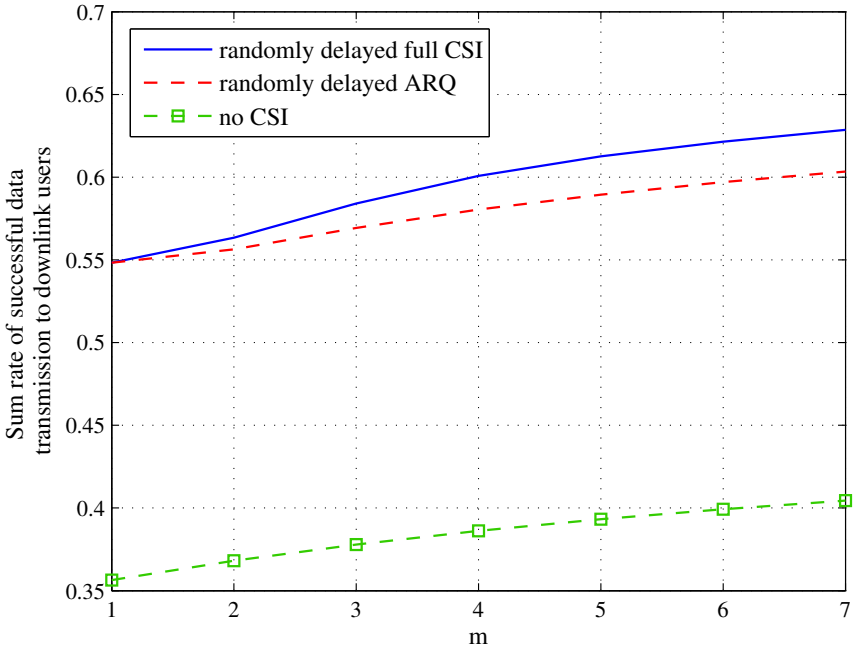

Figure 3.1: Illustration of the gains associated with opportunistic scheduling using randomly delayed ARQ feedback. System parameters used: $p = 0.8700$ $r = 0.1083$ $P_D(d = 0) = \frac{1}{3}$, $P_D(d = 1) = \frac{1}{3}$, $P_D(d = 2) = \frac{1}{3}$, $P_D(d > 2) = 0$, $\pi_m = [0.3358\ 0.1851\ 0.5483]$.

3.2 Problem Setup

3.2.1 Channel Model

We consider a downlink system with N users. For each user, there is an associated queue at the base station (henceforth the scheduler) that accumulates packets intended for that user. We assume that each queue is infinitely backlogged. As before, the channel between the scheduler and each user is modeled by an $i.i.d$ two-state Markov chain, where each state corresponds to the degree of decodability of the data sent through the channel. State 1 (ON) corresponds to full decodability, while state 0 (OFF) corresponds to zero decodability. Time is slotted and the channel of each user remains fixed for a slot and moves into another state in the next slot following the state transition probability of the Markov chain. The time slots of all users are synchronized. The two-state Markov channel is characterized by a 2×2 probability transition matrix

$$P = \begin{bmatrix} p & 1-p \\ r & 1-r \end{bmatrix}, \tag{3.1}$$

where the probabilities p and r retain their definitions from Chapter 2. Similar to Chapter 2, we assume positively correlated Markov channels, i.e., $p > r$.

3.2.2 Scheduling Problem

The scheduler is the central controller that controls the transmission to the users in each slot. In any time slot, the scheduler 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. At the beginning of a time slot, the head-of-line packet of the scheduled user is transmitted. The scheduled user attempts

to decode the received packet and based on the decodability of the packet sends back ACK(bit 1)/NACK(bit 0) feedback signals to the scheduler at the end of the time slot, over an error-free feedback channel. The feedback channel is assumed to suffer from a random delay that is *i.i.d* across users and time. This delayed feedback information, along with the label of the time slot from which it is acquired, will be used by the scheduler in scheduling decisions. The scheduler aims to maximize the sum of the rate of successful transmission of packets to all the users in the system. Note that the downlink scheduling problem directly fits into our general model, discussed in Chapter 1. We formally define the scheduling problem below.

3.2.3 Formal Problem Definition

Since the scheduler must make scheduling decisions based only on a partial observation[4] of the underlying Markov chain, the scheduling problem can be represented by a *Partially Observable Markov Decision Process (POMDP)*. We now formulate our problem in the language of POMDPs.

Horizon: We consider the finite horizon scenario. Time slots are indexed in decreasing order with slot 1 corresponding to the end of the horizon. Throughout this chapter, the horizon is denoted by m, i.e., the scheduling process begins at slot m.

Feedback arriving at slot t: For some slot t, $t \leq m$, let $n(t)$ be the number of ARQ feedback bits ($\{0,1\}$) arriving at the end of slot t from the users scheduled in the previous slots. Due to the random nature of the feedback delay, $n(t)$ can take values in the set $\{0,\ldots,m-t+1\}$. Let F_t represent all the ARQ feedback arriving at the end of slot t. Thus $F_t \in \{0,1\}^{n(t)}$, if $n(t) > 0$ and $F_t = \emptyset$, if $n(t) = 0$. The ARQ feedback is time-stamped and thus, since the scheduler has a record on which

[4]In this case, the set of time-stamped binary delayed feedback on the channels.

users were scheduled in the past slots, it can map the feedback bits F_t to the users and slots they originated from. Let f_k be the feedback that originated during slot k, where $k \leq m$. Note that since in each slot one and only one user is scheduled, f_k is neither empty nor has multiple values, i.e., $f_k \in \{0, 1\}$ with bit 0 mapped to NACK and bit 1 to ACK feedback.

Delay of feedback from user i in slot t: Let $D(i, t)$ be the random variable corresponding to the delay, in number of slots, experienced by the feedback sent by user i in slot t. Let $D(i, t) = 0$ correspond to the case when the ARQ feedback originating from user i in slot t arrives at the scheduler at the end of the same slot t. We assume the distribution of $D(i, t)$ to be *i.i.d* across users i and time t throughout this work, and let $P_D(d)$, $d \in \{0, 1, \ldots\}$ denote the probability mass function of D.

Belief value of user i in slot t - $\pi_t(i)$: This represents the probability that the channel of user $i \in \{1 \ldots N\}$, in slot t, is in the ON state, given all the past feedback about the channel. Define $T^u(.)$, for $u \in \{0, 1, \ldots\}$, as the u-step belief evolution operator given by $T^u(x) = T(T^{(u-1)}(x)) = T^{(u-1)}(T(x))$ with $T(x) = xp + (1 - x)r$ and $T^0(x) = x$ for $x \in [0, 1]$. Now if, at the end of slot $t + 1$, the arriving feedback F_{t+1} contains the ARQ feedback from user i from slot $k \in \{m, m - 1, \ldots, t + 1\}$, i.e., f_k, then, if k is the latest slot from which an ARQ feedback from user i has arrived, then $\pi_t(i)$ is obtained by applying the 1-step belief evolution operator repeatedly over all the time slots between 'now' (slot t) and slot k, i.e.,

$$\pi_t(i) = \begin{cases} T^{(k-t-1)}(p), & \text{if } f_k = 1 \\ T^{(k-t-1)}(r), & \text{if } f_k = 0. \end{cases} \tag{3.2}$$

If k is not the latest slot from which an ARQ feedback from user i has arrived (possible since the random nature of the feedback delay can result in out-of-turn arrival of ARQ feedback), then due to the first-order Markovian nature of the channels, this

ARQ feedback does not have any new information to affect the belief value, and so $\pi_t(i) = T(\pi_{t+1}(i))$. Similarly, if F_{t+1} does not contain any feedback from user i, then $\pi_t(i) = T(\pi_{t+1}(i))$.

Reward structure: In any slot t, a reward of 1 is accrued at the scheduler when the channel of the scheduled user is found to be in the ON state, else 0 is accrued.

Scheduling Policy $\mathfrak{A}_k$: A scheduling policy $\mathfrak{A}_k$ in slot k is a mapping from all the information available at the scheduler in slot k along with the slot index k to a scheduling decision a_k. Formally,

$$\begin{aligned} \mathfrak{A}_k : \quad & ([\pi_m, \pi_{m-1}, \ldots, \pi_k]^k, \{a_m, a_{m-1}, \ldots, a_{k+1}\}) \rightarrow a_k \\ & \forall k \in [1, m], \pi_k \in [0, 1]^N. \end{aligned} \tag{3.3}$$

where $\{a_m, a_{m-1}, \ldots, a_{k+1}\}$ are the past scheduling decisions and $[\pi_m, \pi_{m-1}, \ldots, \pi_k]^k$ are the belief values of the channels of all users, corresponding to slots $\{m, m-1, \ldots, k\}$, held by the scheduler *at the moment* (slot k).

Total expected reward in slot t, V_t: With the scheduling policy, $\{\mathfrak{A}_k\}_{k=1}^t$, fixed, the total expected reward in slot t, i.e., V_t, is the sum of the reward expected in the current slot t and the total reward expected in all the future slots $k < t$. Formally, with a_k denoting the scheduling decision in slot k,

$$\begin{aligned} & V_t([\pi_m, \pi_{m-1}, \ldots, \pi_t]^t, \{a_m, a_{m-1}, \ldots, a_{t+1}\}, \{\mathfrak{A}_k\}_{k=1}^t) \\ = \quad & R_t(\pi_t, a_t) + \mathrm{E}\left[V_{t-1}([\pi_m, \pi_{m-1}, \ldots, \pi_t, \pi_{t-1}]^{t-1}, \right. \\ & \left. \{a_m, a_{m-1}, \ldots, a_{t+1}, a_t\}, \{\mathfrak{A}_k\}_{k=1}^{t-1})\right], \end{aligned} \tag{3.4}$$

where $R_t(\pi_t, a_t)$ is the expected immediate reward and the expectation in the future reward is over the feedback received in slot t, i.e., F_t, along with the originating

slot indices. Note that the belief vector $[\pi_m, \pi_{m-1}, \ldots, \pi_t]^t$ is up-to-date based on all previous scheduling decisions and the ARQ feedback received before slot t. With the reward structure defined earlier, the expected immediate reward can be written as

$$R_t(\pi_t, a_t) \;\; = \;\; \pi_t(a_t).$$

Performance Metric: For a given scheduling policy $\{\mathfrak{A}_k\}_{k=1}^m$, the performance metric is given by the sum throughput (sum rate of successful transmission) over a finite horizon, m:

$$\eta_{\text{sum}}(m, \{\mathfrak{A}_k\}_{k=1}^m) \;\; = \;\; \frac{V_m(\pi_m, \{\mathfrak{A}_k\}_{k=1}^m)}{m}, \tag{3.5}$$

where π_m is the initial belief values of the channels.

3.3 Greedy Policy - Optimality, Performance Evaluation and the Implementation Structure

3.3.1 On the Optimality of the Greedy Policy

Consider the following policy:

$$\begin{aligned} \widehat{\mathfrak{A}}_k : \pi_k \rightarrow a_k \;\; &= \arg\max_i R_k(\pi_k, a_k = i) \\ &= \arg\max_i \pi_k(i) \quad \forall k \geq 1, \pi_k \in [0,1]^N. \end{aligned} \tag{3.6}$$

Since the above given policy attempts to maximize the expected immediate reward, without any regard to the expected future reward, it follows an approach that is fundamentally *greedy* in nature. We henceforth call $\{\widehat{\mathfrak{A}}_k\}_{k=1}^m$ the greedy policy and let $\hat{a}_k$ denote the scheduling decision in slot k under the greedy policy. We now proceed to establish the optimality of the greedy policy when $N = 2$. We first introduce the following lemma.

Lemma 5. *For any $u, v \in \{0, 1, 2, \ldots\}$ and any $x, y \in [0, 1]$ with $x \geq y$,*

$$\begin{aligned} T^u(p) &\geq T^{u+1}(x) \\ T^u(r) &\leq T^{u+1}(x) \\ T^u(x) &\geq T^u(y) \\ T^u(p) &\geq T^v(r). \end{aligned} \tag{3.7}$$

Proof. The proof is moved to Appendix B.1. □

The results of Lemma 5 can be explained intuitively. Note that $T^u(x)$ is the belief value of the channel (probability that the channel is in the ON-state) in the current slot given the belief value, u slots earlier, was x. Also note that $T^u(p)$ (similarly $T^u(r)$) gives the belief value in the current slot given the channel was in the ON state (similarly OFF state) $u+1$ slots earlier. Now, since the Markov channel is positively correlated ($p > r$), the probability that the channel is in the ON state in the current slot given it was in the ON state $u+1$ slots earlier ($T^u(p)$) is at least as high as the probability that the channel is ON in the current slot given it was *ON with probability* $x \in [0, 1]$, $u+1$ slots earlier ($T^{(u+1)}(x)$). This explains the first inequality in Lemma 5. The second and third inequalities can be explained along similar lines. Regarding the last inequality, consider slots t, k such that $t > k$. Due to the Markovian nature of the channel, the closer slot k is to t, the stronger is the memory, i.e., the dependency of the channel state in k with that of t. Now, since the channel is positively correlated, if the channel was in the ON state in slot t, the closer k is to t, the higher is the probability that the channel is ON in slot k. By definition, this probability is given by $T^u(p)$ with $u = t - k - 1$. Thus $T^u(p)$ monotonically decreases with u. Using a similar explanation, $T^u(r)$ monotonically *increases* with u. The limiting value of

both these functions, as $u \to \infty$, is the probability that the channel is ON when no information on the past channel states is available. This is given by the steady state probability[5]. This explains $T^u(p) \geq T^v(r)$ for any $u, v \in \{0, 1, \ldots\}$.

Proposition 5. *For $N = 2$, the sum throughput, $\eta_{sum}(m, \{\mathfrak{A}_k\}_{k=1}^m)$, of the system is maximized by the greedy policy $\{\widehat{\mathfrak{A}}_k\}_{k=1}^m$ for any ARQ delay distribution.*

Proof. Consider a slot $t < m$. Fix a sequence of scheduling decisions $\boldsymbol{a}_{t+1} := \{a_m, a_{m-1}, \ldots, a_{t+1}\}$. Recall the definition of F_{t+1}, the feedback arriving at the end of slot $t+1$, from Section 3.2.3. Let τ_{t+1} denote the originating slots corresponding to feedback F_{t+1}, i.e., if the feedback from users a_u and a_v, for $m \geq u > v \geq t+1$, both arrive at slot $t+1$, then $F_{t+1} = [f_u \ f_v]$ and $\tau_{t+1} = [u \ v]$. Also define $k_1 \in \{\emptyset, m, m-1, \ldots, t+1\}$ as the latest slot from which the ARQ feedback of user 1 is available at the scheduler by (the beginning of) slot t. Formally, if at least one ARQ feedback from user 1 has arrived at the scheduler by slot t, then

$$k_1 \ = \ \min_{k \in \{m, m-1, \ldots, t+1\} \text{ s.t } a_k = 1, \ f_k \text{ has arrived by slot } t} k. \tag{3.8}$$

If no ARQ feedback from user 1 has arrived by slot t, i.e., if $\nexists$ a k such that '$k \in \{m, m-1, \ldots, t+1\}$ s.t $a_k = 1$, f_k has arrived by slot t', then $k_1 = \emptyset$. Let $l_1 = k_1 - t - 1$, when $k_1 \neq \emptyset$, be a measure of 'freshness' of the latest feedback from user 1. Let $l_1 = \emptyset$ when $k_1 = \emptyset$. Similarly define k_2, l_2 for user 2. With these definitions, the proof proceeds in two steps: In step 1, we show that the greedy decision in slot t, given the ARQ feedback and the scheduling decision from slot $\min(k_1, k_2)$, is independent of the feedback and scheduling decision corresponding to slot $\max(k_1, k_2)$. In step 2, we

[5]We will discuss the steady state probability in Section 3.4.

show that, if the greedy policy is implemented in slot t, then the expected immediate reward in slot t is independent of the scheduling decisions $\boldsymbol{a}_{t+1}$. We then provide induction based arguments to establish the proposition.

Step 1: Let $\boldsymbol{F}_{t+1} := \{F_m, F_{m-1}, \ldots, F_{t+1}\}$ and $\boldsymbol{\tau}_{t+1} := \{\tau_m, \tau_{m-1}, \ldots, \tau_{t+1}\}$. The greedy decision in slot t, conditioned on the past feedback and scheduling decisions is given by

$$\hat{a}_t|_{\boldsymbol{F}_{t+1},\boldsymbol{\tau}_{t+1},\boldsymbol{a}_{t+1},\pi_m} \quad = \quad \hat{a}_t|_{f_{k_1},f_{k_2},l_1,l_2,\boldsymbol{a}_{t+1},\pi_m}. \tag{3.9}$$

The preceding equation comes directly from the first order Markovian property of the underlying channels. Consider the case when $k_1 < k_2 \leq m$ ($\Rightarrow l_1 < l_2$) or $k_1 = k_2 = \emptyset$ ($\Rightarrow l_1 = l_2 = \emptyset$). The belief values in slot t as a function of feedback f_{k_1} and f_{k_2} is given below:

$$\begin{aligned}
&(\pi_t(1), \pi_t(2)) \\
&= \begin{cases}
(T^{l_1}(p), T^{l_2}(p)), & \text{if } f_{k_1} = 1, f_{k_2} = 1 \\
(T^{l_1}(p), T^{l_2}(r)), & \text{if } f_{k_1} = 1, f_{k_2} = 0 \\
(T^{l_1}(p), T^{(m-t)}(\pi_m(2))), & \text{if } f_{k_1} = 1, k_2 = \emptyset \\
(T^{l_1}(r), T^{l_2}(p)), & \text{if } f_{k_1} = 0, f_{k_2} = 1 \\
(T^{l_1}(r), T^{l_2}(r)), & \text{if } f_{k_1} = 0, f_{k_2} = 0 \\
(T^{l_1}(r), T^{(m-t)}(\pi_m(2))), & \text{if } f_{k_1} = 0, k_2 = \emptyset \\
(T^{(m-t)}(\pi_m(1)), T^{(m-t)}(\pi_m(2))), & \text{if } k_1 = \emptyset, k_2 = \emptyset
\end{cases}
\end{aligned} \tag{3.10}$$

Using Lemma 5, the greedy decision can be written as

$$\begin{aligned}
&\hat{a}_t|_{f_{k_1},f_{k_2},l_1,l_2,\boldsymbol{a}_{t+1},\pi_m} \\
&= \begin{cases}
1, & \text{if } f_{k_1} = 1 \\
2, & \text{if } f_{k_1} = 0 \\
\arg\max_{i\in\{1,2\}}(\pi_m(i)), & \text{if } k_1 = \emptyset, k_2 = \emptyset.
\end{cases}
\end{aligned} \tag{3.11}$$

Thus the greedy decision is independent of feedback f_{k_2} if $k_1 < k_2$. We now proceed to generalize equation (3.11). Let k^* denote the latest slot for which an ARQ feedback is available from *one of the users* by slot t, i.e.,

$$k^* = \begin{cases} \min\{k_1, k_2\}, & \text{if } k_1 \neq \emptyset, k_2 \neq \emptyset \\ k_1, & \text{if } k_1 \neq \emptyset, k_2 = \emptyset \\ k_2, & \text{if } k_1 = \emptyset, k_2 \neq \emptyset \\ \emptyset, & \text{if } k_1 = \emptyset, k_2 = \emptyset. \end{cases} \tag{3.12}$$

Let $l = k^* - t - 1$ for $k^* \neq \emptyset$ and $l = \emptyset$ for $k^* = \emptyset$ be a measure of freshness of the latest ARQ feedback. Thus, using the preceding discussion, we have

$$\begin{aligned} & \hat{a}_t|_{f_{k_1}, f_{k_2}, l_1, l_2, \boldsymbol{a}_{t+1}, \pi_m} \\ &= \hat{a}_t|_{f_{k^*}, l, \boldsymbol{a}_{t+1}, \pi_m} \\ &= \begin{cases} a_{k^*}, & \text{if } k^* \neq \emptyset, f_{k^*} = 1 \\ \bar{a}_{k^*}, & \text{if } k^* \neq \emptyset, f_{k^*} = 0 \\ \arg\max_{i \in \{1,2\}}(\pi_m(i)), & \text{if } k^* = \emptyset \end{cases} \end{aligned} \tag{3.13}$$

where $\bar{a}_{k^*}$ is the user *not* scheduled in slot k^*. This completes step 1 of the proof.

Step 2: If the greedy policy is implemented in slot t, the immediate reward expected in slot t, conditioned on scheduling decisions $\boldsymbol{a}_{t+1}$ and initial belief π_m can be rewritten as

$$\begin{aligned} & \mathrm{E}_{\pi_t|\boldsymbol{a}_{t+1}, \pi_m} R_t(\pi_t, \hat{a}_t) \\ &= \mathrm{E}_{\pi_t|l=\emptyset, \boldsymbol{a}_{t+1}, \pi_m}(R_t(\pi_t, \hat{a}_t)) P(l = \emptyset | \boldsymbol{a}_{t+1}, \pi_m) \\ &\quad + \mathrm{E}_{l, l\neq\emptyset|\boldsymbol{a}_{t+1}, \pi_m} \mathrm{E}_{\pi_t|l, l\neq\emptyset, \boldsymbol{a}_{t+1}, \pi_m}(R_t(\pi_t, \hat{a}_t)), \end{aligned} \tag{3.14}$$

where l is defined after (3.12). Note that

$$\mathrm{E}_{\pi_t|l=\emptyset, \boldsymbol{a}_{t+1}, \pi_m}(R_t(\pi_t, \hat{a}_t)) = \max_i T^{(m-t)}(\pi_m(i)) \tag{3.15}$$

since, with $l = \emptyset$, i.e., no past feedback at the scheduler, the belief values at slot t is independent of the past scheduling decisions and is simply given by $\pi_t = T^{(m-t)}(\pi_m)$. Now rewriting the second part of (3.14),

$$\begin{aligned}
&\mathrm{E}_{l,l\neq\emptyset|\boldsymbol{a}_{t+1},\pi_m}\,\mathrm{E}_{\pi_t|l,l\neq\emptyset,\boldsymbol{a}_{t+1},\pi_m}(R_t(\pi_t,\hat{a}_t)) \\
&\quad = \mathrm{E}_{l,l\neq\emptyset|\boldsymbol{a}_{t+1},\pi_m}\,\mathrm{E}_{\pi_{l+t+1}|l,l\neq\emptyset,\boldsymbol{a}_{t+1},\pi_m}\,\mathrm{E}_{\pi_t|\pi_{l+t+1},l,l\neq\emptyset,\boldsymbol{a}_{t+1},\pi_m}(R_t(\pi_t,\hat{a}_t)). \qquad (3.16)
\end{aligned}$$

Consider $\mathrm{E}_{\pi_t|\pi_{l+t+1},l,l\neq\emptyset,\boldsymbol{a}_{t+1},\pi_m}(R_t(\pi_t,\hat{a}_t))$. From the first step of the proof, the greedy decision in slot t can be made solely based on the latest feedback, i.e., $f_{k^*=l+t+1}$. This was recorded in (3.13). Thus, if the feedback f_{k^*} is an ACK (occurs with probability $\pi_{l+t+1}(a_{l+t+1})$) reschedule the user a_{l+t+1} in slot t. Conditioned on $f_{k^*} = 1$, the belief value $\pi_t(a_{l+t+1})$ and hence the expected immediate reward in slot t is given by $T^l(p)$. If the feedback is a NACK, schedule the other user denoted by $\bar{a}_{l+t+1}$. Conditioned on $f_{k^*} = 0$, the belief value $\pi_t(\bar{a}_{l+t+1})$ and hence the expected immediate reward in slot t is given by $T^{(l+1)}(\pi_{l+t+1}(\bar{a}_{l+t+1})) = \pi_{l+t+1}(\bar{a}_{l+t+1})T^l(p) + (1 - \pi_{l+t+1}(\bar{a}_{l+t+1}))T^l(r)$. Averaging over $f_{k^*=l+t+1}$, we have

$$\begin{aligned}
&\mathrm{E}_{\pi_t|\pi_{l+t+1},l,l\neq\emptyset,\boldsymbol{a}_{t+1},\pi_m}(R_t(\pi_t,\hat{a}_t)) \\
&\quad = \pi_{l+t+1}(a_{l+t+1})T^l(p) + (1 - \pi_{l+t+1}(a_{l+t+1})) \times \\
&\qquad \Big(\pi_{l+t+1}(\bar{a}_{l+t+1})T^l(p) + (1 - \pi_{l+t+1}(\bar{a}_{l+t+1}))T^l(r)\Big) \\
&\quad = P\big(\{S_{l+t+1}(1) = 1 \cup S_{l+t+1}(2) = 1\}| \\
&\qquad\qquad \pi_{l+t+1}, l, l \neq \emptyset, \boldsymbol{a}_{t+1}, \pi_m\big)T^l(p) \\
&\qquad + P\big(\{S_{l+t+1}(1) = 0 \cap S_{l+t+1}(2) = 0\}| \\
&\qquad\qquad \pi_{l+t+1}, l, l \neq \emptyset, \boldsymbol{a}_{t+1}, \pi_m\big)T^l(r)
\end{aligned} \qquad (3.17)$$

where $S_k(i)$ is the 1/0 state of the channel of user i in slot k. From (3.16),

$$
\begin{aligned}
&\mathrm{E}_{l,l\neq\emptyset|\boldsymbol{a}_{t+1},\pi_m}\,\mathrm{E}_{\pi_t|l,l\neq\emptyset,\boldsymbol{a}_{t+1},\pi_m}(R_t(\pi_t,\hat{a}_t)) \\
&= \mathrm{E}_{l,l\neq\emptyset|\boldsymbol{a}_{t+1},\pi_m}\,\mathrm{E}_{\pi_{l+t+1}|l,l\neq\emptyset,\boldsymbol{a}_{t+1},\pi_m} \\
&\quad \Big(P\big(\{S_{l+t+1}(1)=1\cup S_{l+t+1}(2)=1\}|\pi_{l+t+1},l,l\neq\emptyset,\boldsymbol{a}_{t+1},\pi_m\big)T^l(p) \\
&\quad +P\big(\{S_{l+t+1}(1)=0\cap S_{l+t+1}(2)=0\}|\pi_{l+t+1},l,l\neq\emptyset,\boldsymbol{a}_{t+1},\pi_m\big)T^l(r)\Big) \\
&= \mathrm{E}_{l,l\neq\emptyset|\boldsymbol{a}_{t+1},\pi_m}\Big(P\big(\{S_{l+t+1}(1)=1\cup S_{l+t+1}(2)=1\}|l,l\neq\emptyset,\boldsymbol{a}_{t+1},\pi_m\big)T^l(p) \\
&\qquad\qquad +P\big(\{S_{l+t+1}(1)=0\cap S_{l+t+1}(2)=0\}|l,l\neq\emptyset,\boldsymbol{a}_{t+1},\pi_m\big)T^l(r)\Big) \\
&= \mathrm{E}_{l,l\neq\emptyset|\boldsymbol{a}_{t+1},\pi_m}\Big(P\big(\{S_{l+t+1}(1)=1\cup S_{l+t+1}(2)=1\}|\pi_m\big)T^l(p) \\
&\quad +P\big(\{S_{l+t+1}(1)=0\cap S_{l+t+1}(2)=0\}|\pi_m\big)T^l(r)\Big) \qquad (3.18)
\end{aligned}
$$

We have used the following argument in the last equality: the event $(\{S_{l+t+1}(1)=1\cup S_{l+t+1}(2)=1\})$ is controlled by the underlying Markov dynamics and is independent of the scheduling decisions $\boldsymbol{a}_{t+1}$. Likewise, this event is independent of the value of l since we have assumed that the feedback channel and the forward channel are independent.

Recall $D(i,k)$ is the random variable indicating the delay incurred by the ARQ feedback sent by user i in slot k. Let L be the random variable corresponding to the quantity l, the degree of freshness of the latest ARQ feedback, and $P_L(.)$ be the probability mass function of L.

Therefore, for $0 \leq l \leq m-t-1$,

$$\begin{aligned}
& P_L(l|\boldsymbol{a}_{t+1}, \pi_m) \\
&= P\big(\{D(a_{l+t+1}, l+t+1) \leq l, D(a_{l+t}, l+t) > (l-1), \\
&\qquad D(a_{l+t-1}, l+t-1) > (l-2)), \ldots, \\
&\qquad\qquad D(a_{t+1}, t+1) > 0\}|\boldsymbol{a}_{t+1}, \pi_m\big) \\
&= P\big(\{D(a_{l+t+1}, l+t+1) \leq l, D(a_{l+t}, l+t) > (l-1), \\
&\qquad D(a_{l+t-1}, l+t-1) > (l-2)), \ldots, \\
&\qquad\qquad D(a_{t+1}, t+1) > 0\}|\boldsymbol{a}_{t+1}\big) \\
&= P(D(1, l+t+1) \leq l) \prod_{k=t+l}^{t+1} P(D(1,k) > k-t-1)
\end{aligned} \tag{3.19}$$

where we have used the independence between the forward and the feedback channel to remove the condition on π_m in the second equality. The last equality comes from the assumption that the ARQ delay is $i.i.d$ across users and time[6]. Similarly

$$P_L(l = \emptyset|\boldsymbol{a}_{t+1}, \pi_m) = \prod_{k=m}^{t+1} P(D(a_k, k) > k-t-1) \tag{3.20}$$

[6]Note: here we do not require the ARQ delay to be identically distributed across time.

Applying the preceding equations in (3.14), we have

$$\begin{aligned}
&\mathrm{E}_{\pi_t|\boldsymbol{a}_{t+1},\pi_m} R_t(\pi_t, \hat{a}_t) \\
&= \prod_{k=m}^{t+1} P(D(a_k, k) > k - t - 1) \max_i T^{(m-t)}(\pi_m(i)) \\
&\quad + \sum_{l=0}^{m-t-1} P(D(1, l+t+1) \leq l) \prod_{k=t+l}^{t+1} P(D(1,k) > k - t - 1) \\
&\qquad \Big(P\big(\{S_{l+t+1}(1) = 1 \cup S_{l+t+1}(2) = 1\}|\pi_m\big) T^l(p) \\
&\qquad\quad + P\big(\{S_{l+t+1}(1) = 0 \cap S_{l+t+1}(2) = 0\}|\pi_m\big) T^l(r) \Big)
\end{aligned} \tag{3.21}$$

The expected reward in slot t is thus independent of the actions $\{a_m, a_{m-1} \ldots a_{t+1}\}$ if the greedy policy is implemented in slot t. By extension, the total reward expected from slot t until the horizon is independent of the scheduling vector $\boldsymbol{a}_{t+1}$ if the greedy policy is implemented in slots $\{t, t-1, \ldots, 1\}$, i.e.,

$$\sum_{k=t}^{1} \mathrm{E}_{\pi_k|\boldsymbol{a}_{t+1},\pi_m} R_k(\pi_k, \hat{a}_k) \;\; = \;\; \sum_{k=t}^{1} \mathrm{E}_{\pi_k|\pi_m} R_k(\pi_k, \hat{a}_k). \tag{3.22}$$

Thus, if the greedy policy is optimal in slots $\{t, t-1, \ldots, 1\}$, then, it is also optimal in slot $t+1$. Since t is arbitrary and since the greedy policy is optimal at the horizon, by induction, the greedy policy is optimal in every slot $\{m, m-1, \ldots, 1\}$. This establishes the proposition. □

Remarks: From the discussion following (3.19), the ARQ delay need not be identically distributed across time for the preceding proof to hold. Thus, the greedy policy is optimal for $N = 2$ even when the ARQ delay distribution is time-variant. Also, since m is arbitrary, the greedy policy maximizes the sum throughput over an infinite horizon. We record this below.

Corollary 3. *For $N = 2$, the greedy policy is optimal when the performance metric is the sum throughput over an infinite horizon, i.e.,*

$$\{\hat{\mathfrak{A}}_k\}_{k\geq 1} \;=\; \arg \max_{\{\mathfrak{A}_k\}_{k\geq 1}} \lim_{m\to\infty} \frac{V_m(\pi, \{\mathfrak{A}_k\}_{k\geq 1})}{m} \tag{3.23}$$

for any initial belief π.

The optimality of the greedy policy does not extend to the case $N > 2$. We record this in the following proposition.

Proposition 6. *The greedy policy is not, in general, optimal when there are more than two users in the downlink.*

Proof outline: We establish the proposition using an analytic counterexample. For $N = 3$, horizon $m = 4$ and deterministic ARQ delay of $D = 1$, i.e., $P_D(d = 1) = 1$, we explicitly evaluate the total expected reward, $V_m(\pi_m, \{\hat{\mathfrak{A}}_k\}_{k=1}^m)$, corresponding to the greedy policy $\hat{\mathfrak{A}}_k$ and the total expected reward, $V_m(\pi_m, \{\tilde{\mathfrak{A}}_k\}_{k=1}^m)$, corresponding to an arbitrarily defined policy $\tilde{\mathfrak{A}}_k$, as a function of the system parameters $p, r, \pi_m|_{m=4}$. For specific sets of system parameters we show that the total expected reward corresponding to the greedy policy is less than that of the policy $\tilde{\mathfrak{A}}_k$, thus establishing the sub-optimality of the greedy policy. A formal proof can be found in Appendix B.2.

It is interesting to contrast this result with that in [45]. Here, the authors showed that, with the ARQ feedback being instantaneous, i.e., end-of-slot, the greedy policy is optimal for any number of users. We have now shown that, when the ARQ is randomly delayed, the greedy policy is optimal when $N = 2$ and not, in general, optimal when $N > 2$. Thus, minimal generalization from both the systems disturbs the optimality properties of the greedy policy, as illustrated in Fig. 3.2. This essentially

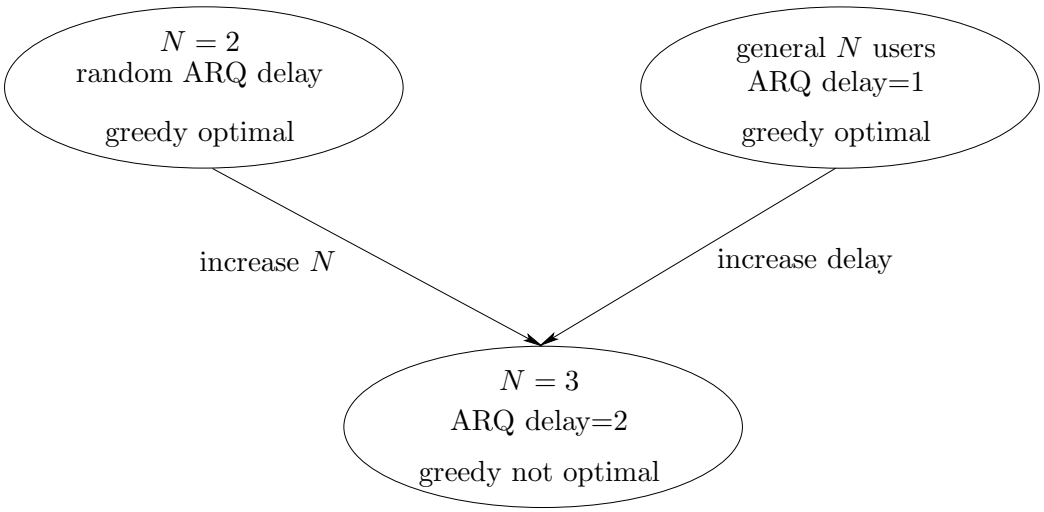

Figure 3.2: Illustration showing the volatility of the greedy policy optimality.

underlines the volatile dynamics of optimal scheduling in networks with Markov-modeled channels.

Despite the suboptimality of the greedy policy, numerical studies suggest that the greedy policy has near-optimal performance. We discuss this in the next subsection.

3.3.2 Performance Evaluation of the Greedy Policy

Table 3.1 and Table 3.2 provide a comparison of the total expected reward over a horizon m under the greedy policy, i.e., $(V_m(\pi_m, \text{greedy}))$ and the reward under the optimal policy, i.e., $(V_m(\pi_m, \text{opt}))$. The optimal reward is evaluated by a brute-force search over the scheduling decisions in every slot $t \in \{m, m-1, \ldots, 1\}$. The quantity $\%\text{subopt}= \frac{V_m(\pi_m,\text{opt})-V_m(\pi_m,\text{greedy})}{V_m(\pi_m,\text{opt})} \times 100\%$ captures the degree of suboptimality of the greedy policy. The system parameters p, r, π_m are (uniform) randomly generated. Fixing a value for the maximum ARQ delay, the delay probability mass function is constructed as follows: for each value of d, $P_D(d)$ is generated randomly (uniform over

$p = 0.7965,\ r = 0.1365,\ N = 3$ $\pi_m = [0.1351\ 0.2523\ 0.2410]$ $P_D(2) = 1, P_D(d \neq 2) = 0$			
m	$V_m(\pi_m, \text{opt})$	$V_m(\pi_m, \text{greedy})$	%subopt
1	0.2523	0.2523	0 %
2	0.5553	0.5553	0 %
3	0.8918	0.8918	0 %
4	1.3022	1.3022	0 %
5	1.7405	1.7345	0.3500 %
6	2.2011	2.1805	0.9400 %
7	2.6728	2.6553	0.6500 %

$p = 0.9172,\ r = 0.2858,\ N = 3$ $\pi_m = [0.7572\ 0.7537\ 0.3804]$ $P_D(0) = 0.8822, P_D(1) = 0.1178$ $P_D(d > 1) = 0$			
m	$V_m(\pi_m, \text{opt})$	$V_m(\pi_m, \text{greedy})$	%subopt
1	0.7572	0.7572	0 %
2	1.6230	1.6230	0 %
3	2.5074	2.5067	0.0261 %
4	3.3957	3.3948	0.0263 %
5	4.2861	4.2851	0.0235 %
6	5.1780	5.1769	0.0207 %
7	6.0707	6.0696	0.0182 %

$p = 0.6619,\ r = 0.2389,\ N = 3$ $\pi_m = [0.7678\ 0.1459\ 0.7698]$ $P_D(0) = 0.5908, P_D(1) = 0.3959$ $P_D(2) = 0.0132, P_D(d > 2) = 0$			
m	$V_m(\pi_m, \text{opt})$	$V_m(\pi_m, \text{greedy})$	%subopt
1	0.7698	0.7698	0 %
2	1.3785	1.3785	0 %
3	1.9315	1.9312	0.0155 %
4	2.4584	2.4573	0.0447 %
5	2.9735	2.9720	0.0504 %
6	3.4843	3.4825	0.0517 %
7	3.9933	3.9914	0.0476 %

Table 3.1: Illustration of the near optimal performance of the greedy policy. Each table corresponds to a fixed set of system parameters. Three users in the downlink.

$p = 0.4109,\ r = 0.0226,\ N = 4$ $\pi_m = [0.3869\ 0.8476\ 0.8608\ 0.8535]$ $P_D(1) = 1, P_D(d \neq 1) = 0$			
m	$V_m(\pi_m, \text{opt})$	$V_m(\pi_m, \text{greedy})$	%subopt
1	0.8608	0.8608	0 %
2	1.2176	1.2176	0 %
3	1.3967	1.3967	0 %
4	1.5179	1.5162	0.1072 %
5	1.6061	1.5945	0.7179 %
6	1.6694	1.6558	0.8188 %
7	1.7219	1.7069	0.8715 %

$p = 0.9464,\ r = 0.1666,\ N = 4$ $\pi_m = [0.6898\ 0.6996\ 0.0619\ 0.4757]$ $P_D(0) = 0.5387, P_D(1) = 0.4613$ $P_D(d > 1) = 0$			
m	$V_m(\pi_m, \text{opt})$	$V_m(\pi_m, \text{greedy})$	%subopt
1	0.6996	0.6996	0 %
2	1.4988	1.4988	0 %
3	2.3743	2.3715	0.1179 %
4	3.2675	3.2596	0.2418 %
5	4.1651	4.1558	0.2233 %
6	5.0662	5.0558	0.2053 %
7	5.9700	5.9586	0.1910 %

$p = 0.9281,\ r = 0.2824,\ N = 4$ $\pi_m = [0.4541\ 0.6528\ 0.6477\ 0.5767]$ $P_D(0) = 0.6647, P_D(1) = 0.1844$ $P_D(2) = 0.1510, P_D(d > 2) = 0$			
m	$V_m(\pi_m, \text{opt})$	$V_m(\pi_m, \text{greedy})$	%subopt
1	0.6528	0.6528	0 %
2	1.4533	1.4533	0 %
3	2.3190	2.3170	0.0844 %
4	3.2070	3.2015	0.1699 %
5	4.1006	4.0936	0.1705 %
6	4.9965	4.9888	0.1533 %
7	5.8934	5.8854	0.1364 %

Table 3.2: Illustration of the near optimal performance of the greedy policy. Each table corresponds to a fixed set of system parameters. Four users in the downlink.

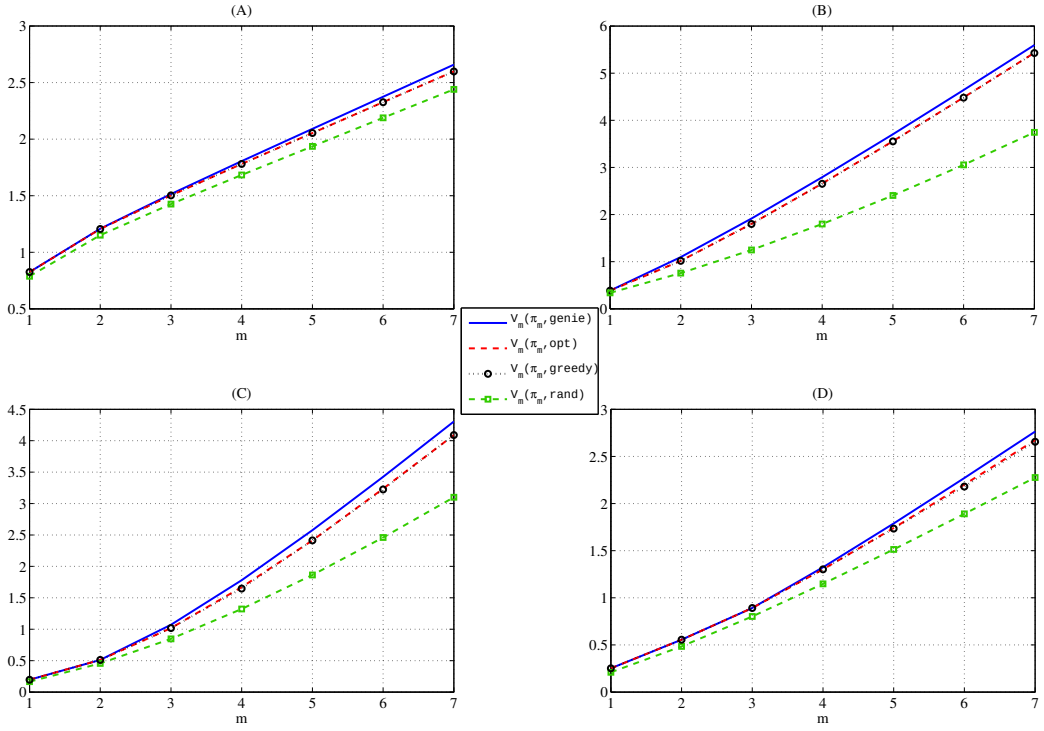

Figure 3.3: Total expected reward of the greedy policy in comparison with system-level performance limits. System parameters used: plot (A) $N = 3$, $p = 0.4070$, $r = 0.1999$, $P_D(0) = 0.3379, P_D(1) = 0.5666, P_D(2) = 0.0954$, $\pi_m = [0.7487\ 0.8256\ 0.7900]$, (B) $N = 3$, $p = 0.9930$, $r = 0.1267$, $P_D(0) = 0.8855, P_D(1) = 0.1145$, $\pi_m = [0.3631\ 0.2662\ 0.3857]$, (C) $N = 3$, $p = 0.9694$, $r = 0.1556$, $P_D(0) = 0, P_D(1) = 1$, $\pi_m = [0.1207\ 0.1962\ 0.1791]$, (D) $N = 3$, $p = 0.7965$, $r = 0.1365$, $P_D(0) = 0, P_D(1) = 0, P_D(2) = 1$, $\pi_m = [0.1351\ 0.2523\ 0.2410]$.

$[0,1]$) and normalized by $\sum_d (P_D(d))$. We see that the value of %subopt is below 1% for all the system parameters considered, thus suggesting near optimal performance of the greedy policy.

We study the performance of the greedy policy in a larger perspective in Fig. 3.3. Here $V_m(\pi_m, \text{greedy})$ is plotted alongside $V_m(\pi_m, \text{opt})$, $V_m(\pi_m, \text{genie})$ and $V_m(\pi_m, \text{rand})$. $V_m(\pi_m, \text{genie})$ corresponds to the total expected reward when, for any k, feedback f_k includes the channel state information, corresponding to slot k, of not only the scheduled user a_k but also that of all the users in the system. We call this the genie-aided system. The quantity $V_m(\pi_m, \text{rand})$ is the total expected reward when random scheduling is performed. From Fig. 3.3 we observe that the greedy policy achieves a performance comparable to that of the optimal policy in both the original and the genie-aided systems, while $V_m(\pi_m, \text{rand})$ is significantly lower than $V_m(\pi_m, \text{greedy})$. These observations, apart from demonstrating the near optimality of the greedy policy, underline the effectiveness of our larger approach: exploit multiuser diversity and use delayed ARQ feedback for that purpose.

3.3.3 Structure of the Greedy Policy

Motivated by the near optimal performance of the greedy policy, we proceed to study its structure, which turns out to be very amenable for practical implementation. We begin by defining the following quantity:

Schedule order vector, O_t, in slot t: The user indices in decreasing order of $\pi_t(i)$, i.e.,

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

Thus, the greedy decision in slot t is $\hat{a}_t = O_t(1)$.

Now, in any slot $t \leq m$, any user i falls under one of the following two cases:

1) The scheduler has received at least one ARQ feedback from user i by the beginning of slot t. Let k_i, for $m \geq k_i > t$, be the latest slot for which the ARQ feedback from user i is available at the scheduler. Since the channel is first-order Markovian, the belief value of the channel of user i in the current slot t is dependent only on the feedback f_{k_i} and k_i. The belief value is given by

$$\pi_t(i) = \begin{cases} T^{k_i - t - 1}(p) & \text{if } f_{k_i} = 1 \\ T^{k_i - t - 1}(r) & \text{if } f_{k_i} = 0. \end{cases} \tag{3.24}$$

2) The scheduler does not have any ARQ feedback from user i by the beginning of slot t. In this case

$$\pi_t(i) = T^{(m-t)}\pi_m(i). \tag{3.25}$$

Recall that $\pi_m(i)$ is the initial belief value of the channel of user i when the scheduling process started at slot m.

At slot t, let $\mathcal{A}_t$ denote the set of users, i, whose latest feedback, f_{k_i}, is an ACK. Let $\mathcal{N}_t$ denote the set of users, j, whose latest feedback, f_{k_j}, is a NACK. Let the users from whom the scheduler has not yet received any feedback constitute set $\mathcal{X}_t$. From (3.24) and (3.25), using Lemma 5, the greedy decision in slot t can be written as

$$\hat{a}_t = \begin{cases} \arg\min_{i \in \mathcal{A}_t} k_i & \text{if } \mathcal{A}_t \neq \emptyset \\ \arg\max_{i \in \mathcal{X}_t} \pi_m(i) & \text{if } \mathcal{A}_t = \emptyset \text{ and } \mathcal{X}_t \neq \emptyset \\ \arg\max_{i \in \mathcal{N}_t} k_i & \text{if } \mathcal{A}_t = \emptyset \text{ and } \mathcal{X}_t = \emptyset. \end{cases} \tag{3.26}$$

Now, for ease of implementation, we visualize the sets $\mathcal{A}_t$, $\mathcal{X}_t$ and $\mathcal{N}_t$ as queues with elements ordered in the following specific ways: Let $\mathcal{A}_t(i)$ denote the i^{th} element of

queue $\mathcal{A}_t$ and the elements be ordered such that $k_{\mathcal{A}_t(1)} < k_{\mathcal{A}_t(2)} \ldots < k_{\mathcal{A}_t(n(A_t))}$, where $n(A)$ denotes the cardinality of set A. Note that the user that gave an ACK from the most recent slot lies at the head of queue $\mathcal{A}_t$. The elements of $\mathcal{X}_t$ are ordered such that $\pi_m(\mathcal{X}_t(1)) \geq \pi_m(\mathcal{X}_t(2)) \ldots \geq \pi_m(\mathcal{X}_t(n(X_t)))$. The elements of $\mathcal{N}_t$ satisfy $k_{\mathcal{N}_t(1)} > k_{\mathcal{N}_t(2)} \ldots > k_{\mathcal{N}_t(n(N_t))}$, i.e., the user with the oldest NACK feedback lies on top of queue $\mathcal{N}_t$. Define a combined queue constructed by concatenating the queues $\mathcal{A}_t$, $\mathcal{X}_t$ and $\mathcal{N}_t$ in that order. From (3.24) and (3.25), using Lemma 5, we see that the users in the combined queue are arranged in decreasing order (top-down) of belief values with the top-most user being the greedy decision in slot t. Thus the combined queue is, in fact, the schedule order vector O_t.

We now discuss the evolution of the schedule order vector. For every user a whose ARQ feedback is contained in F_t, implement the following procedure: Let t_a indicate the originating slot for the ARQ feedback from user a contained in F_t. Now, if t_a is the latest slot from which the ARQ feedback of user a is available at the scheduler, then $k_a = t_a$. The new schedule order vector O_{t-1} is formed by removing user a from its current position (in O_t) and placing it in the sub-queue $\mathcal{A}_{t-1}$ (if $f_{k_a} = 1$) or in the sub-queue $\mathcal{N}_{t-1}$ (if $f_{k_a} = 0$) at an appropriate location (so that the ordering based on k_i is not violated). If $t_a \neq k_a$, i.e., t_a is not the latest slot, then user a is not moved. Similarly, users whose ARQ feedback are not contained in F_t are not moved. The last two statements are direct consequences of the following facts:

- For an user a whose ARQ feedback is contained in F_t but is not the latest feedback from that user, the belief value evolves as $\pi_{t-1}(a) = T(\pi_t(a))$. Similarly, for an user b whose ARQ feedback is *not* contained in F_t, the belief value evolves as $\pi_{t-1}(b) = T(\pi_t(b))$. Both these cases were discussed in Section 3.2.3.

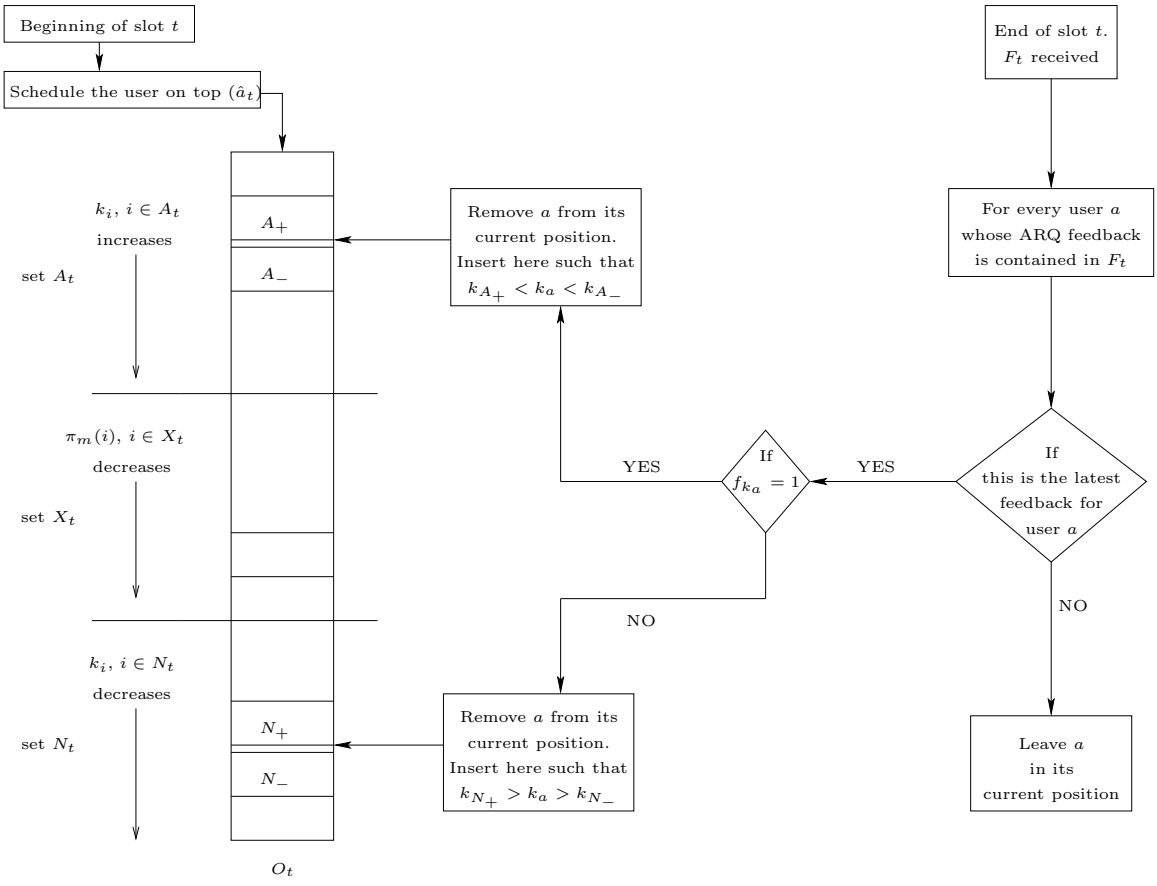

Figure 3.4: Greedy policy implementation under random ARQ delay.

- From Lemma 5, if $x \geq y$, then $T(x) \geq T(y)$.

Now, at slot $t-1$, the user on top of O_{t-1} is the greedy decision. Thus the greedy decision in any slot is determined by the latest ARQ feedback and the corresponding originating slot index of all the users in the system. Note that this implementation does not require the Markov channel statistics (other than the knowledge that $p > r$) and the statistics of the ARQ feedback delay. An illustration of the greedy policy implementation is provided in Fig. 3.4.

For the special case of deterministic ARQ feedback delay $D = d$, the evolution from O_t to O_{t-1} is greatly simplified as follows. At the end of slot t, since $D = d$, F_t contains feedback only from the user scheduled in slot $t+d$, i.e., user $\hat{a}_{t+d}$. Thus $F_t = f_{t+d}$. The feedback bits $f_m, f_{m-1}, \ldots, f_{t+d+1}$ from users $\hat{a}_m, \hat{a}_{m-1}, \ldots, \hat{a}_{t+d+1}$ have already arrived at the end of slots $m-d, m-1-d, \ldots, t+1$ and the feedback from users $\hat{a}_{t+d-1}, \hat{a}_{t+d-2}, \ldots$ are yet to arrive. Thus $F_t = f_{t+d}$ from user $\hat{a}_{t+d}$ is the latest feedback available from *any* user. Thus, recalling the ordering rules for $\mathcal{A}_{t-1}$ and $\mathcal{N}_{t-1}$, if $F_t = 1$, user $\hat{a}_{t+d}$ is removed from its current position and placed on top in the updated schedule order vector, i.e., $O_{t-1} = [\hat{a}_{t+d} \quad O_t - \hat{a}_{t+d}]$, [7] (user $\hat{a}_{t+d}$ becomes the greedy decision in slot $t-1$). If $F_t = 0$, $\hat{a}_{t+d}$ is placed at the bottom, i.e., $O_{t-1} = [O_t - \hat{a}_{t+d} \quad \hat{a}_{t+d}]$. When there is no ARQ delay ($D = d = 0$), the implementation becomes even simpler: on receiving an ACK, $O_{t-1} = O_t$, and on NACK, $O_{t-1} = [O_t - O_t(1) \quad O_t(1)]$, since $\hat{a}_{t+d} = \hat{a}_t = O_t(1)$. This results in a simple round robin implementation of the greedy policy as discussed in [7, 45]. Fig. 3.5 and Fig. 3.6 illustrate the greedy policy implementation in the deterministically delayed ARQ and instantaneous ARQ systems, respectively.

[7] If $Z = [z_1 \; z_2 \; z_3]$ then $Z - z_2 := [z_1 \; z_3]$ and hence $[z_2 \;\; Z - z_2] = [z_2 \; z_1 \; z_3]$

3.4 On Downlink Sum Capacity and Capacity Region

We now proceed to study the fundamental limits on the downlink system performance — the sum capacity and the capacity region.

3.4.1 Sum Capacity of the Downlink

The sum capacity of the downlink is defined as the maximum sum throughput over an infinite horizon with steady state initial conditions. Formally, with N users in the system,

$$C_{\text{sum}}(N) = \max_{\{\mathfrak{A}_k\}_{k\geq 1}} \lim_{m\to\infty} \frac{V_m(\pi_{ss}, \{\mathfrak{A}_k\}_{k\geq 1})}{m}, \tag{3.27}$$

where $\forall i \in \{1,\ldots,N\}$, $\pi_{ss}(i) = p_s$, the steady state probability of the Markov channel. We now proceed to derive p_s. The Markov chain transition matrix $P = \begin{bmatrix} p & 1-p \\ r & 1-r \end{bmatrix}$ can be expressed as $P = U\Lambda V$, where

$$\begin{aligned} U &= \begin{bmatrix} 1 & 1 \\ 1 & \frac{-r}{1-p} \end{bmatrix} \\ \Lambda &= \begin{bmatrix} 1 & 0 \\ 0 & p-r \end{bmatrix} \\ V &= \frac{1}{1+\frac{r}{1-p}} \begin{bmatrix} \frac{r}{1-p} & 1 \\ 1 & -1 \end{bmatrix}, \end{aligned}$$

with $VU = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. Assuming[8] $p + (1-r) < 2$,

$$\begin{aligned} \lim_{n\to\infty} P^n &= \begin{bmatrix} \frac{r}{1-(p-r)} & 1-\frac{r}{1-(p-r)} \\ \frac{r}{1-(p-r)} & 1-\frac{r}{1-(p-r)} \end{bmatrix} \\ \Rightarrow p_s &= \frac{r}{1-(p-r)}. \end{aligned}$$

[8] $p + (1-r) = 2$ leads to $P = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, a trivial case with no steady state.

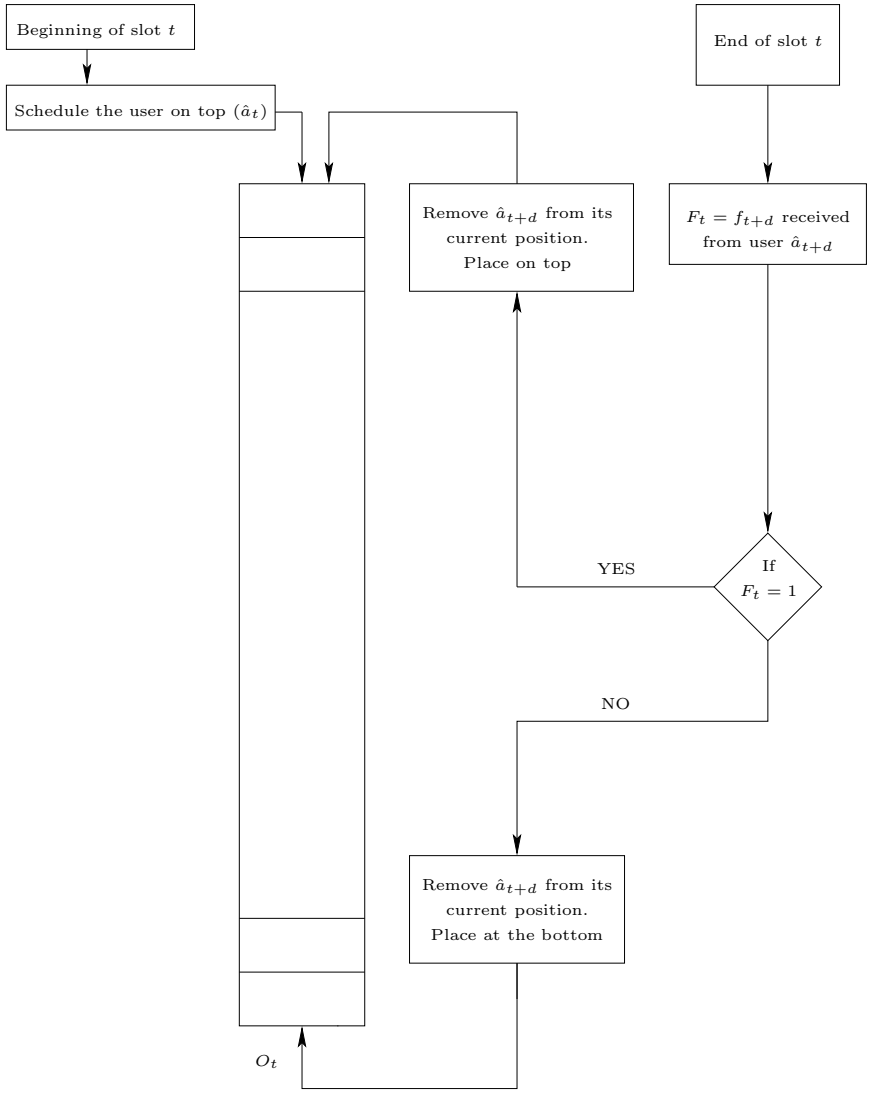

Figure 3.5: Greedy policy implementation under deterministically delayed ARQ, i.e., $D = d$.

We now define the genie-aided system formally as follows: In any slot k, the feedback f_k contains the channel state information, corresponding to slot k, of not only the scheduled user but also that of all the users in the system. We retain the delay profile from the original system. Thus, in the genie-aided system, the cumulative feedback f_k arrive at the scheduler with delay $D(a_k, k)$ that is *i.i.d* across scheduling choice a_k and originating slot k with the probability mass function $P_D(d)$. We now report our result on the sum capacity of the downlink with two users.

Proposition 7. *When $N = 2$, the sum capacity of the Markov-modeled downlink with randomly delayed ARQ equals that of the genie-aided system. This sum capacity equals*

$$\begin{aligned} & C_{\text{sum}}(N=2) \\ &= \sum_{l=0}^{\infty} \Big[p_s T^l(p) + (1-p_s)p_s \Big] P(D \leq l) \prod_{d=0}^{l-1} P(D > d). \end{aligned} \tag{3.28}$$

Furthermore, the greedy policy achieves this sum capacity.

Proof. Recall the definition of the genie-aided system: the feedback f_k from any slot k contains the channel state information, corresponding to slot k, of not only the scheduled user but also that of all the users in the system. Thus, in the genie-aided system, since the delay of the (cumulative) feedback f_k is *i.i.d* across the scheduling choice, the scheduling decision in the current slot does not affect the information available for scheduling in future slots. Hence, the greedy policy is optimal in the genie-aided system.

We now focus on the sum throughput of the greedy policy in the genie-aided system. Recall, from Section 3.3.1, the quantity L – the measure of freshness of the

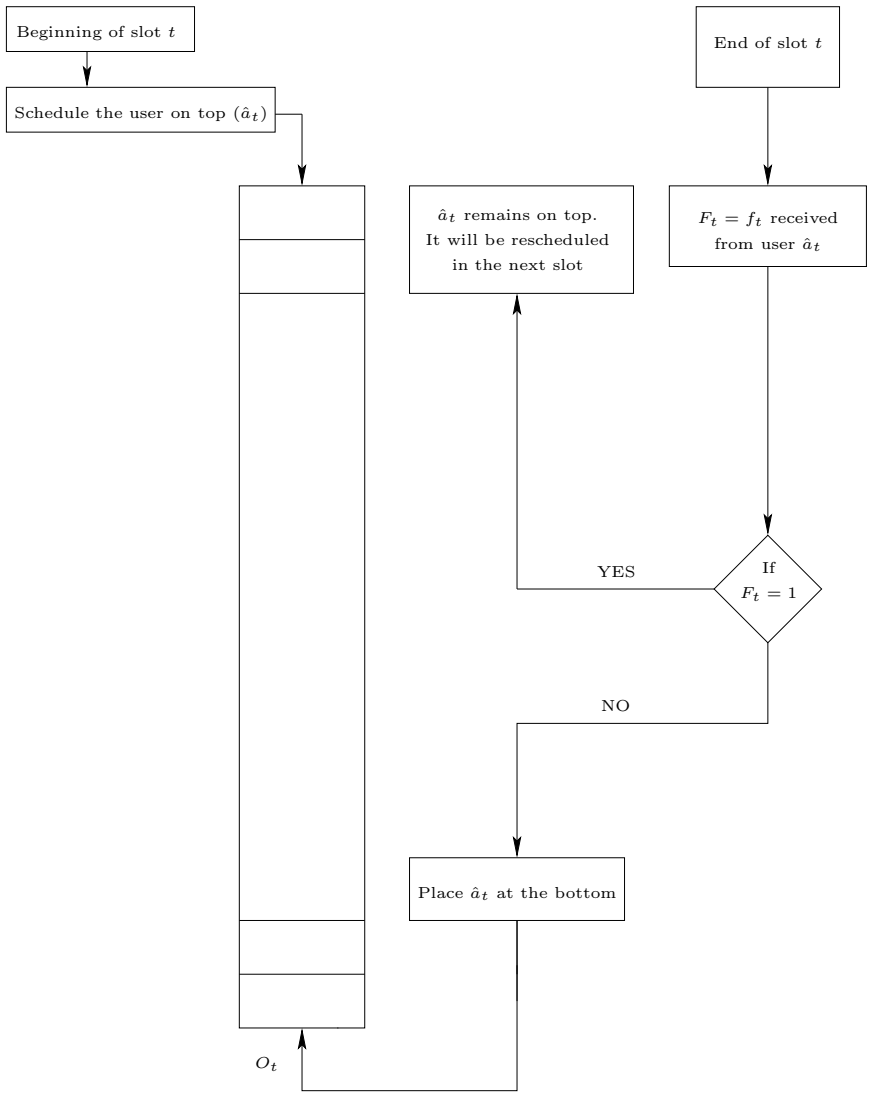

Figure 3.6: Greedy policy implementation under instantaneous (end of slot) ARQ, i.e., $D = 0$.

latest ARQ feedback. We defined L such that $L = l \Rightarrow$ the latest feedback is $l+1$ slots old. We extend the meaning of L to the genie-aided system. Due to the first order Markovian nature of the channels, in the genie-aided system, conditioned on the latest feedback, f_{t+l+1} (with t denoting the current slot), the belief values (and hence the greedy scheduling decision) in the current slot are independent of the feedback from previous slots, i.e., $f_{k,k>t+l+1}$. Thus, with $R^{\text{greedy}}_{\text{genie}}(l,N)$ denoting the conditional (conditioned on $L=l$) immediate reward corresponding to the greedy policy, in the N-user genie-aided system with steady state initial conditions, the sum capacity of the genie-aided system can be written as

$$C^{\text{genie}}_{\text{sum}}(N) \;=\; \mathrm{E}_l\, R^{\text{greedy}}_{\text{genie}}(l,N). \tag{3.29}$$

We now evaluate $R^{\text{greedy}}_{\text{genie}}(l,N)$. From Lemma 5, the belief value (in the current slot) of an user with an ON channel $l+1$ slots earlier, i.e., $T^l(p)$, is higher than the belief value of an user with an OFF channel $l+1$ slots earlier, i.e., $T^l(r)$. Thus, in steady state,

$$\begin{aligned} & R^{\text{greedy}}_{\text{genie}}(l,N) \\ &= P(\text{at least one of the } N \text{ users has an ON channel in} \\ &\qquad \text{steady state})T^l(p) \\ &\quad + P(\text{all users have OFF channels in steady state})T^l(r) \\ &= (1-(1-p_s)^N)T^l(p) + (1-p_s)^N T^l(r). \end{aligned} \tag{3.30}$$

We now focus on the probability $P(L=l)$. With t as the current slot, the latest feedback is $l+1$ slots old if (a) the feedback from slot $t+l+1$ (f_{t+l+1}) arrives at the scheduler by the end of slot $t+1$, and (b) the feedback from the later slots

$(t+l,\ldots,t+1)$ do not arrive at the scheduler by the end of slot $t+1$. Since the feedback delay is *i.i.d* across users and time, events (a) and (b) are independent with probabilities given by $P(D \leq l)$ and $\prod_{d=0}^{l-1} P(D > d)$, respectively. Thus the sum capacity of the genie-aided system with N users is given by

$$\begin{aligned} & C_{\text{sum}}^{\text{genie}}(N) \\ &= \mathrm{E}_l \, R_{\text{genie}}^{\text{greedy}}(l, N) \\ &= \sum_{l=0}^{\infty} \Big[(1-(1-p_s)^N) T^l(p) + (1-p_s)^N T^l(r) \Big] \times \\ &\qquad\qquad P(D \leq l) \prod_{d=0}^{l-1} P(D > d). \end{aligned} \tag{3.31}$$

When $N = 2$, with minor algebraic manipulations, we have

$$\begin{aligned} & C_{\text{sum}}^{\text{genie}}(2) \\ &= \sum_{l=0}^{\infty} \Big[p_s T^l(p) + (1-p_s) p_s \Big] P(D \leq l) \prod_{d=0}^{l-1} P(D > d). \end{aligned} \tag{3.32}$$

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 when $N = 2$. We established in the course of the proof of Proposition 5 that, in the original system with $N = 2$, conditioned on $L = l$, the greedy decision in the current slot t is solely determined by the ARQ feedback from slot $t+l+1$ with the following decision rule: When the user scheduled in slot $t+l+1$, i.e., a_{t+l+1}, sends back an ACK, that user is scheduled in the current slot t, i.e., $\hat{a}_t = a_{t+l+1}$. Otherwise, the other user is scheduled in slot t. We can interpret this decision logic of the greedy policy as below:

When at least one of the users had an ON channel in slot $t+l+1$, that user[9] *is identified for scheduling in the current slot t, leading to an expected current reward of $T^l(p)$. Reward $T^l(r)$ is accrued only when both the channels were in the OFF state in slot $t+l+1$.*

Note that the decision rule and the accrued immediate rewards corresponding to the greedy policy in the original system are the same as that of the greedy policy in the genie-aided system. Thus, in the original system, under the greedy policy, no improvement in the immediate reward can be achieved even if the channel states of both the users in slot $t+l+1$ are available at the scheduler in slot t. This, along with the fact that both the systems have the same delay profile, establishes the equivalence between the original and the genie-aided systems, when $N=2$, 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 $N=2$ (Proposition 5) and in the genie-aided system for a general value of N. Thus the sum capacity of the original system for $N=2$ is given by $C_{\text{sum}}^{\text{genie}}(2)$ in (3.32). The proposition thus follows. □

Remarks: Insights on the result in Proposition 7 can be obtained by examining the fundamental trade-off when scheduling in the Markov-modeled downlink. In particular, scheduling must take into account

1) data transmission in the current slot, which influences the immediate reward, and

[9]User a_{t+l+1} is given higher priority if both channels were ON.

2) probing of the channel for future scheduling decisions, which influences the reward expected in future slots.

The optimal schedule strikes a balance between these two objectives (that need not contradict each other). From the discussion in the proof of Proposition 7, we see that, in the original system, when $N = 2$, the choice of the user whose channel is probed becomes irrelevant as far as the optimal future reward is concerned. Similarly, in the genie-aided system, since the channel state information of all the users (general N system) is sent to the scheduler (with equal delay that is *i.i.d* across the scheduling choice) irrespective of which user was scheduled, the optimal future reward is independent of the current scheduling decision. This results in the optimality of the greedy policy in the original and the genie-aided systems and creates a sum capacity equivalence between these two systems, when $N = 2$.

The equivalence with the genie-aided system vanishes when $N > 2$, since observing only one user is not enough to capture an 'ON-user', if one exists. This was possible when $N = 2$. Thus, when $N > 2$, there is room for throughput improvement when the channel state information of all the users is available at the scheduler even if there is a delay (the genie-aided system). The genie-aided system sum capacity is thus an upper bound to the sum capacity of the original system. We record this next.

Corollary 4. *When $N > 2$, the sum capacity, $C_{\text{sum}}(N)$, of the downlink can be bounded as*

$$C_{\text{sum}}(2) \leq C_{\text{sum}}(N) \leq C_{\text{sum}}^{\text{genie}}(N) \tag{3.33}$$

Proof. The lower bound $C_{\text{sum}}(2)$, given in (3.28), is achieved by the scheduler when, in each slot, it considers only two users (fixed set) for scheduling and ignores the rest,

effectively emulating a two-user downlink. The upper bound is the sum capacity of the genie-aided system with N users, as given in (3.31). $\square$

3.4.2 Bounds on the Capacity Region of the Downlink

Define the capacity region of the downlink as the *exhaustive* set of achievable throughput vectors. Formally, let $\mu_i^{\mathfrak{A}}$ denote the throughput of user i under policy $\mathfrak{A}$. Let $I_k(i)$ be the indicator function on whether user i was scheduled in slot k, i.e.,

$$I_k(i) \;=\; \begin{cases} 1 & \text{if } i = a_k \\ 0 & \text{otherwise.} \end{cases} \tag{3.34}$$

Thus

$$\mu_i^{\mathfrak{A}} \;=\; \lim_{m\to\infty} \frac{\mathrm{E}\left[\sum_{k=1}^{m} R_k^{\mathfrak{A}}(\pi_k, a_k) I_k(i)\right]}{m}, \tag{3.35}$$

where $R_k^{\mathfrak{A}}(\pi_k, a_k)$ is the immediate reward accrued by the scheduler in slot k under policy $\mathfrak{A}$. The expectation is over the belief vector π_k with steady state initial conditions. Now, the capacity region of the downlink, $\mathcal{C}$, is defined as the union of the throughput vectors, $(\mu_1^{\mathfrak{A}}, \ldots, \mu_N^{\mathfrak{A}})$, over all scheduling policies, i.e.,

$$\mathcal{C} \;=\; \cup_{\mathfrak{A}} \{(\mu_1^{\mathfrak{A}}, \ldots, \mu_N^{\mathfrak{A}})\}. \tag{3.36}$$

Let $H_{\text{convex}}(X)$ be the convex hull of the set of points X, defined as

$$\begin{aligned} & H_{\text{convex}}(X) \\ &= \left\{ \sum_{i=1}^{n(X)} \beta_i x_i \;\middle|\; x_i \in X, \beta_i \in \mathbb{R}, \beta_i \geq 0, \sum_{i=1}^{n(X)} \beta_i = 1 \right\}. \end{aligned}$$

where $n(X)$ is the cardinality of set X. With these definitions we now state our results on the downlink capacity region.

Proposition 8. *An outer bound on the capacity region of the Markov-modeled downlink with randomly delayed ARQ is given by the complement of the N-dimensional polyhedron $\mathcal{P}$ represented by*

$$\begin{aligned}\mathcal{P} &= \Big\{(x_1 \geq 0, x_2 \geq 0 \ldots x_N \geq 0) : \\ &\quad \sum_{i \in S} x_i \leq C_{\text{sum}}^{\text{genie}}(n(S)), \forall S \subseteq \{1, \ldots N\}\Big\}, \end{aligned} \tag{3.37}$$

where

$$C_{\text{sum}}^{\text{genie}}(N) = \sum_{l=0}^{\infty} \Big[(1-(1-p_s)^N)T^l(p) + (1-p_s)^N T^l(r)\Big] P(D \leq l) \prod_{d=0}^{l-1} P(D > d).$$

An inner bound on the capacity region is given by the set of points $(x_1, \ldots, x_N)$ such that

$$(x_1, \ldots, x_N) \in H_{\text{convex}}(O, \{X_i\}_{\forall i \in \{1,\ldots,N\}}, \{Y_{j,k}\}_{\forall j,k \in \{1,\ldots,N\}, j \neq k}) \tag{3.38}$$

where $O, X_i, Y_{j,k} \in \mathbb{R}^N$. O is the origin $(0, \ldots, 0)$. $X_i = (0, \ldots, 0, p_s, 0, \ldots, 0)$ with p_s at the i^{th} location. $Y_{j,k,j\neq k} = (0, \ldots, 0, \frac{C_{\text{sum}}(2)}{2}, 0, \ldots, 0, \frac{C_{\text{sum}}(2)}{2}, 0, \ldots, 0)$ with $\frac{C_{\text{sum}}(2)}{2}$ at locations j and k, where

$$C_{\text{sum}}(2) = \sum_{l=0}^{\infty} \Big[p_s T^l(p) + (1-p_s)p_s\Big] P(D \leq l) \prod_{d=0}^{l-1} P(D > d).$$

Proof. Considering the genie-aided system, for any policy $\mathfrak{A}$, let the throughput vector be denoted by $(\mu_1^{\mathfrak{A},\text{genie}}, \ldots, \mu_N^{\mathfrak{A},\text{genie}})$. For a subset of users $S \subseteq \{1 \ldots N\}$, by the definition of sum capacity, we have

$$\sum_{i \in S} \mu_i^{\mathfrak{A},\text{genie}} \leq C_{\text{sum}}^{\text{genie}}(n(S)). \tag{3.39}$$

This establishes the complement of the polyhedron $\mathcal{P}$ as an outer bound on the capacity region of the genie-aided system, and by extension, an outer bound on the capacity region of the original system.

Now, consider the inner bound $H_{\text{convex}}(O, \{X_i\}_{\forall i \in \{1,\ldots,N\}}, \{Y_{j,k}\}_{\forall j,k \in \{1,\ldots,N\}, j \neq k})$. In the original system, throughput vector $X_i = (0, \ldots, 0, p_s, 0, \ldots, 0)$ can be achieved by scheduling to user i at all times. Recall that the greedy policy achieves the sum capacity when $N = 2$. Also the sum throughput $C_{\text{sum}}(2)$ is split equally between the two users thanks to the inherent symmetry between users. Thus throughput vector $Y_{j,k,j\neq k} = (0, \ldots, 0, \frac{C_{\text{sum}}(2)}{2}, 0, \ldots, 0, \frac{C_{\text{sum}}(2)}{2}, 0, \ldots, 0)$ can be achieved by greedy scheduling over the users j and k alone at all slots. Throughput vector O corresponds to idling in every slot. Therefore, any throughput vector in the convex hull $H_{\text{convex}}(O, \{X_i\}_{\forall i \in \{1,\ldots,N\}}, \{Y_{j,k}\}_{\forall j,k \in \{1,\ldots,N\}, j \neq k})$ can be achieved by time sharing between the policies that achieve throughput vectors $\in \{O, X_i, Y_{j,k,j\neq k}\}$. This establishes the result on the inner bound. □

Fig. 3.7 illustrates the capacity region bounds from Proposition 8 when $N = 2$ and when $N = 3$.

For the special case of $N = 2$ users and deterministic ARQ feedback delay, $D = d$, we obtain the exact capacity region of the genie-aided system and hence tighter bounds to the capacity region of the original system.

Proposition 9. *For $N = 2$ users, with a deterministic ARQ delay of $D = d$, $d \geq 0$ slots, the capacity region of the genie-aided system is given by the set of points (x_1, x_2)*

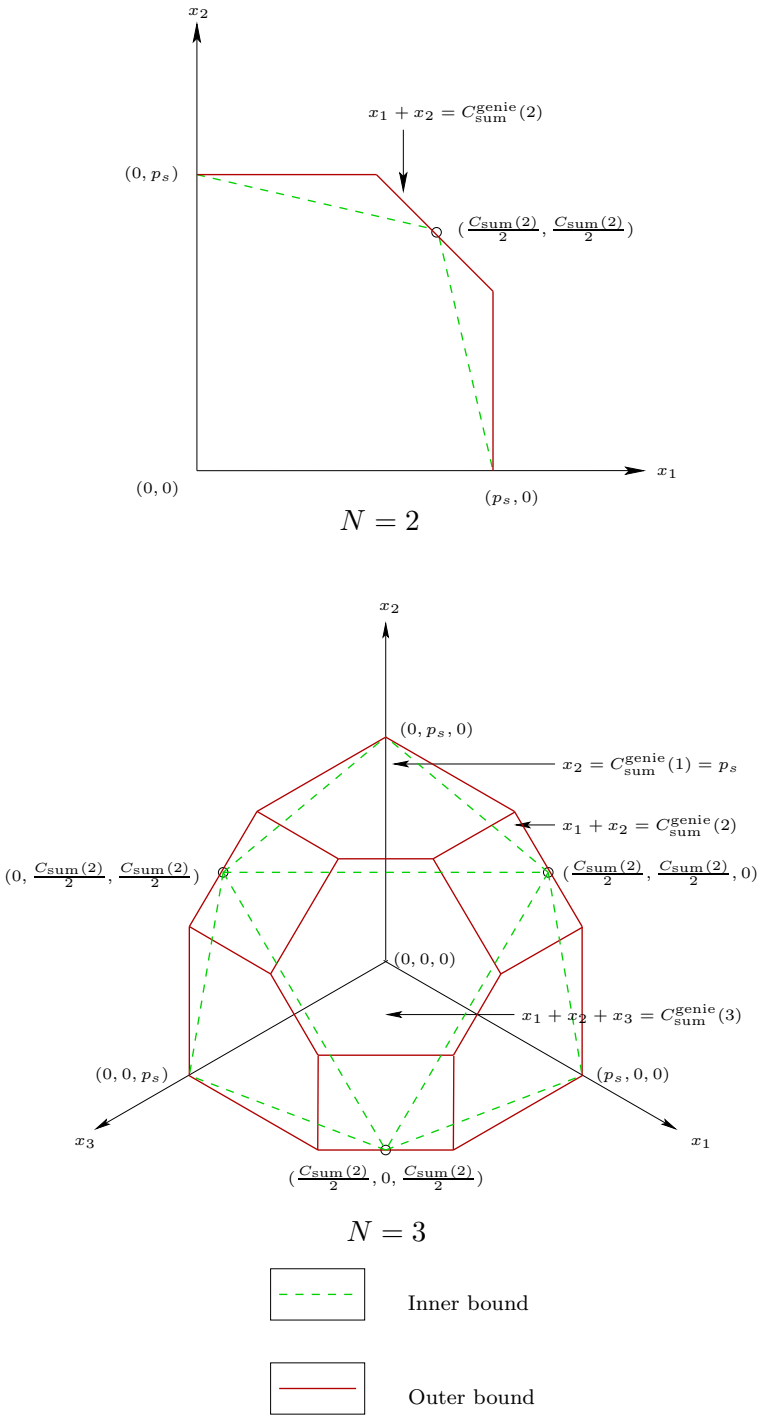

Figure 3.7: Illustration of bounds on the capacity region of the downlink with randomly delayed ARQ when $N = 2$ and when $N = 3$.

such that

$$\begin{aligned}(x_1, x_2) &\in H_{\text{convex}}(O, X_1, Z_1, Z_2, X_2)\\ \textit{where } O &= (0,0)\\ X_1 &= (p_s, 0)\\ X_2 &= (0, p_s)\\ Z_1 &= \big(p_s T^d(p) + (1-p_s)^2 T^d(r), (1-p_s)p_s T^d(p)\big)\\ Z_2 &= \big((1-p_s)p_s T^d(p), p_s T^d(p) + (1-p_s)^2 T^d(r)\big). \end{aligned} \tag{3.40}$$

Proof. The relative positions of the points $X_1,\ X_2,\ Z_1,\ Z_2$ and O are illustrated in Fig. 3.8.

We first show that the region complementary to $H_{\text{convex}}(O, X_1, Z_1, Z_2, X_2)$ is an outer bound on the capacity region of the genie-aided downlink. Consider a broad class of schedulers in the genie-aided system, with each member identified by the parameters $\alpha_i \in [0,1],\ i \in \{1, \ldots, 4\}$. A member of this class obeys the following decision logic at slot t:

- If $\begin{bmatrix} S_{t+d+1}(1) \\ S_{t+d+1}(2) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, then schedule user 1 with probability α_1 and user 2 w.p. $1-\alpha_1$.
- If $\begin{bmatrix} S_{t+d+1}(1) \\ S_{t+d+1}(2) \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, then $a_t = \begin{cases} 1 \ w.p.\ \alpha_2 \\ 2 \ w.p.\ 1-\alpha_2 \end{cases}$
- If $\begin{bmatrix} S_{t+d+1}(1) \\ S_{t+d+1}(2) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, then $a_t = \begin{cases} 1 \ w.p.\ \alpha_3 \\ 2 \ w.p.\ 1-\alpha_3 \end{cases}$
- If $\begin{bmatrix} S_{t+d+1}(1) \\ S_{t+d+1}(2) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, then $a_t = \begin{cases} 1 \ w.p.\ \alpha_4 \\ 2 \ w.p.\ 1-\alpha_4 \end{cases}$

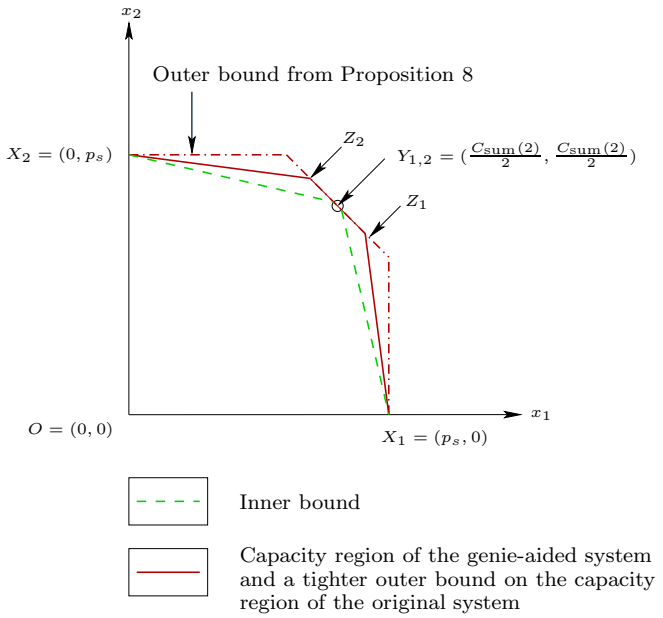

Figure 3.8: Illustration of the capacity region of the genie-aided system and tighter bounds on the capacity region of the original system when $N = 2$, with deterministic ARQ delay.

Note that, thanks to the first order Markovian nature of the underlying channels, any scheduling policy in the genie-aided system falls under the above class of schedulers or will have a member of this class achieving the same throughput vector as itself. We now proceed to show that the throughput vector achieved by any member of this class belongs to $H_{\text{convex}}(O, X_1, Z_1, Z_2, X_2)$.

With $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_4\} \in [0, 1]^4$ fixed, the throughput for user 1 is given by

$$\begin{aligned}
\mu_1^{\boldsymbol{\alpha},\text{genie}} &= \sum_{i,j\in\{0,1\}} P\Big(\begin{bmatrix} S_{t+d+1}(1) \\ S_{t+d+1}(2) \end{bmatrix} = \begin{bmatrix} i \\ j \end{bmatrix} \Big) \times \\
&\quad P\Big(a_t = 1 \Big| \begin{bmatrix} S_{t+d+1}(1) \\ S_{t+d+1}(2) \end{bmatrix} = \begin{bmatrix} i \\ j \end{bmatrix} \Big) P(S_t(1) = 1 | S_{t+d+1}(1) = i) \\
&= (1 - p_s)^2 \alpha_1 T^d(r) + (1 - p_s) p_s \alpha_2 T^d(r) \\
&\quad + p_s(1 - p_s)\alpha_3 T^d(p) + p_s^2 \alpha_4 T^d(p), \qquad (3.41)
\end{aligned}$$

with $p_s = \frac{r}{1-(p-r)}$. Similarly,

$$\begin{aligned}\mu_2^{\boldsymbol{\alpha},\text{genie}} &= (1-p_s)^2(1-\alpha_1)T^d(r) + (1-p_s)p_s(1-\alpha_2)T^d(p) \\ &\quad + p_s(1-p_s)(1-\alpha_3)T^d(r) + p_s^2(1-\alpha_4)T^d(p). \end{aligned} \tag{3.42}$$

For notational simplicity, we will henceforth denote the throughputs simply by μ_1 and μ_2. The sum throughput is now given by

$$\mu_1 + \mu_2 = p_s + (1-p_s)p_s(T^d(p) - T^d(r))(\alpha_3 - \alpha_2). \tag{3.43}$$

Note that the values of α_1 and α_4 are irrelevant from the sum throughput point of view. Consider the following two cases.

Case 1, when $\alpha_3 \leq \alpha_2$:

$$0 \leq \mu_1 + \mu_2 \leq p_s.$$

Since $X_1(1) + X_1(2) = X_2(1) + X_2(2) = p_s$, we have

$$(\mu_1, \mu_2) \in H_{\text{convex}}(O, X_1, X_2). \tag{3.44}$$

Case 2, when $\alpha_3 > \alpha_2$:

$$\begin{aligned} p_s < \mu_1 + \mu_2 &\leq p_s + (1-p_s)p_s(T^d(p) - T^d(r)) \\ &= p_s T^d(p) + (1-p_s)p_s. \end{aligned}$$

Since $Z_1(1) + Z_1(2) = Z_2(1) + Z_2(2) = p_s T^d(p) + (1-p_s)^2 T^d(r) + (1-p_s)p_s T^d(p) = p_s T^d(p) + (1-p_s)p_s$, we can find points $E_{X_1Z_1}$ and $E_{X_2Z_2}$ on edges X_1Z_1 and X_2Z_2, respectively, such that $E_{X_1Z_1}(1) + E_{X_1Z_1}(2) = E_{X_2Z_2}(1) + E_{X_2Z_2}(2) = \mu_1 + \mu_2$. Any point $P_{X_1Z_1}$ on the edge X_1Z_1 can be written as a convex combination of points X_1

and Z_1, i.e., $\exists\ \beta \in [0,1]$ such that

$$\begin{aligned} P_{X_1Z_1} &= X_1\beta + Z_1(1-\beta) \\ &= \Big(p_s\beta + (p_sT^d(p) + (1-p_s)^2T^d(r))(1-\beta), \\ &\qquad (1-p_s)p_sT^d(p)(1-\beta)\Big). \end{aligned}$$

With $\beta = 1-(\alpha_3-\alpha_2)$, we have $P_{X_1Z_1}(1) + P_{X_1Z_1}(2) = \mu_1 + \mu_2$. Thus

$$\begin{aligned} E_{X_1Z_1} = \Big(&p_s(1-(\alpha_3-\alpha_2)) + (p_sT^d(p) + (1-p_s)^2T^d(r)) \times \\ &(\alpha_3-\alpha_2), (1-p_s)p_sT^d(p)(\alpha_3-\alpha_2)\Big). \end{aligned}$$

Due to the symmetry between X_1, Z_1 and X_2, Z_2, we have $E_{X_2Z_2} = (E_{X_1Z_1}(2),\ E_{X_1Z_1}(1))$. Using μ_1 from (3.41), it can be shown that, for any $\alpha_{i\in\{1\ldots4\}} \in [0,1]$ with $\alpha_3 > \alpha_2$,

$$E_{X_2Z_2}(1) \leq \mu_1 \leq E_{X_1Z_1}(1). \tag{3.45}$$

Since $E_{X_1Z_1}(1) + E_{X_1Z_1}(2) = E_{X_2Z_2}(1) + E_{X_2Z_2}(2) = \mu_1 + \mu_2$, (3.45) translates to

$$(\mu_1, \mu_2) \in H_{\text{convex}}(E_{X_1Z_1}, E_{X_2Z_2}).$$

The above relation, along with the fact that $E_{X_1Z_1} \in H_{\text{convex}}(X_1, Z_1)$ and $E_{X_2Z_2} \in H_{\text{convex}}(X_2, Z_2)$, yields

$$(\mu_1, \mu_2) \in H_{\text{convex}}(X_1, Z_1, Z_2, X_2). \tag{3.46}$$

Combining the results in (3.44) and (3.46), we establish that the region complementary to $H_{\text{convex}}(O, X_1, Z_1, Z_2, X_2)$ is an outer bound on the capacity region of the genie-aided system.

Revisiting the class of schedulers identified by $\boldsymbol{\alpha}$, it can be shown from (3.41) and (3.42) that a scheduler with $\boldsymbol{\alpha} = \{1,0,1,1\}$ achieves a throughput vector $(\mu_1, \mu_2) =$

$Z_1 = \big(p_s T^d(p) + (1-p_s)^2 T^d(r), (1-p_s)p_s T^d(p)\big)$. Similarly, a scheduler with $\boldsymbol{\alpha} = \{0,0,1,0\}$ achieves a throughput vector $(\mu_1, \mu_2) = Z_2 = \big((1-p_s)p_s T^d(p), p_s T^d(p) + (1-p_s)^2 T^d(r)\big)$. Throughput vectors X_1 or X_2 can be achieved by scheduling to only user 1 or 2, respectively, at all times. Thus any throughput vector within the region $H_{\text{convex}}(O, X_1, Z_1, Z_2, X_2)$ can be supported by time sharing between the schedulers that achieve throughput vector $\in \{O, X_1, Z_1, Z_2, X_2\}$. This establishes $H_{\text{convex}}(O, X_1, Z_1, Z_2, X_2)$ as an inner bound on the capacity region of the genie-aided system.

Combining the outer and inner bound results establishes the proposition. □

We now report tighter bounds on the capacity region of the original system, when $N = 2$ and the ARQ delay is deterministic.

Corollary 5. *For $N = 2$ users, with a deterministic ARQ delay of $D = d$, $d \geq 0$ slots, an outer bound on the capacity region of the original system is given by the set of points (x_1, x_2) such that*

$$\begin{aligned}
(x_1, x_2) &\notin H_{\text{convex}}(O, X_1, Z_1, Z_2, X_2) \\
\textit{where } O &= (0,0) \\
X_1 &= (p_s, 0) \\
X_2 &= (0, p_s) \\
Z_1 &= \big(p_s T^d(p) + (1-p_s)^2 T^d(r), (1-p_s)p_s T^d(p)\big) \\
Z_2 &= \big((1-p_s)p_s T^d(p), p_s T^d(p) + (1-p_s)^2 T^d(r)\big)
\end{aligned} \tag{3.47}$$

and an inner bound is given by the set of points (x_1, x_2) *such that*

$$\begin{aligned}(x_1, x_2) &\in H_{\text{convex}}(O, X_1, Y_{1,2}, X_2) \\ \textit{where } Y_{1,2} &= (\frac{C_{\text{sum}}(2)}{2}, \frac{C_{\text{sum}}(2)}{2})\end{aligned}$$

with $C_{\text{sum}}(2) = p_s T^d(p) + (1 - p_s)p_s$*, the sum capacity of the system.*

Proof. The outer bound is the region complementary to the capacity region of the genie-aided system reported in Proposition 9. The inner bound was obtained in Proposition 8 with $C_{\text{sum}}(2)$ from (3.28) re-derived using $P(D = d) = 1$. □

Fig. 3.8 illustrates the improved outer bound from Corollary 5 along with the bounds derived in Proposition 8.

3.5 Summary

We studied opportunistic multiuser scheduling in Markov-modeled downlink using delayed ARQ feedback from the users. For the case of two users in the system, we showed that the greedy policy is sum throughput optimal for any distribution of the ARQ feedback delay. However, for more than two users, there exists scenarios for which the greedy policy is not optimal. Nevertheless, extensive numerical experiments suggest that the greedy policy has near-optimal performance. Encouraged by this, we studied the structure of the greedy policy and showed that it can be implemented by a simple algorithm that does not require the statistics of the underlying Markov channel nor the ARQ feedback delay, thus making it robust against errors in estimation of these statistics. Focusing on the fundamental limits of the downlink system, we obtained an elegant closed form expression for the sum capacity of the

two-user downlink and derived inner and outer bounds on the capacity region of the Markov-modeled downlink with randomly delayed ARQ feedback.

CHAPTER 4

OPPORTUNISTIC SCHEDULING IN CELLULAR DOWNLINK MODELED BY THREE STATE MARKOV CHAINS

4.1 Background

In the preceding chapter, we studied joint channel estimation - opportunistic scheduling using randomly delayed ARQ feedback in cellular downlink. With the channels modeled by *two*-state Markov chains, we showed the optimality of the greedy policy when the number of downlink users is two. Although modeling the channels by two state Markov chains is a welcome change from the traditional memoryless models, the scheduler can do better by discriminating the channel conditions on a finer level, i.e., if the channel is modeled by higher state Markov chains. Under such a model, it would be interesting to see if the optimality properties of the greedy policy are preserved from the two-state Markov model. This is the focus of this chapter. We begin with a description of the problem setup.

4.2 Problem Setup

4.2.1 Channel Model - Probability Transition Matrix

We consider a cellular downlink with two users. The channel between the scheduler and each user is modeled by an *i.i.d*, first order, *three-state* Markov chain. As before, 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 three-state Markov channel is characterized by a 3×3 probability transition matrix

$$P = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix}, \tag{4.1}$$

where p_{ij} is the probability of evolving from state i to state j in the next slot.

State 1 is assumed to represent the lower end of the channel strength spectrum and state 3 represents the higher end. We assume that the Markov chain is positively correlated in time. Thus $p_{ii} \geq p_{ji}$ if $j \neq i$. Also, motivated by observation of realistic channels, we assume that the channel evolves in a smooth fashion across time. Thus $p_{21} \geq p_{31}$ and $p_{23} \geq p_{13}$. Also, observing that state 3 represents a region of the channel strength spectrum that is not bounded from above, it is reasonable to assume $p_{32} \leq p_{12}$. To summarize, the transition matrix elements are related as below:

$$\begin{aligned} p_{11} \geq p_{21} \geq p_{31} \\ p_{22} \geq p_{12} \geq p_{32} \\ p_{33} \geq p_{23} \geq p_{13} \end{aligned} \tag{4.2}$$

4.2.2 Scheduling Problem

The scheduling setup is unchanged from Chapter 3 except for the following modification to the feedback mechanism: the scheduled user, based on measurements of

the signal strength of the received data packet, obtains information on the state of the channel and sends this back to the scheduler. We call this feedback as F_i with $i \in \{1, 2, 3\}$. Thus the feedback is not 1-bit or ARQ any more. Also, this feedback is received at the scheduler instantaneously. As before, the feedback information, along with the label of the slot 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 reward of the system. Details are discussed next.

4.2.3 Formal Problem Definition

Similar to previous chapters, the scheduling problem is modeled as a POMDP. The entities we use in the analysis are explained next. Although, many of these entities are retained from previous chapters, to avoid any ambiguity, we describe them here.

Horizon: Similar to Chapter 3, we consider the finite horizon scenario. The horizon is denoted by m.

Action a_k: Indicates the index of the user (1 or 2) scheduled in slot k.

Belief vector of user i *at the* k^{th} *slot* $\pi_{k,i}$: Element $\pi_{k,i}(j)$ denotes the probability that the channel of user $i \in \{1, 2\}$ in the k^{th} control interval is in state $j \in \{1, 2, 3\}$, given all the past information about that channel. If F_j was received from user i, $l + 1$ slots earlier with $l \in 0, 1, 2, \ldots$, then the belief vector in the current interval k is given by $\pi_{k,i} = [p_{j1}\ p_{j2}\ p_{j3}]P^l$. We will henceforth represent the vector $[p_{j1}\ p_{j2}\ p_{j3}]$ by $\mathbf{p}_j$. If user i is not scheduled in slot k, then the belief vector of this user evolves to the next interval as follows: $\pi_{k-1,i} = \pi_{k,i}P$.

Stationary Scheduling Policy $\mathfrak{A}_k$: A stationary scheduling policy $\mathfrak{A}_k$ in slot k is a stationary mapping from the belief vectors and the slot index to an action as follows:

$$\mathfrak{A}_k : (\pi_{k,1}, \pi_{k,2}) \rightarrow a_k \quad \forall k \geq 1.$$

Reward Structure: In any slot k, a reward of α_i is accrued when the scheduled user sends back F_i. Let state 1 be defined such that no reward is accrued when an user in state 1 is scheduled, i.e., $\alpha_1 = 0$. This assumption can be satisfied by letting state 1 represent the channel strengths that do not allow any useful data transfer. Since state 3 represents channel strengths that are better than those represented by state 2, we have $\alpha_3 \geq \alpha_2$. Throughout this chapter, we will assume $\alpha_3 = 1$ without loss of generality.

Net Expected Reward in the slot m, V_m: With the belief vectors, $\pi_{m,1}$, $\pi_{m,2}$ and the scheduling policy, $\{\mathfrak{A}_k\}_{k \leq m}$, fixed, the net expected reward, V_m, is the sum of the reward, $R_m(\pi_{m,a_m}, a_m)$, expected in the current slot m and $\mathrm{E}[V_{m-1}]$, the net reward expected in the future slots conditioned on the belief vectors and the scheduling decision in the current slot. Formally,

$$\begin{aligned} V_m(\pi_{m,1}, \pi_{m,2}, \{\mathfrak{A}_k\}_{k \leq m}) &= R_m(\pi_{m,a_m}, a_m) \\ &\quad + \mathrm{E}[V_{m-1}(\pi_{m-1,1}, \pi_{m-1,2}, \{\mathfrak{A}_k\}_{k \leq m-1}) | \pi_{m,1}, \pi_{m,1}, a_m], \end{aligned}$$

where the expectation is over the belief vectors $\pi_{m-1,1}, \pi_{m-1,2}$. With [10] $\alpha = [\alpha_1 \; \alpha_2 \; \alpha_3]^T$, the expected current reward can be written as

$$R_m(\pi_{m,a_m}, a_m) = \pi_{m,a_m} \alpha.$$

Note that if a_m was observed to be in state i in the previous interval then $\pi_{m,a_m} = \mathbf{p}_i$ and $R_m(\pi_{m,a_m}, a_m) = \mathbf{p}_i \alpha$.

[10] $\mathbf{x}^T$ indicates the transpose of vector $\mathbf{x}$.

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

$$\eta_{\text{sum}}(\{\mathfrak{A}_k\}_{k\geq 1}) \;=\; \lim_{m\to\infty} \frac{V_m(\pi_{ss}, \pi_{ss}, \{\mathfrak{A}_k\}_{k\geq 1})}{m}, \tag{4.3}$$

where π_{ss} is the steady state probability vector of the underlying Markov channels.

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

$$\{\mathfrak{A}_k^*\}_{k\geq 1} \;=\; \arg \max_{\{\mathfrak{A}_k\}_{k\geq 1}} \eta_{\text{sum}}(\{\mathfrak{A}_k\}_{k\geq 1}). \tag{4.4}$$

4.3 Structure of the Greedy Policy

Recall from previous chapter, the definition of the greedy policy:

$$\begin{aligned} \widehat{\mathfrak{A}}_k : (\pi_{k,1}, \pi_{k,2}) \to a_k \;&=\; \arg\max_{a_k} R_k(\pi_{k,a_k}, a_k) \\ &=\; \arg\max_{i} \pi_{k,i}\alpha \quad \forall k \geq 1. \end{aligned}$$

We proceed to derive the implementation structure of the greedy policy. First, we record a few preparatory results in Lemma 6 to Lemma 8. Proofs can be found in the appendix.

Lemma 6. *For any $k \geq 0$, the immediate reward expected by scheduling an user that was observed $k+1$ slots earlier, to be in state 2, lies between the rewards corresponding to states 3 and 1, i.e,*

$$\mathbf{p}_1 P^k \alpha \leq \mathbf{p}_2 P^k \alpha \leq \mathbf{p}_3 P^k \alpha, \forall k \in 0, 1, 2, \ldots \tag{4.5}$$

Lemma 7. *The immediate reward expected by scheduling an user that was observed, $k+1$ control intervals earlier, to be in state 3, monotonically decreases to $\pi_{ss}\alpha$ as k*

increases from $0 \rightarrow \infty$*, i.e.,*

$$\begin{aligned} \mathbf{p}_3 P^{k+1}\alpha &\leq \mathbf{p}_3 P^k \alpha, \ \forall k \in 0, 1, 2, \ldots \\ \mathbf{p}_3 \lim_{k\rightarrow\infty} P^k \alpha &= \pi_{ss}\alpha \end{aligned} \tag{4.6}$$

Note that $\pi_{ss}\alpha$ *is the immediate reward expected when no past information about the user is available or when the belief vector of the user equals the steady state vector,* π_{ss}*.*

Lemma 8. *The immediate reward expected by scheduling an user that was observed,* $k+1$ *control intervals earlier, to be in state 1, monotonically increases to* $\pi_{ss}\alpha$ *as* k *increases from* $0 \rightarrow \infty$*, i.e.,*

$$\begin{aligned} \mathbf{p}_1 P^{k+1}\alpha &\geq \mathbf{p}_1 P^k \alpha, \ \forall k \in 0, 1, 2, \ldots \\ \mathbf{p}_1 \lim_{k\rightarrow\infty} P^k \alpha &= \pi_{ss}\alpha \end{aligned} \tag{4.7}$$

Note that, from the above lemmas, we have

$$\mathbf{p}_2 \lim_{k\rightarrow\infty} P^k \alpha = \pi_{ss}\alpha. \tag{4.8}$$

In all the above results, the immediate reward approaches $\pi_{ss}\alpha$ as the time since the last observation of the user increases. This is because, in the underlying first order Markov chain, the dependency between the states in two slots (memory) diminishes as the time gap between the slots increases. These lemmas are instrumental in obtaining the algorithm for implementing the greedy policy, that will be summarized soon. We first identify two types of system based on the property of the P matrix and the reward values.

- Type A system: when $\mathbf{p}_2\alpha \geq \pi_{ss}\alpha$

- Type B system: when $\mathbf{p}_2\alpha < \pi_{ss}\alpha$

The implementation algorithm for the greedy policy significantly changes depending on the type of the system.

Proposition 10. *When the system is type A, the greedy policy is implemented as follows*

- *If feedback F_3 or F_2 was received from the user scheduled in the previous control interval (identified as user s), reschedule the user in the current slot.*
- *Schedule the other user (identified as user u) if feedback F_1 was received.*

Proof. Referring to Fig. 4.1, when F_3 was received from user s, the expected reward

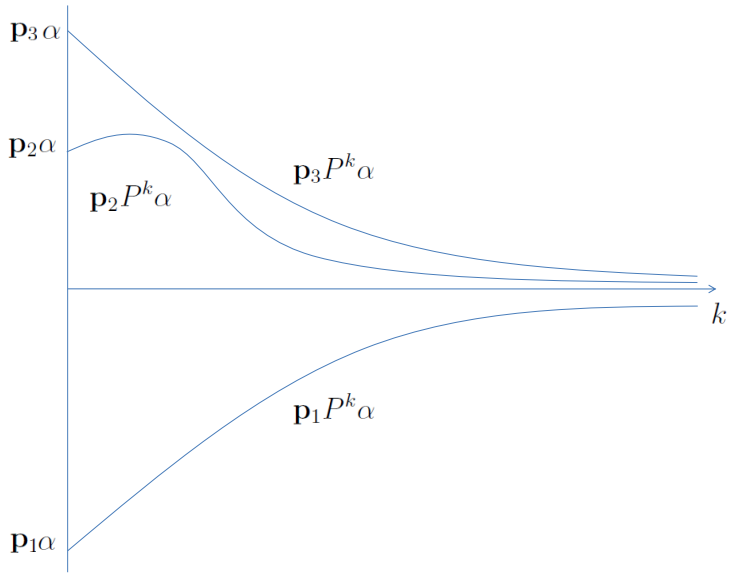

Figure 4.1: Type A system.

if s is scheduled again is given by $p_3\alpha$. The expected reward if u is scheduled is a

point on one of the three curves (for $k > 0$) in the figure. Note that $p_3\alpha$ is greater than any point (the y-dimension) on any of the curves, thus establishing 'retain the schedule if F_3 is received' policy. This result essentially stems from the following facts: 1) Higher reward ($\alpha_3 = 1$) is accrued when the scheduled user happens to be in state 3 than in other states. 2) The Markov channel is positively correlated in time ($p_{ii} \geq p_{ji}$ if $i \neq j$).

Similarly when F_1 was received from user s, the expected reward if s is scheduled again is given by $p_1\alpha$ which is less than any other point on the three curves, thus establishing 'switch if F_1 is received' policy.

When F_2 is received, assuming the greedy policy was implemented so far since the beginning of the scheduling process, the reward expected if u is scheduled lies on the lower curve $\mathbf{p}_1 P^k \alpha$ for $k > 0$. This is because the first time (since the beginning of the scheduling process) a F_2 is received (call this interval m_0), if greedy policy was implemented so far, user u (the waiting user) would not have given F_3 when it was dropped and since this is the first time F_2 is observed by the scheduler, u would not have sent F_2 either, when it was dropped. Therefore u must have sent F_1 the last time it was scheduled (and hence dropped). Thus the reward expected if u is scheduled now (at m_0) falls on the bottom curve leading to retaining of user s (since $\mathbf{p}_2\alpha \geq \mathbf{p}_1 P^k \alpha$ for any $k \geq 0$). In the next instance of F_2 reception, the same logic holds (as long as greedy policy is implemented all along until this instance) and so on for subsequent instances of F_2. Note that the condition *greedy must be implemented since the beginning until 'now'* is quite natural given our interest in implementing the policy in the current interval. Thus there is no loss of generality here.

These arguments establish the proposition. □

An illustration of the implementation of the greedy policy in the type A system is provided in Fig. 4.2.

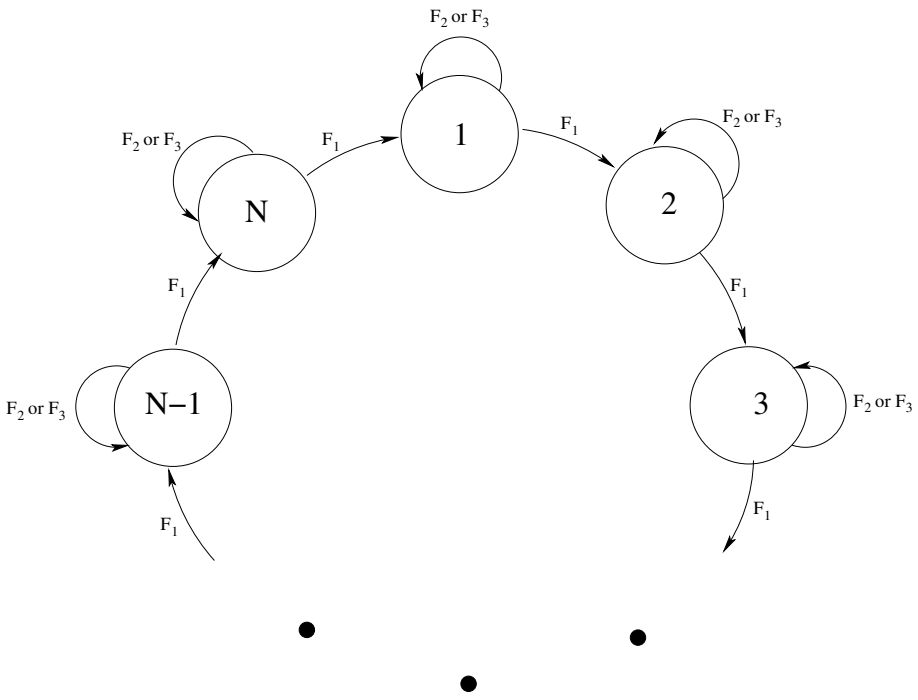

Figure 4.2: Round-robin implementation of the greedy policy in the type A system.

Proposition 11. *When the system is type B, the greedy policy is implemented as follows*

- *If feedback F_3 was received from the user scheduled in the previous control interval (call it user s), reschedule the user in the current slot.*
- *If feedback F_1 was received, schedule the other user.*
- *If feedback F_2 was received, calculate the expected immediate reward if the other user (identified as user u) is scheduled in the current interval (identified as m)*

as follows: $\pi_{m,u}\alpha$ *where* $\pi_{m,u}$ *is the belief vector of user* u *in the current control interval* m*. Now, schedule user* s *is* $\mathbf{p}_2\alpha \geq \pi_{m,u}\alpha$*. Otherwise, schedule user* u*.*

Proof. Refer to Fig. 4.3. The argument for F_3 and F_1 are the same as in the previous

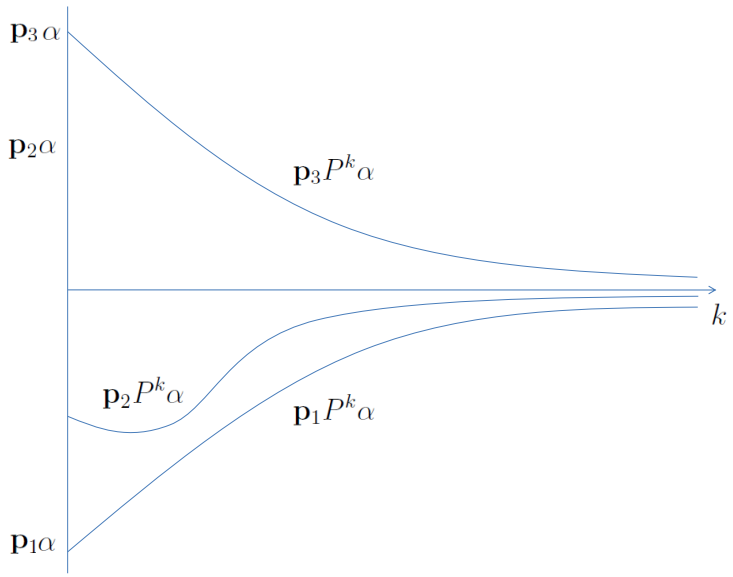

Figure 4.3: Type B system.

case. When F_2 is received, as seen from the Fig. 4.3, the waiting user u could have an expected reward greater than that of s if u had been dropped due to F_1 at least k_0 intervals earlier or if $\mathbf{p}_2P^k\alpha$ does not monotonically increase to $\pi_{ss}\alpha$ (Fig. 4.3 shows such a situation). Thus it is necessary to explicitly calculate the expected reward of user u before making a greedy decision. □

An illustration of the implementation of the greedy policy in the type B system is provided in Fig. 4.4.

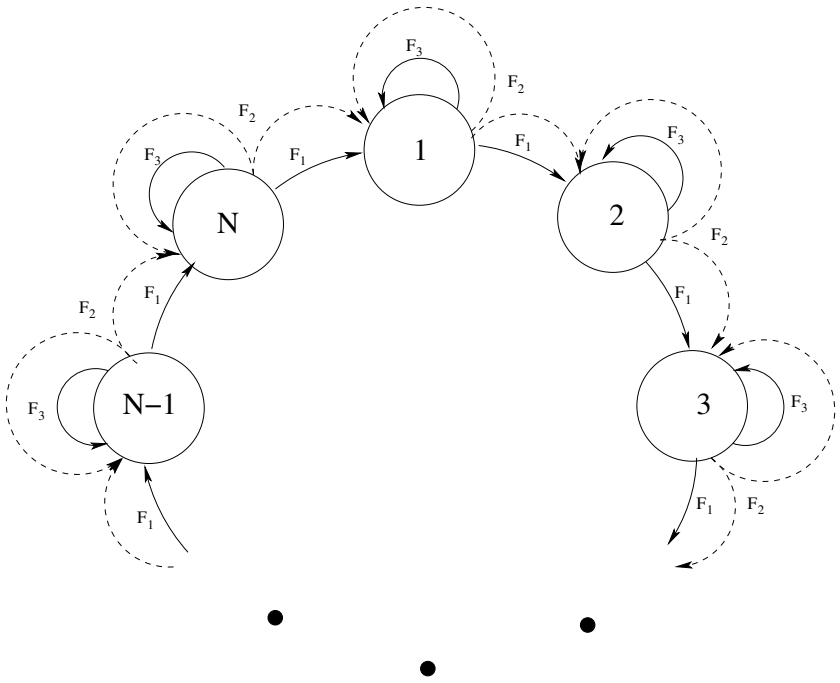

Figure 4.4: Implementation of the greedy policy in the type B system.

Note that the results in Lemma 6-8 and hence the implementation structure of the greedy policy in Propositions 10-11 hold even when $\alpha_1 > 0$ as long as $\alpha_1 \leq \alpha_2 \leq \alpha_3$.

4.4 Comparison with the Genie-aided System

From the discussion in the previous chapter, when the downlink has two users and the channels are modeled by *two* states, we have the following interpretation: when the feedback is instantaneous, if the user scheduled (user s) in the previous slot was observed to be in the best state, the scheduler retains the schedule (and hence accrues the best possible reward) since there is nothing more to gain by scheduling to the other user, while a loss is possible if the other user was in the worst state. Similarly, if user s was observed to be in the worst state, the scheduler switches to the other,

since there is nothing more to lose by scheduling to the other user (as compared to scheduling s again), while a gain is possible if the other user was in the best state. Thus the two user, two state system is equivalent, in performance, to a genie-aided system where the scheduler learns about the states of both the users at the end of every interval.

This equivalence does not hold in the three state system considered in this chapter. The *nothing more to gain* argument works when s was observed to be in state 3 and the *nothing more to lose* argument works when s was observed to be in state 1. However, when s was observed to be in state 2, i.e., when F_2 was received, by scheduling to the other user (user u), the scheduler may either gain (if u was in state 3) or lose (if u was in state 3) as compared to when it schedules s again. Thus with information about the state of the other user, there is definitely a room for improvement. Thus the three state (in general, more than two states) system is not equivalent to the genie-aided system. Note that, the genie aided system can be redefined as follows: the scheduler learns about the state of both the users if and only if s was observed to be in state 2. We see from the discussion so far that this modified definition does not impart any performance loss in the genie-aided system.

From the preceding discussion, it can be seen that the original three user system approaches the genie-aided system under any of the following conditions:

- $\mathbf{p}_2\alpha = \mathbf{p}_3\alpha$. Thus on receiving F_2 from user s, nothing more can be gained by scheduling the other user u (while a loss is possible on switching). Hence, s is rescheduled. Thus there is no need to learn the previous slot state of u.

- $\mathbf{p}_2\alpha = \mathbf{p}_1\alpha$. Thus on receiving F_2 from user s, nothing more can be lost by scheduling the other user u (while a gain is possible on switching). Hence, u is scheduled. Again, there is no need to learn the previous slot state of u.

With mathematical analysis, it can be seen that condition 1 is achieved if and only if $\alpha_2 = \alpha_3$ and $p_{21} = p_{31}$. While condition 2 can be achieved if and only if $\alpha_2 = \alpha_3$ and $p_{11} = p_{21}$. When the first set of conditions is satisfied, it can be seen that the states 2 and 3 can be merged at a very generic level (not specific to the type of information used for scheduling) with the reduced transition matrix given as below:

$$\begin{bmatrix} p_{11} & p_{12} + p_{13} \\ p_{21} & p_{22} + p_{23} \end{bmatrix} \tag{4.9}$$

where row 1 and column 1 corresponds to state 1 and row 2 and column 2 corresponds to the merged state. Thus the channel is effectively modeled by a two-state Markov chain thus explaining the equivalence with the genie-aided system.

However, it is interesting to note that, when the second set of conditions is satisfied, such a merger is not possible between states 1 and 2 since we still have $p_{13} \leq p_{23}$ making them different in their relationship with state 3. However, in the context of the channel feedback based scheduling problem, they are synonymous and render the original system equivalent to the genie-aided system.

4.5 Bounds on the System Sum Capacity

Proposition 12. *For the type A system, a lower bound to the sum capacity, $S_{LB,A}$, is given as*

$$S_{LB,A} \;\geq\; \mathbf{p}_2\alpha - \pi_{ss}^2(1)(\mathbf{p}_2\alpha - \mathbf{p}_1\alpha) \tag{4.10}$$

where $\pi_{ss}(1)$ is the steady state probability that the state of the user is 1.

This bound is obtained by replacing expected reward given F_3, i.e., $\mathbf{p}_3\alpha$ with $\mathbf{p}_2\alpha$ in the sum reward evaluation of the greedy policy. Thus this is in fact a lower bound to the greedy policy. Note that $S_{LB,A}$ decreases as the steady state probability of the less rewarding state 1 ($\pi_{ss}(1)$) increases. Also notice that as $\mathbf{p}_1\alpha \rightarrow \mathbf{p}_2\alpha$, $S_{LB} \rightarrow \mathbf{p}_2\alpha$. This is expected in light of the approach we used in obtaining S_{LB}, since the only reward that we accrue in any slot is now $\mathbf{p}_2\alpha$. Also, the bound approaches the system sum reward capacity when states 2 and 3 become increasingly synonymous. This happens as $\alpha_2 \rightarrow \alpha_3$ and $p_{31} \rightarrow p_{21}$. The last statement comes from our discussion in the previous section, on the equivalence with the genie-aided system.

Proposition 13. *For the type B system, a lower bound to the sum reward capacity is given as*

$$S_{LB,B} = (2\pi_{ss}(3) - \pi_{ss}^2(3))\mathbf{p}_3\alpha + (1 - \pi_{ss}(3))^2\mathbf{p}_3\alpha \tag{4.11}$$

The proof proceeds as follows: In any slot the expected immediate reward after a feedback F_2 is received in the previous interval is replaced by the reward that would be expected if the other (not scheduled in the previous interval) user were scheduled. Note that, by the implementation structure of the greedy policy, this latter reward is $\leq$ the reward corresponding to the greedy choice[11]. Next we replace $\mathbf{p}_2\alpha$ with $\mathbf{p}_1\alpha$ giving the sum reward capacity lower bound.

Note that $S_{LB,B}$ is the same as a two user system that accrues reward $\mathbf{p}_3\alpha$ if at least one of the users are in state 3 and reward $\mathbf{p}_1\alpha$ if none of them are in state 3. This interpretation is strikingly similar to the interpretation we made in the two-state tow user problem in our preliminary research. However, note that the present

[11]The replacement is only with respect to the accrued reward in the sum reward expression, while the actual schedule decision is always maintained as greedy, so as not to disturb the initial conditions of the problem for the future intervals.

interpretation does not yield to the case when the state of both users are available. For instance, if none of the users is in state 3 and at least one of them is in state 2, then, ideally, if the states of both the users are known, a reward of $\mathbf{p}_2\alpha$ must be accrued instead of $\mathbf{p}_1\alpha$. This demonstrates the loss in performance due to lack of knowledge of both user states, thus differentiating the 3-state system from the 2-state system.

Proposition 14. *An upper bound to the system sum reward capacity is given as*

$$S_{UB} = (2\pi_{ss}(3) - \pi_{ss}^2(3))\mathbf{p}_3\alpha + (2\pi_{ss}(1)\pi_{ss}(2) + \pi_{ss}^2(2))\mathbf{p}_2\alpha + \pi_{ss}^2(1)\mathbf{p}_1\alpha$$

The bound is actually the sum reward capacity of the genie-aided system. Here if at least one of the users was in state 3 in the previous interval, the greedy policy schedules that user and accrues a reward $\mathbf{p}_3\alpha$. If none of the users were in state 3 but at least one of them in state 2, that user is scheduled and a reward of $\mathbf{p}_2\alpha$ is accrued. If both the users were in state 1, a reward of $\mathbf{p}_1\alpha$ is accrued.

4.6 On the Optimality of the Greedy Policy

We proceed by introducing the following properties of the P matrix. The proofs are tedious and hence moved to the appendix.

Lemma 9. *When* $\mathbf{p}_2P[001]^T \leq p_{23}$ *(condition (A)), then* $\mathbf{p}_2P^{k+1}[001]^T \leq \mathbf{p}_2P^k[001]^T$ $\forall k \geq 0$. *Also the steady state element* $\pi_{ss}(3) \leq p_{23}$ *and* $p_2P^k[001]^T$ *monotonically decreases to* $\pi_{ss}(3)$ *as* $k \to \infty$. *(A) is also a necessary condition for the preceding statement to hold.*

Lemma 10. *Under (A) from previous lemma,* $\mathbf{p}_1 P^k [001]^T$ *monotonically increases to* $\pi_{ss}(3)$ *as* $k \to \infty$*, i.e,* $\mathbf{p}_1 P^{k+1} [001]^T \geq \mathbf{p}_1 P^k [001]^T \forall k \in 0, 1, 2, \ldots$ *and* $\mathbf{p}_1 \lim_{k\to\infty} P^k [001]^T = \pi_{ss}(3) \leq p_{23}$

Using the preceding properties of the transition matrix, we show that, under certain conditions, the greedy policy is optimal among a special class of policies. We record this below. The proof is provided in the appendix.

Proposition 15. *When* $p_{12} = p_{22} = p_{32}$ *and* $p_{23}p_{31} \geq p_{21}p_{13}$*, greedy policy is optimal among the policies that retain the schedule when feedback* F_3 *is received.*

Note that, in light of the positive correlation property of the Markov chain, it seems counterintuitive that the globally optimal policy would reject an user that was in the best state possible in the previous slot, thereby (from Proposition 15) strongly suggesting the global optimality of the greedy policy.

4.7 Summary

We considered joint channel estimation - opportunistic scheduling in a Markov-modeled two-user downlink system when the Markov state space is three and studied the optimality properties of the greedy policy. We showed that the greedy policy that is optimal when the channel state space is two (Chapter 3) is not necessarily optimal in an increased state space. Specifically, we show that the equivalence with the genie-aided system that was observed in the two-state case is upset with the introduction of the third Markov channel state. Apart from this analysis, we obtained implementation structure of the greedy policy and obtained bounds on the system sum capacity. For specific conditions on the transition matrix, we showed that the greedy policy is optimal among a specific class of policies.

CHAPTER 5

OPPORTUNISTIC SCHEDULING USING ARQ FEEDBACK IN MULTI-CELLULAR DOWNLINK

5.1 Introduction

In Chapters 3 and 4, we studied joint channel estimation - opportunistic scheduling in a *single*-cellular downlink. As a natural extension of this analysis, in this chapter, we study the joint scheduling problem in multi-cellular downlinks. In a multi-cellular downlink, transmission in a cell interferes with transmissions in the adjacent cells. It follows that the channel state of any user in a cell is a function of the transmissions and schedule decisions in the adjacent cells, effectively imparting a convolved dependence between the scheduling choices in neighboring cells. We now face the following question:

How do we exploit the channel memory and the ARQ feedback mechanism for opportunistic scheduling in a multi-cellular environment ?

We address this problem by following a two layered approach: A well established inter-cell interference (ICI) control mechanism is adopted and assumed to be in place. On top of this layer we optimize ARQ based opportunistic scheduling across the cells.

We now proceed to introduce our choice of the ICI mechanism after a short literature survey on the topic.

Traditionally, ICI is controlled by staggering the transmissions in adjacent cells across orthogonal frequency bands and reusing these bands in geographically far-apart cells. This is the well known frequency reuse based ICI control mechanism [46] that is prevalent in narrowband systems, such as the GSM. Other ICI control mechanisms have also been studied. In [47], a capture division packet access (CDPA) mechanism is proposed. Here, users are allowed to transmit on the same carrier in adjacent cells, i.e., no frequency reuse based ICI control is deployed. The effect of interference is quantified by a capture probability defined as the likelihood of successful transmission under ICI. Upon collision, a retransmission is performed. The authors demonstrate that CDPA outperforms traditional TDMA based strategies under certain operating conditions. Notice that, in the preceding scenario, users at the periphery of the cell suffer from low signal to interference ratio and hence low capture probability compared to the users near the base station. This is the classic *near-far* effect. If this is not addressed properly, under QoS requirements on fairness across users, the far users will act as a bottleneck thus bringing down the overall system utility. Taking note of this crucial phenomenon, the authors in [48] proposed a novel reduced power channel reuse (RPCR) scheme that aims to equalize the capture probabilities of the near and far users. By formally classifying the users into two groups: near and far (based on a generic "distance" metric that need not be a function of the geometric distance), RPCR works as follows: If, in a carrier, a far user is scheduled in a cell, the power of transmission in the same carrier in the adjacent cell is deliberately reduced. This power reduction naturally limits transmission to near users in the adjacent cell.

Thus, at any time, in any carrier, cell 1 and cell 2 transmits to users belonging to complementary groups with different power levels (full power for the far user). The authors of [48] formulated and studied the optimal channel selection policy that assigns users to the near/far groups. They showed that the RPCR scheme is superior in performance to other ICI control mechanisms in terms of sum throughput under uniform fairness constraints. A similar protocol called 'cell breathing' was shown to provide system level gains in [49, 50].

Encouraged by the positive results associated with the RPCR and cell breathing, we adopt cell breathing as our ICI control mechanism. If the channel of the users are time-variant, it is readily seen that, without violating the cell breathing protocol, the performance of the system can be improved by opportunistic multiuser scheduling with coordination across cells. We address this joint opportunistic scheduling problem in a two-cell system in this work. It is worth noting that the scheduling analysis for the two-cell system readily extends to a multi-cell configuration with the use of six-directional antennae [46] at the base stations. Each cell can now be divided into six regions and the joint scheduling analysis in this work can be applied in each region independently. This is illustrated in Fig. 5.1.

By demonstrating that the channel can still be modeled by $i.i.d$ two-state Markov chains, like in the single cell case, we study the ARQ based joint opportunistic scheduling scheme in the two-cell system, under two scenarios: (a) when the cooperation between the cells is asymmetric (b) when there is symmetric cooperation. In the asymmetric case, we show that the optimal scheduling policy is a variant of the greedy policy. In the symmetric scenario, however, a direct optimality analysis of the scheduling problem appears intractable. We therefore establish a link between

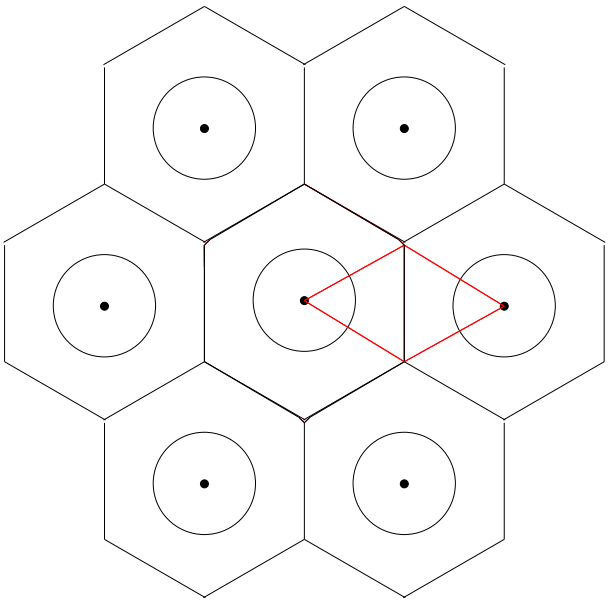

Figure 5.1: Multi-cell extension: with six directional antennae at the base stations, each cell can be split into six regions and the two-cell joint scheduling can be performed on these regions independently. One such region is highlighted.

the scheduling problem and *Restless Multiarmed Bandit (RMAB) processes*. We introduce the notion of *Whittle's indexability* from the RMAB theory and perform an indexability analysis for the system at hand. Based on this analysis, we propose an index policy and demonstrate, via numerical results, the near-optimality of the proposed policy. We describe the problem setup next.

5.2 Problem Setup

5.2.1 Channel Model

Consider a two-cell system. Consistent with [48, 49], within each cell, we cluster users into near and far users. We use geometric distance between the users and their respective base stations as the metric for this classification. Denote by n_i, f_i, the set

of near and far users, respectively, in cell $i \in \{1, 2\}$. A user in a group is denoted by the label of the group for notational simplicity. Let the distance between base station i and user j (in any cell) be d_{ij}. By way of the two level clustering we assume, d_{if_i} characterizes all far users in cell i. Likewise d_{in_i} characterizes all near users in cell i. Let N_i, F_i, be the number of near and far users in cell i, respectively. Denote the normalized (with respect to attenuation loss) fading coefficient of the link between base station i and user j (in any cell) as h_{ij}. We assume h_{ij} are *i.i.d.* Consider cell 1 as the primary cell and cell 2 the interfering cell. If a far user f is[12] served in the primary cell with power P_f and if the interfering base station is transmitting at power P_{I_f} (I_f indicates interference to the far user in the primary cell) then the SINR at this user is given as below.

$$\text{SINR}_f = \frac{\frac{P_f}{d_{1f}^{\alpha}}|h_{1f}|^2}{N_0 + \frac{P_{I_f}}{d_{2f}^{\alpha}}|h_{2f}|^2}, \tag{5.1}$$

where N_0 indicates the variance of the additive noise. Here, we have used the attenuation model from [51] with $\alpha \geq 2$ being the attenuation coefficient. Likewise, if a near user is served in the primary cell with the interfering base station power being P_{I_n}, the SINR is given by

$$\text{SINR}_n = \frac{\frac{P_n}{d_{1n}^{\alpha}}|h_{1n}|^2}{N_0 + \frac{P_{I_n}}{d_{2n}^{\alpha}}|h_{2n}|^2}. \tag{5.2}$$

An illustration of these two scenarios is provided in Fig. 5.2.

Consistent with the assumed two level clustering, the two base stations are each allowed to choose one of two power levels, i.e., $P_f, P_n, P_{I_f}, P_{I_n} \in \{P_1, P_2\}$ with $P_2 < P_1$. By observation, since $d_{1f} > d_{1n}$, the average SINR of the far and near users can be equalized if $P_{I_f} < P_{I_n}$ and $P_f > P_n$. This, along with the constraint on the

[12]We have dropped the suffix 1 as the context is clear.

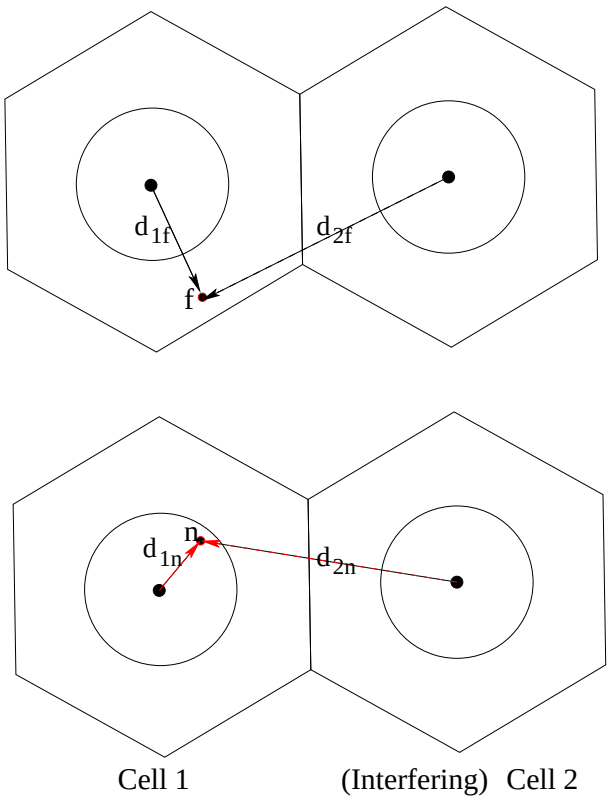

Figure 5.2: Illustration showing transmissions and interference caused when a far user and a near user are served (at different times).

alphabet size of the power levels, leads to the cell breathing rule [48–50]: *A far user is served with power P_1 and a near user with power $P_2 < P_1$. Whenever a far user is scheduled in a cell, a near user is scheduled in the adjacent cell and vice versa.*

Since the links between the base stations and users h_{ij} are *i.i.d*, with the SINR values equalized under cell breathing, we have the following: SINR_{f_1}, SINR_{n_1}, SINR_{f_2}, SINR_{n_2} are *i.i.d.* Similar to our previous system models, we model the fading coefficients with memory, i.e., with two-state Markov chains. Under this model, we see that, under cell breathing, the SINR channel (henceforth, simply the 'channel') of each user can also be modeled using an *i.i.d* two-state Markov chain. As before, the channel of each user remains fixed over a time 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

$$P = \begin{bmatrix} p & q \\ r & s \end{bmatrix}, \tag{5.3}$$

with p and r as defined in previous chapters. Also, consistent with previous chapters, we assume positively correlated Markov channels, i.e., $p > r$.

5.2.2 Scheduling Problem

The base stations are the central controllers that control each transmission to the users within their respective cells in each slot. In particular, in each slot, each base station schedules the transmission of the head of line packet of exactly one user (a data queue is maintained for each user to collect the data meant for that user), while maintaining the cell breathing protocol: That is, in any cell, if a far user is scheduled, transmission takes place at full power P_1, while, for a near user, the lower power

$P_2 < P_1$ is used. Furthermore, a traditional ARQ based transmission is deployed in each cell. That is, 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., it is not successfully decoded by the user (as occurs when the channel is in the OFF state), a NACK is reported by the user at the end of the slot, and the packet is retained at the head of the queue for retransmission at a later time. If the packet does go through (i.e., the ON state), an ACK is reported and the packet is removed from the queue. The ARQ feedback is assumed to be transmitted over a dedicated error-free channel. At the end of the slot, the base stations of neighboring cells share their ARQ information. Thus each base station has all channel information available to its neighbors, hence facilitating joint scheduling among the base stations. An illustration of the two-cell cooperative scheduling setup is provided in Fig. 5.3. The performance metric that the base stations aim to maximize is a discounted reward over an infinite horizon.

Note that, by considering every legitimate near-far and far-near pair as a single cumulative user, the two-cell scheduling model fits into the general scheduling model discussed in Chapter 1. This is illustrated in Fig. 5.4. We now formally define the problem.

5.2.3 Formal Problem Definition

We now introduce the terms/entities that we use in this study.

Horizon: We consider the infinite horizon scenario. For the sake of the proofs, finite horizons are also considered with slot k indicating that there are $k-1$ more slots until the horizon.

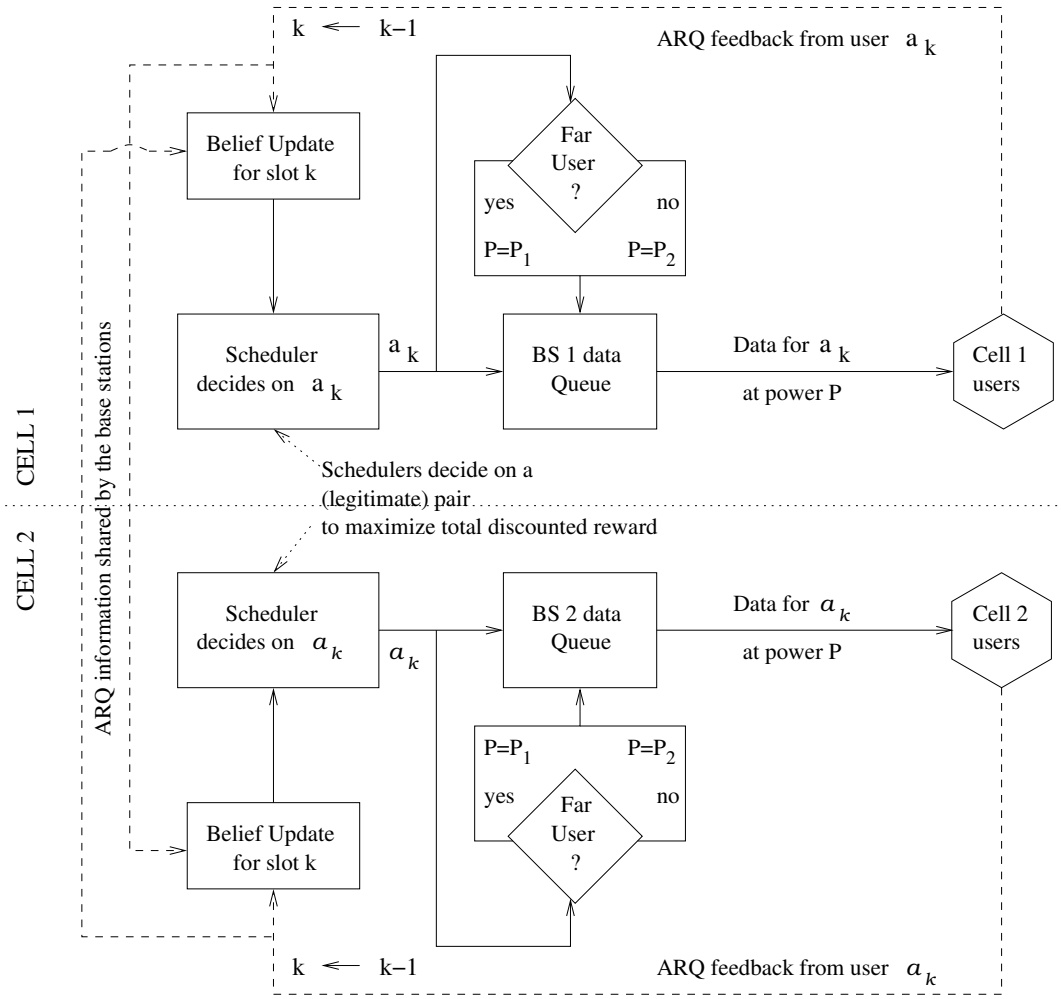

Figure 5.3: Illustration of the two-cell cooperative scheduling setup. By sharing the ARQ feedback in each slot, the base stations maintain the same information on the belief values corresponding to all the users. Thus, without further interaction, the base stations schedule a legitimate pair of users.

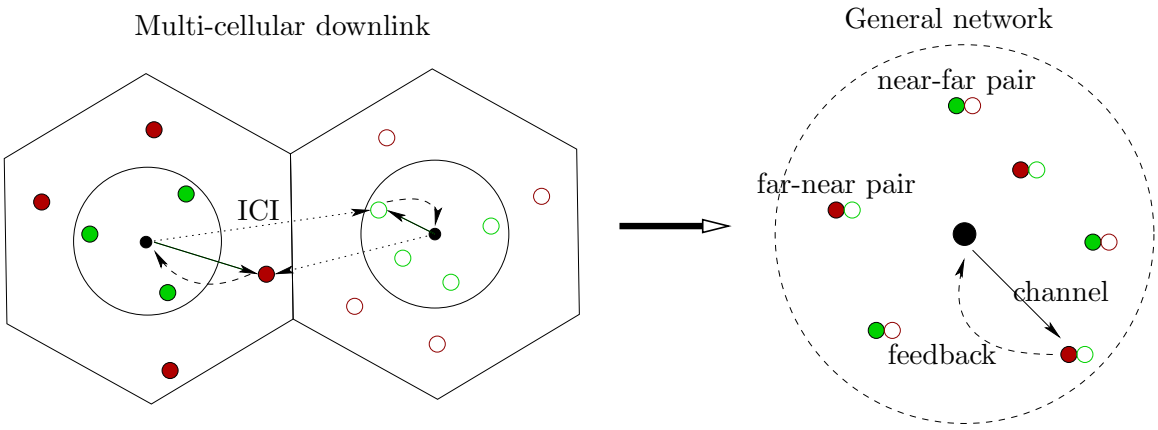

Figure 5.4: The two-cell scheduling model as a special case of the general one-to-many scheduling model.

Action (a_k, a_k): Indicates the indices of the user pair scheduled in cells 1 and 2 in time slot k. With cell breathing in place, we have the following constraint: $(\mathsf{a}_k, a_k) \in \{(n_1, f_2), (f_1, n_2)\}$. We denote this admissible set by $\mathcal{B}$.

Belief values at the k^{th} time slot: Denote by $\pi_k^{n_c}, \pi_k^{f_c}$, with $c \in \{1, 2\}$, the vectors of the belief values (the probability of having an ON state) of the users in group n_c and f_c, respectively, in slot k. Let F_k^c indicate the ARQ feedback received at the end of slot k from cell c. We denote an ACK by 1 and a NACK by 0. The belief values of users in group n_1 evolve as below:

$$\pi_{k-1}^{n_1}(i) = \begin{cases} p, & \text{if } i = \mathsf{a}_k, F_k^1 = 1 \\ r, & \text{if } i = \mathsf{a}_k, F_k^1 = 0 \\ p\pi_k^{n_1}(i) + r(1 - \pi_k^{n_1}(i), & \text{if } i \neq \mathsf{a}_k. \end{cases} \tag{5.4}$$

where the first case indicates that, in cell 1, user i from the near group is scheduled in slot k and an ACK feedback was received. Thus, according to the Markov chain statistics, $\pi_{k-1}^{n_1}(i) = p$. 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 slot k and hence the cell 1 base station must estimate the belief value at the current slot from that at the previous slot and the Markov chain statistics. A similar evolution holds for the users in other groups.

Stationary Scheduling Policy $\mathfrak{A}$: A stationary scheduling policy $\mathfrak{A}$ is a mapping, in any time slot, from the belief values to an action as follows:

$$\mathfrak{A} : (\{\pi^{n_1}, \pi^{f_1}, \pi^{n_2}, \pi^{f_2}\}) \rightarrow (\mathsf{a}, a) \in \mathcal{B}.$$

Reward Structure: In any time slot k, in cell c, a reward of 1 is accrued when the transmission in cell c is successful, i.e, when $F_k^c = 1$, and no reward is accrued otherwise. The total immediate reward in any slot is simply the sum of the immediate rewards accrued by cells 1 and 2.

Expected Discounted Reward under Policy $\mathfrak{A}$: We consider the discounted reward over an infinite horizon. Under policy $\mathfrak{A}$ and belief values $\pi^{n_1}, \pi^{f_1}, \pi^{n_2}, \pi^{f_2}$, the expected discounted reward over an infinite horizon is given by

$$\begin{aligned} & V(\pi^{n_1}, \pi^{f_1}, \pi^{n_2}, \pi^{f_2}, \mathfrak{A}) \\ &= R(\pi^{n_1}, \pi^{f_1}, \pi^{n_2}, \pi^{f_2}, (\mathsf{a}, a)) + \beta \, \mathrm{E}[V(T(\pi^{n_1}), T(\pi^{f_1}), T(\pi^{n_2}), T(\pi^{f_2}), \mathfrak{A})], \end{aligned} \tag{5.5}$$

where the expectation is over the belief values $T(\pi^{n_1}), T(\pi^{f_1}), T(\pi^{n_2}), T(\pi^{f_2})$ and $T(.)$ is the belief evolution operator conditioned on the belief values and the ARQ feedbacks from the previous slot. The discount factor $\beta \in (0, 1)$ gives greater weight to the immediate reward than the future reward, a typical arrangement in infinite horizon

dynamic programming problems. The expected current reward is given by

$$R(\pi^{n_1}, \pi^{f_1}, \pi^{n_2}, \pi^{f_2}, (\mathsf{a}, a)) \;=\; \begin{cases} \pi^{n_1}(\mathsf{a}) + \pi^{f_2}(a), & \text{if } (\mathsf{a}, a) \in (n_1, f_2) \\ \pi^{f_1}(\mathsf{a}) + \pi^{n_2}(a), & \text{if } (\mathsf{a}, a) \in (f_1, n_2). \end{cases} \tag{5.6}$$

Optimal Stationary Policy: A stationary policy that maximizes the total expected discounted reward is optimal. Thus, for any $\{\pi^{n_1}, \pi^{f_1}, \pi^{n_2}, \pi^{f_2}\}$,

$$\mathfrak{A}^* \;=\; \arg\max_{\mathfrak{A}} V(\pi^{n_1}, \pi^{f_1}, \pi^{n_2}, \pi^{f_2}, \mathfrak{A}) \tag{5.7}$$

5.3 Optimal Scheduling under Asymmetric Cooperation between Cells

Consider a system where cell breathing is deployed by the following asymmetric cooperation between the cells: Base station 1 schedules transmission to its users without any regard to the decisions in cell 2, while base station 2 schedules based on the user group choice of base station 1, to conform with the cell breathing protocol. Base station 1 is aware of this compromise made by base station 2 and therefore adopts the two state Markov model for the channels of cell 1 users. Such an asymmetric cooperation can result from scenarios such as (1) Cell 1 covers the heart of a city with higher data rate requirements compared to cell 2, which covers the suburbs, (2) Sharing of ARQ feedback information between the adjacent base stations is not mutual due to, e.g., a partial link failure between the base stations, (3) In the context of game theory, base station 1 being a selfish player and base station 2 being a rule-abiding player.

We first study the optimal scheduling policy in a finite horizon, discounted reward setup. Consider cell 1 under the asymmetric cooperation scenario. Since base station 1 makes scheduling decisions unilaterally, the opportunistic scheduling problem in

cell 1 is the same as that of a single cell system. For a single cell Markov modeled downlink with N users and instantaneous ARQ, it has been established [45] that the greedy policy that maximizes the immediate reward is optimal, with or without reward discounting, for both finite and infinite horizons. Thus base station 1, under asymmetric cooperation, implements the greedy policy within its cell. We denote this policy by $\hat{\mathfrak{A}}$.

We now proceed to study, under asymmetric cooperation, the optimal scheduling policy of cell 2. Let $h > 0$ be the length of the horizon. Fix a realization of the channel states of the users in cell 1 from time h until the horizon. With a fixed scheduling policy in cell 1 (in this case, the greedy policy), we can define a *sporadic*, i.e., non-consecutive (in general), sequence of time instants $\{t_{\mathbf{n}}, t_{\mathbf{n}-1}, \ldots t_1\}$ with $h \geq t_{\mathbf{n}} \geq t_{\mathbf{n}-1}, \ldots t_1 \geq 1$, where a near user is scheduled in cell 1. Note that, for any k such that $2 \leq k \leq \mathbf{n}$, $t_k \neq t_{k-1} + 1$, in general — hence the name sporadic. Now, by definition, at slots $\{h, h-1, \ldots 1\} \setminus \{t_{\mathbf{n}}, t_{\mathbf{n}-1}, \ldots t_1\}$, a far user is scheduled in cell 1. Define $\mathbf{t}_k := \{t_k, t_{k-1}, \ldots t_1\}$ with $k \leq \mathbf{n}$. Note that, in the sporadic time axis $\mathbf{t}_{\mathbf{n}}$, base station 2, in order to maintain cell breathing, schedules far users. Likewise, in $\{h, h-1, \ldots 1\} \setminus \mathbf{t}_{\mathbf{n}}$, base station 2 schedules near users.

Let $\mathfrak{A}^f$ and $\mathfrak{A}^n$ denote the scheduling policies adopted by base station 2 in the sporadic time axes corresponding, respectively, to near and far scheduling decisions in cell 1. Let $\{\hat{\mathfrak{A}}, \mathfrak{A}^f, \mathfrak{A}^n\}$ denote the two-cell scheduling policy with $\hat{\mathfrak{A}}$ indicating the use of greedy policy in cell 1. Defining the *per-cell discounted reward* in cell c as the net discounted reward earned in cell c alone, we introduce the following lemma.

Lemma 11. *Under the asymmetric cooperation assumption, if, for any fixed h, for every realization of $\{\mathbf{n}, \mathbf{t_n}\}$, the scheduling policy $\{\mathfrak{A}^f, \mathfrak{A}^n\}$ maximizes the per-cell discounted reward in cell 2, then $\{\hat{\mathfrak{A}}, \mathfrak{A}^f, \mathfrak{A}^n\}$ is optimal.*

Proof. The lemma is not obvious due to a possible influence of the policies $\mathfrak{A}^f$ and $\mathfrak{A}^n$ on the sporadic time axis $\mathbf{t_n}$, potentially invalidating the realization based argument. Under the asymmetric cooperation assumption, since base station 1 makes near/far scheduling decisions without consulting base station 2 and since the channel states evolve independently at the underlying physical layer[13], $\{\mathbf{n}, \mathbf{t_n}\}$ is independent of the scheduling decisions and observations made in cell 2 within the sporadic time axes and hence is independent of $\mathfrak{A}^f$ and $\mathfrak{A}^n$. This decoupling along with the fact that the greedy policy is optimal in cell 1 establishes the lemma. □

We now proceed to show that the greedy policy is optimal within a realization of the sporadic time axes. Note that the greedy policy was shown to be optimal on a non-sporadic time axis in [45] under instantaneous ARQ. However, in the current case, since the belief values evolve across non-uniform time steps, we need a rigorous optimality proof in the changed setting.

Fix a realization of $\{\mathbf{n}, \mathbf{t_n}\}$ throughout the following analysis. Note that base station 2 schedules to far users in $\mathbf{t_n}$. The net expected reward accrued by base station 2 from $t_{k \leq \mathbf{n}}$ on the time axis $\mathbf{t_n}$ is given as follows. With $\mathfrak{A}_k^f$ indicating the scheduling policy in effect from slot k until the horizon on $\mathbf{t_n}$,

$$V_{t_k}(\pi_{t_k}^{f_2}, \{a_{t_k}, \{\mathfrak{A}_{t_l}^f\}_{k>l>0}\}) \;=\; \pi_{t_k}^{f_2}(a_{t_k}) + \beta_{k-1}\,\mathrm{E}\left[V_{t_{k-1}}\big(\pi_{t_{k-1}}, \{\mathfrak{A}_{t_l}^f\}_{k-1>l>0} | \pi_{t_k}, a_{t_k}\big)\right], \tag{5.8}$$

[13]Note that the inter-cell, intra-cell *base station to user* links are assumed to be statistically independent.

where $\beta_k \triangleq \beta^{h-t_k}$ and a_{t_k} is the far user from cell 2 scheduled in time slot t_k. We now establish the structure of the greedy policy on the sporadic time axis. In any slot t_k, $k < \mathbf{n}$, the belief values of the users are given as follows.

$$\pi_{t_k}^{f_2}(i) = \begin{cases} p, & \text{if } i = a_{t_{k+1}}, F_{t_{k+1}}^2 = 1 \\ r, & \text{if } i = a_{t_{k+1}}, F_{t_{k+1}}^2 = 0 \\ T^{(t_{k+1}-t_k)}(\pi_{t_{k+1}}^{f_2}(i)), & \text{if } i \neq a_{t_{k+1}}. \end{cases} \tag{5.9}$$

Note that for $0 \leq x \leq 1$, $T(x) = x(p-r) + r$. Thus $T(x) \in [r, p]$. Since $T^l(x) = T(T^{l-1}(x))$, by induction, $T^l(x) \in [r, p]$, when $l > 0$. Also $T(x_1) \geq T(x_2)$ if $x_1 \geq x_2$. Thus, by induction, $T^l(x_1) \geq T^l(x_2)$ if $x_1 \geq x_2$. We now introduce the schedule order vector O_{t_k} as the ordered arrangement of the index of the users in decreasing order of $\pi_{t_k}(i)$, i.e.,

$$\begin{aligned} O_{t_k}(1) &= \arg\max_i \pi_k(i) \\ &\vdots \\ O_{t_k}(F_2) &= \arg\min_i \pi_k(i). \end{aligned}$$

From the preceding discussion on the structure of $T(x)$ and the evolution of the belief values, the schedule order vector evolves as below:

$$O_{t_{k-1}} = \begin{cases} [a_{t_k}\ \{O_{t_k} - a_{t_k}\}], & \text{if } f_{t_k} = 1 \\ [\{O_{t_k} - a_{t_k}\}\ a_{t_k}], & \text{if } f_{t_k} = 0, \end{cases} \tag{5.10}$$

The greedy policy, which aims to maximize the immediate reward (i.e., belief value), picks the user at the top of the schedule order vector and thus has a round-robin structure, with a user switch triggered by a NACK, on the sporadic time axis. A similar behavior was seen in the non-sporadic axis when the ARQ is delayed by a deterministic quantity in Chapter 3. We now proceed to show the optimality of the greedy policy on $\mathbf{t_n}$ by first deriving a sufficient condition for optimality.

Consider a slot $t_m, m \leq \mathbf{n}$ with belief values $\pi_{t_m}^{f_2}$ (we will drop the superscript f_2 in the rest of this analysis for notational convenience) and action a_{t_m}. Let the users be indexed in the order of their belief values in slot t_m, i.e, $O_{t_m} = [1 \dots N_f]$. Assuming $\{\mathfrak{A}_{t_k}\}_{k \leq m-1} = \{\widehat{\mathfrak{A}}_{t_k}\}_{k \leq m-1}$. Let S_{t_k}, the state vector, denote the 1/0 channel states of the users at t_k. We write the net expected reward as follows

$$\begin{aligned} & V_{t_m}(\pi_{t_m}, \{a_{t_m}, \{\widehat{\mathfrak{A}}_{t_k}\}_{k \leq m-1}\}) \\ &= \pi_{t_m}(a_{t_m}) + \beta_{m-1} \sum_{S_{t_m}} P_{S_{t_m}|\pi_{t_m}}(S_{t_m}|\pi_{t_m}) \hat{V}_{t_{m-1}}(S_{t_m}, O_{t_{m-1}}), \end{aligned}$$

where $\hat{V}_{t_{m-1}}$ is the expected future reward conditioned on the state vector in the previous slot on the sporadic time axis, i.e., t_m. The *hat* on this quantity emphasizes the use of the greedy policy in all $t_k, k \leq m-1$. $P_{S_{t_m}|\pi_{t_m}}(S_{t_m}|\pi_{t_m})$ is the conditional probability of the current state vector S_{t_m} given the belief vector π_{t_m}. Note that the schedule order vector $O_{t_{m-1}}$ is only a function of O_{t_m} and the state $S_{t_m}(a_{t_m})$, 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 $a_{t_m} = n$ and $a_{t_m} = n+1$ where $n \in \{1 \dots F_2 - 1\}$. Let Y and X be random binary vectors of lengths $n-1$ and $F_2 - n - 1$ (empty when the length is non-positive) respectively. Then,

$$\begin{aligned} & V_{t_m}(\pi_{t_m}, \{a_{t_m} = n, \{\widehat{\mathfrak{A}}_{t_k}\}_{k \leq m-1}\}) \\ &= \pi_{t_m}(n) + \beta_{m-1} \Big(\sum_{Y,X} P_{S_{t_m}|\pi_{t_m}}([Y\ 0\ 0\ X]|\pi_{t_m}) \times \hat{V}_{t_{m-1}}\big([Y\ 0\ 0\ X], [\{O_{t_m} - n\}\ n]\big) \\ &\quad + \sum_{Y,X} P_{S_{t_m}|\pi_{t_m}}([Y\ 0\ 1\ X]|\pi_{t_m}) \times \hat{V}_{t_{m-1}}\big([Y\ 0\ 1\ X], [\{O_{t_m} - n\}\ n]\big) \\ &\quad + \sum_{Y,X} P_{S_{t_m}|\pi_{t_m}}([Y\ 1\ 0\ X]|\pi_{t_m}) \times \hat{V}_{t_{m-1}}\big([Y\ 1\ 0\ X], [n\ \{O_{t_m} - n\}]\big) \\ &\quad + \sum_{Y,X} P_{S_{t_m}|\pi_{t_m}}([Y\ 1\ 1\ X]|\pi_{t_m}) \times \hat{V}_{t_{m-1}}\big([Y\ 1\ 1\ X], [n\ \{O_{t_m} - n\}]\big) \Big). \end{aligned} \tag{5.11}$$

Since the Markov channel statistics are identical across the users, we have the following symmetry property: for any $k \geq 1$,

$$\begin{aligned} \hat{V}_{t_k}(S_{t_{k+1}}, O_{t_k}) &= \hat{V}_{t_k}(\tilde{S}_{t_{k+1}}, \tilde{O}_{t_k}) \\ \text{if} \quad S_{t_{k+1}}(O_{t_k}(i)) &= \tilde{S}_{t_{k+1}}(\tilde{O}_{t_k}(i)) \ \ \forall\, i \in \{1 \dots F_2\}. \end{aligned} \tag{5.12}$$

Expanding $V_{t_m}(\pi_{t_m}, \{a_{t_m} = n+1, \{\widehat{\mathfrak{A}}_{t_k}\}_{k \leq m-1}\})$ along the lines of (C.8), and using the symmetry property, with further mathematical simplification, we can evaluate the difference in the net expected reward as follows,

$$\begin{aligned} & V_{t_m}(\pi_{t_m}, \{a_{t_m} = n, \{\widehat{\mathfrak{A}}_{t_k}\}_{k \leq m-1}\}) - V_{t_m}(\pi_{t_m}, \{a_{t_m} = n+1, \{\widehat{\mathfrak{A}}_{t_k}\}_{k \leq m-1}\}) \\ = \ & \Big(\pi_{t_m}(n) - \pi_{t_m}(n+1)\Big)\Big(1 - \beta_{m-1} \sum_{Y,X} \Big[\big[\hat{V}_{t_{m-1}}\big([Y\ 1\ X\ 0], [1 \dots F_2]\big) \\ & - \hat{V}_{t_{m-1}}\big([1\ Y\ 0\ X], [1 \dots F_2]\big)\big] \times P_{S_{t_m}|\pi_{t_m}}\big([S_{t_m}(1) \dots S_{t_m}(n-1)] = Y | \pi_{t_m}\big) \times \\ & P_{S_{t_m}|\pi_{t_m}}\big([S_{t_m}(n+2) \dots S_{t_m}(F_2)] = X | \pi_{t_m}\big)\Big]\Big). \end{aligned} \tag{5.13}$$

Lemma 12. *The greedy policy maximizes the per-cell discounted reward in cell 2, on the sporadic time axis* $\mathbf{t_n}$*, if the following (sufficient) condition holds.*

$$\hat{V}_{t_{m-1}}\big([Y\ 1\ X\ 0], [1 \dots F_2]\big) - \hat{V}_{t_{m-1}}\big([1\ Y\ 0\ X], [1 \dots F_2]\big) \ \leq \ 1, \tag{5.14}$$

$\forall\, \mathbf{n} \geq m > 1$, $n \in \{1 \dots F_2 - 1\}$, *where* Y *and* X *are random binary vectors of length* $n-1$ *and* $F_2 - n - 1$*, respectively, and* $\hat{V}_{t_{m-1}}$ *is the reward accrued under the greedy policy, i.e., when* $\mathfrak{A}_{t_k} = \widehat{\mathfrak{A}}$ *for all* $k \leq m-1$.

Proof. The proof is established using backward induction. Details are available in the appendix. □

We now formally introduce the optimal multiuser scheduling policy in the two-cell system with asymmetric cooperation.

Proposition 16. *The policy* $\{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\}$ *is optimal in the finite horizon, discounted reward setup.*

Proof. Using sample path arguments, we show that the sufficient condition in Lemma 12 holds. We then use Lemma 11 and Lemma 12 to establish the proposition. Details of the proof are available in the appendix. □

We now show that the optimality of $\{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\}$ extends to the infinite horizon, discounted reward setup as well. The argument is similar to that used in [45] in the single cell system.

Proposition 17. *The policy* $\{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\}$ *is optimal in the infinite horizon, discounted reward setup.*

Proof. Let $V(\pi, \mathfrak{A})$ be the total reward corresponding to the infinite horizon, discounted reward, two cell scheduling problem (asymmetric case) when policy $\mathfrak{A}$ is in effect in every slot. The belief values are represented by π. By definition,

$$V(\pi, \mathfrak{A}) = \mathrm{E}_{\mathcal{R}|\pi} \lim_{h\to\infty} \sum_{k=1}^{h} R_k(\mathcal{R}, \mathfrak{A}), \tag{5.15}$$

where $R_k(\mathcal{R}, \mathfrak{A})$ is the immediate reward earned in slot k under policy $\mathfrak{A}$ and when the channel realization is $\mathcal{R}$. The expectation is performed over the channel realization $\mathcal{R}$ conditioned on the initial belief values π. Since $\sum_{k=1}^{h} R_k(\mathcal{R}, \mathfrak{A})$ is upper bounded by $\frac{1}{1-\beta}$ and lower bounded by 0 uniformly for all $h > 0$, using the Bounded Convergence Theorem [52], we can interchange the expectation and limit to give

$$\begin{aligned} V(\pi, \mathfrak{A}) &= \lim_{h\to\infty} \mathrm{E}_{\mathcal{R}|\pi} \sum_{k=1}^{h} R_k(\mathcal{R}, \mathfrak{A}) \\ &= \lim_{h\to\infty} V_h(\pi, \mathfrak{A}), \end{aligned} \tag{5.16}$$

where $V_h(\pi, \mathfrak{A})$ is the finite horizon, discounted total reward under policy $\mathfrak{A}$. Now, with the optimality of $\{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\}$ established for the finite horizon, discounted reward setup, the optimal finite horizon, discounted reward is given by

$$
\begin{aligned}
&V_h(\pi, \{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\}) \\
&= \max_{(\mathsf{a}_h, a_h)} \Big(R_h(\pi, (\mathsf{a}_h, a_h)) + \beta\big(\pi(\mathsf{a}_h)\pi(a_h) V_{h-1}(T(\pi)|_{(\mathsf{a}_h, a_h), F_h^1=1, F_h^2=1}, \{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\}) \\
&\quad + (1-\pi(\mathsf{a}_h))\pi(a_h) V_{h-1}(T(\pi)|_{(\mathsf{a}_h, a_h), F_h^1=0, F_h^2=1}, \{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\}) \\
&\quad + \pi(\mathsf{a}_h)(1-\pi(a_h)) V_{h-1}(T(\pi)|_{(\mathsf{a}_h, a_h), F_h^1=1, F_h^2=0}, \{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\}) \\
&\quad + (1-\pi(\mathsf{a}_h))(1-\pi(a_h)) V_{h-1}(T(\pi)|_{(\mathsf{a}_h, a_h), F_h^1=0, F_h^2=0}, \{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\})\big)\Big). \qquad (5.17)
\end{aligned}
$$

Taking $\lim_{h\to\infty}$ on both sides and interchanging the limit and max on the right hand side (possible due to the finiteness of the scheduling action space), we have

$$
\begin{aligned}
&V(\pi, \{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\}) \\
&= \max_{(\mathsf{a}, a)} \Big(R(\pi, (\mathsf{a}, a)) + \beta\big(\pi(\mathsf{a})\pi(a) V(T(\pi)|_{(\mathsf{a}, a), F^1=1, F^2=1}, \{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\}) \\
&\quad + (1-\pi(\mathsf{a}))\pi(a) V(T(\pi)|_{(\mathsf{a}, a), F^1=0, F^2=1}, \{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\}) \\
&\quad + \pi(\mathsf{a})(1-\pi(a)) V(T(\pi)|_{(\mathsf{a}, a), F^1=1, F^2=0}, \{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\}) \\
&\quad + (1-\pi(\mathsf{a}))(1-\pi(a)) V(T(\pi)|_{(\mathsf{a}, a), F^1=0, F^2=0}, \{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\})\big)\Big) \\
&= \max_{(\mathsf{a}, a)} \big(R(\pi, (\mathsf{a}, a)) + \beta\, \mathrm{E}\, V(T(\pi)|_{(\mathsf{a}, a), F^1, F^2}, \{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\})\big), \qquad (5.18)
\end{aligned}
$$

where we have used $\lim_{h\to\infty} V_h(\pi, \mathfrak{A}) = V(\pi, \mathfrak{A})$ from (5.16) in the first equality. The expectation in the last equality is over the ARQ feedback. Note that the preceding equation is the Bellman equation [36]. Since a policy that satisfies the Bellman equation is optimal, we have $\{\hat{\mathfrak{A}}, \hat{\mathfrak{A}}^f, \hat{\mathfrak{A}}^n\}$ is optimal in the infinite horizon, discounted reward, two cell, asymmetric scheduling problem. □

Recall from Chapter 3 that the greedy policy can be easily implemented by a simple round-robin algorithm in a single cell system, when the ARQ feedback is deterministically delayed, or, in the present context, instantaneous. Thus the preceding result on the optimality of a policy with cell-wise greedy components is very encouraging from an implementation point of view. Fig. 5.5 provides an illustration of the simple round-robin implementation of the optimal policy in the two cell system under asymmetric cooperation.

5.4 Scheduling under Symmetric Cooperation between Cells - Index Policy

A direct optimality analysis of the ARQ based scheduling problem with symmetric cooperation appears very difficult due to the complex relationship between the schedule decisions across space and time. We therefore establish a connection between the scheduling problem and the restless multiarmed bandit processes (RMAB) [23] and make use of the established theory behind RMAB in our analysis. We proceed with a survey on the RMAB theory.

5.4.1 Restless Multiarmed Bandit Processes

Multi-armed bandit processes (MAB) [53] are defined as a family of sequential dynamic resource allocation problems in the presence of several competing, *independently evolving* projects. They are characterized by a fundamental trade-off between decisions guaranteeing high immediate rewards versus those that sacrifice immediate rewards for better future rewards. Several technological and scientific disciplines such as sensor management, manufacturing systems, economics, queueing and communication networks [53] encounter resource allocation problems that can be modeled as

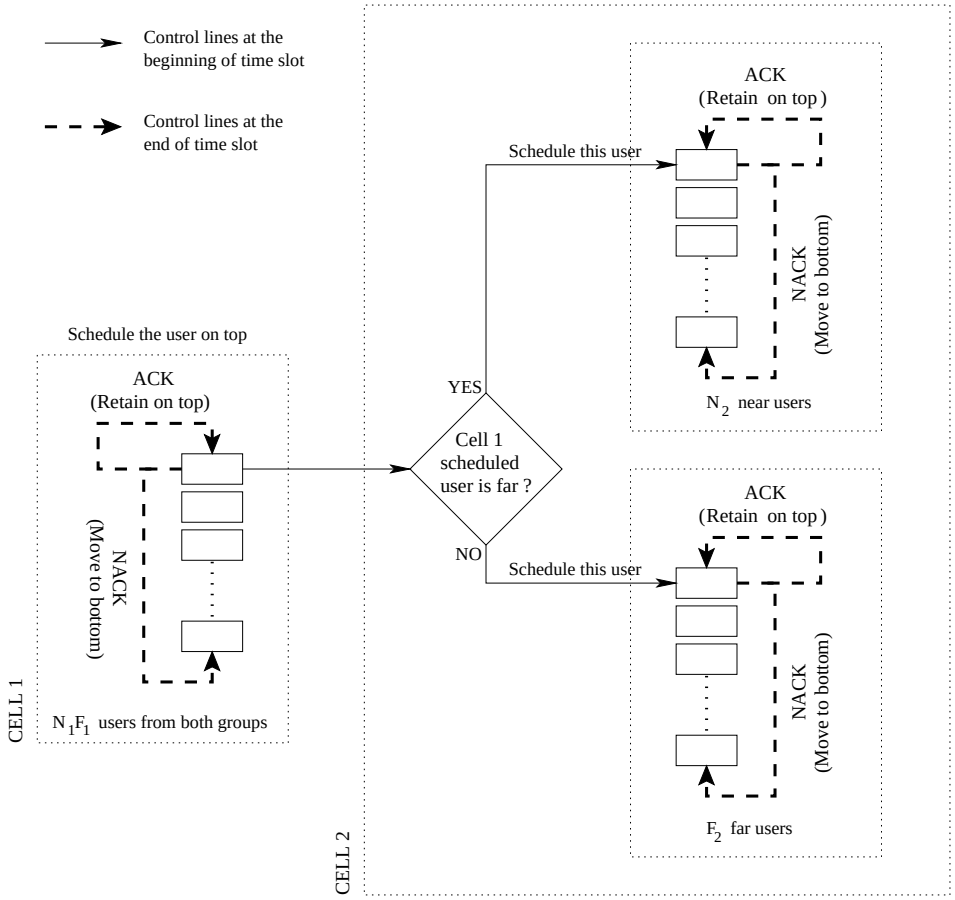

Figure 5.5: Illustration of the optimal scheduling policy implementation under asymmetric cooperation. During initialization, the users are ordered based on their belief values across groups in cell 1 and within groups in cell 2. Based on these ordered user lists, the optimal scheduling policy follows the illustrated round robin algorithm.

MAB processes. In the classic MAB process, in each slot, a single project must be allocated the available system resources. The state of the scheduled project evolves from the current time slot to the next time slot, whereas, the states of those not scheduled remain frozen. Gittins and Jones [54] studied these processes and showed that the optimal solution is of the index type: for each bandit process (i.e., project or arm of the MAB), an index that is a function of the state of the project is computed and the project with the highest index is scheduled. This index was referred to by the authors as the Dynamic Allocation Index, but is now simply known as *Gittin's index*. Note that the optimal scheduling policy, which originally required the solution of an N-armed bandit process (N being the number of projects), is now reduced to the determination of the Gittin's index for N single-armed bandit processes, thus significantly reducing the solution complexity. This complexity reduction is one of the main reasons behind the immense interest in index policies for the MAB process and its variant, the RMAB process, discussed next.

Whittle [23] generalized the MAB process as follows: In each slot, exactly $M \geq 1$ projects are scheduled. Furthermore, the states of the remaining $N - M$ projects are not frozen like in MAB, but evolve in time, and contribute rewards (W) known as *passivity* rewards. These processes are called *Restless* multiarmed bandit processes (RMAB), the term "restless" reflecting the state evolution of the unscheduled projects. For the RMAB, Whittle defined *indexability* property and the associated indexability framework as follows:

Indexability Framework: *Consider only one project of the RMAB. The scheduler in each slot must either activate the project or let it stay idle. In the former case a reward dependent on the state of the project is accrued. This reward structure is the*

same as the one used in the original RMAB. In the case of the inactivate decision, a reward W for passivity is accrued. The goal of the scheduler is to maximize the total discounted reward over an infinite horizon. For a project state π, the value of W corresponding to equal net rewards for active and passive decisions is defined as the index $I(\pi)$. Denote the optimal activate/idle scheduling policy as the W-subsidy policy. The notion of *Indexability* was defined by Whittle as follows.

Whittle's Indexability: *Let $D(W)$ be the set of states under which a project would be made passive under the W-subsidy policy. The project is indexable if $D(W)$ increases monotonically from ϕ to $\mathcal{S}$ as W increases from $-\infty$ to ∞.*

Above, ϕ denotes the empty set and $\mathcal{S}$ is the set of all possible project states. The indexability property yields a consistent ordering of states with respect to indices, i.e., if $I(\pi_1) > I(\pi_2)$ and if it is optimal to activate a given project when in state π_2, then it is optimal to activate the same project when in state π_1. A graphical interpretation of indexability is given in Fig. 5.6.

Returning to the RMAB scheduling problem, Whittle proposed the following *index* scheduling policy: In each slot, activate the M projects that have the greatest indices. Note that the natural ordering of states based on indices (under indexability) gives credibility to the index policy. Whittle showed that, under indexability, when the strict constraint on the number of projects per slot (M) is relaxed to an average constraint, the index scheduling policy becomes optimal. He also showed that when the *restless* aspect is removed from the RMAB and $W = 0$, the index reduces to the Gittins index and hence the index policy becomes optimal. He conjectured that, in the restless case, with $\frac{M}{N}$ fixed, as $M, N \to \infty$, the index policy is optimal. This was later proved to be true in [55] except for very special cases of RMAB processes.

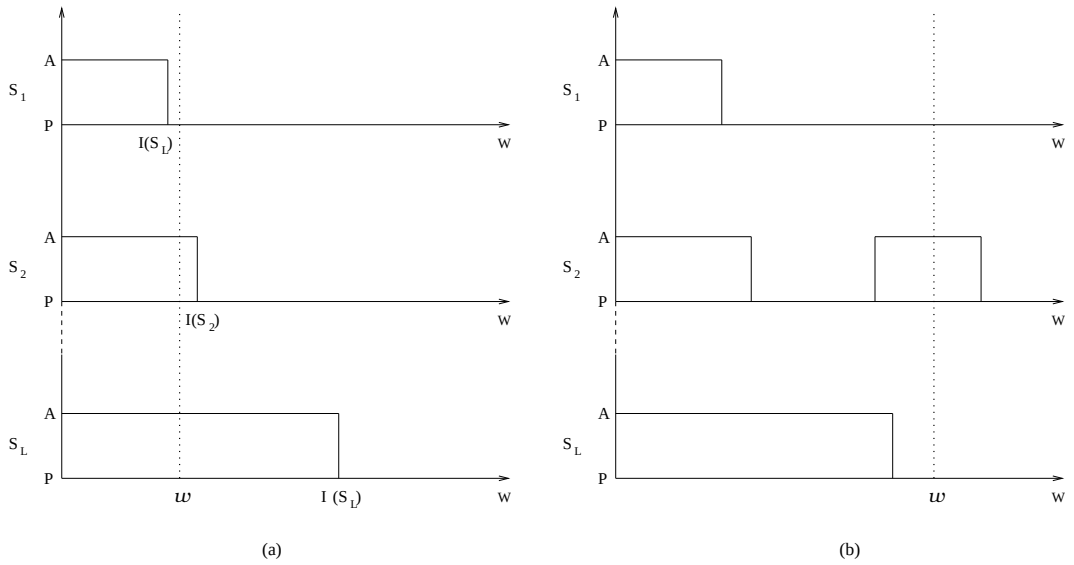

Figure 5.6: Optimal W-subsidy policy versus W is plotted for a given project over various states in (a) an indexable system and (b) a non-indexable system. Let A indicate when an active decision is optimal and P indicate when passivity is optimal. In the indexable system (a), states are ordered based on the index values $I(S_i)$. It is clear that if it is optimal to activate at state S_i, it is also optimal to activate at states $S_j, j \geq i$. This is highlighted at $W = w$ with $S_i = S_2$. This ordering is absent in the non-indexable system (b), for instance, when $W = w$. From (a) and (b) it is evident that the ON-OFF structure of the optimal schedule plot is necessary and sufficient for indexability to hold.

Indexability is a very strict requirement [23] that is hard to check. There have been several works [55–61] on indexability and index policies for various RMAB processes. In [55], for a special class of RMAB, the authors showed that, if the RMAB is indexable, then, under certain technical conditions, the index policy is optimal. In [56], the authors provided a sufficient condition for the indexability of a single restless bandit. The authors in [58] investigated indexability under a set of conditions called Partial Conservation Laws (PCL). They identified a class of RMAB processes that satisfy the PCL and are indexable in the sense of Whittle. They also showed that, under PCL, if the rewards belong to a certain "admissible region" then a priority index-based allocation policy is optimal. In [60], the authors re-examined the channel-probing based cognitive radio scheduling (equivalently, the ARQ based, single cell scheduling) problem of [45] from an RMAB perspective and studied the performance of the Whittle's index policy for non-identical arms. In [61], the authors considered a RMAB process with improving/deteriorating jobs in a queueing network scenario. They established the indexability of the processes and demonstrated, via numerical analysis, the strong performance of the index policy. Performance guarantees for the index policy were also obtained. Thus we conclude that the notion of indexability and the corresponding index policy offer a promising starting point towards the analysis of RMAB scheduling.

Returning to the ARQ-based two-cell scheduling problem, we first consider the special case that the near users in cell 1 are permanently paired (one to one) with far users in cell 2 and vice versa (which requires $N = F$). Thus, when a given user is scheduled in a cell, its partner would be scheduled in the adjacent cell. In this case, we can visualize each pair as a restless bandit with one and only one pair scheduled

in any slot. Thus the ARQ-based scheduling problem in the two-cell system becomes a RMAB process.

In our ARQ-based two-cell scheduling problem, we have no permanent pairing condition in general, and thus we have a set of $2NF$ projects, counting all possible legitimate pairings across cells. These projects do not evolve independently and hence do not constitute a RMAB process[14]. Thus we have a more complex variant of the RMAB process. To the best of our knowledge, there exists no analysis of scheduling for this variant of the RMAB process.

From our earlier discussion, we recall that the index policies are very attractive from an implementation point of view. From an optimality point of view, the attractiveness of the index policies can be attributed to the natural ordering of states (and hence projects) based on indices, as guaranteed by indexability. Encouraged by these properties, we study the two-cell scheduling problem in the framework of Whittle's indexability. Using the structural results of this study, for the case $p > r$ and $p + r \geq 1$, we propose an index policy for the ARQ based, two-cell scheduling problem at hand.

5.4.2 Indexability Analysis

Returning to the two-cell ARQ based scheduling problem, we perform a Whittle indexability analysis on a single legitimate project made up of a near-far or a far-near user pair. In each slot, the state of the project, given by (π_1, π_2), is made up of the belief values of the channels of the users. If the project is scheduled in a slot, i.e., if the users are scheduled, the belief value evolves into one of the following states: $\{(r, r), (r, p), (p, r), (p, p)\}$ corresponding, respectively, to ARQ feedbacks

[14]The projects must evolve independently in RMAB process, by definition.

(NACK,NACK), (NACK,ACK), (ACK,NACK) and (ACK,ACK) respectively. Recall that p and r are the elements of the probability transition matrix of the Markov modeled user channels with $p \geq r$.

In the reminder of this subsection, we report results on the partial characterization, along with the thresholdability properties, of the optimal W-subsidy scheduling policy. We also provide partial indexability results when $p > r$ *and* $p + r \geq 1$. Consistent with the two cell scheduling setup, the W-subsidy scheduling is performed over an infinite horizon with discounted rewards. Define V^a and V^p as the total discounted reward functions corresponding to activate and inactivate decisions in the current slot and optimal decisions in future slots. Let V denote the optimal total discounted reward. Thus, with $V_{xy} := V(x, y)$,

$$\begin{aligned} V^a(\pi_1, \pi_2) &= \pi_1 + \pi_2 \\ &\quad + \beta\Big(\pi_1\pi_2 V_{pp} + \pi_1(1-\pi_2)V_{pr} + (1-\pi_1)\pi_2 V_{rp} + (1-\pi_1)(1-\pi_2)V_{rr}\Big) \\ V^p(\pi_1, \pi_2) &= W + \beta V(T(\pi_1), T(\pi_2)) \\ V(\pi_1, \pi_2) &= \max(V^a(\pi_1, \pi_2), V^p(\pi_1, \pi_2)) \end{aligned} \tag{5.19}$$

where recall $T(\pi)$ is the one-step evolution operator on the belief value π when an inactivate decision is made. From the property of the Markov channel, $T(\pi) = \pi p + (1 - \pi)r$. Note that the immediate reward on activate decision is given by the sum of the belief values, consistent with the reward structure in the original two-cell scheduling problem.

With the reward structure thus defined, the Whittle indexability setup can be linked to the broadcast scheduling problem we studied in Chapter 2, when the number of broadcast users = 2. Recall that we had established thresholdability properties of

the two-user broadcast towards designing a threshold scheduling policy. Using this analysis, we now report the following results for the Whittle indexability analysis of the current scheduling problem. Proposition 18 to Corollary 7 are reproduced from Chapter 2 and hence the proofs are omitted.

Proposition 18. *When the reward for passivity, $W \notin (2r, 2p)$, the optimal W-subsidy scheduling policy is given by*

$$(\pi_1, \pi_2) \in \begin{cases} \mathcal{A}, & \text{if } \pi_1 + \pi_2 > W \\ \mathcal{P}, & \text{if } \pi_1 + \pi_2 \leq W \end{cases}$$

Recall the proof proceeds by establishing that the future reward after activate and inactivate decisions are equal when $W \notin (2r, 2p)$, thus allowing a direct comparison of the immediate rewards.

Let R_I denote the region $\{(\pi_1, \pi_2); \pi_1 \in [\pi_{ss}, 1], \pi_2 \in [\pi_{ss}, 1]\}$. Let R_{II} denote the union of the regions $R_{II}^1 \doteq \{(\pi_1, \pi_2); \pi_1 \in [0, \pi_{ss}], \pi_2 \in [\pi_{ss}, 2\pi_{ss} - \pi_1]\}$ and $R_{II}^2 \doteq \{(\pi_1, \pi_2); \pi_2 \in [0, \pi_{ss}], \pi_1 \in [\pi_{ss}, 2\pi_{ss} - \pi_2]\}$. Let $\mathcal{A}$ be the set of states (π_1, π_2) in which it is optimal to activate. Let $\mathcal{P}$ be the set corresponding to optimal inactivate decision. From the analysis in Chapter 2, we have the following thresholdability property on the W-subsidy scheduling policy when W is such that $V^a(\pi_{ss}, \pi_{ss}) \geq V^p(\pi_{ss}, \pi_{ss})$.

Proposition 19. *If W is such that $V^a(\pi_{ss}, \pi_{ss}) \geq V^p(\pi_{ss}, \pi_{ss})$, then*

(1) $R_I \in \mathcal{A}$

(2) $V^a(\pi_{ss}, \pi_{ss}) = V^p(\pi_{ss}, \pi_{ss}) \Rightarrow R_{II} \in \mathcal{P}$

(3) $V^a(\pi_{ss}, \pi_{ss}) > V^p(\pi_{ss}, \pi_{ss}) \Rightarrow$ *(thresholdability property) In the region R_{II}^1, if for $k \in [-1, 0]$, $\exists$ a π_1^* and $\pi_2^* = \pi_1^* k + \pi_{ss}(1-k)$ such that $V^a(\pi_1^*, \pi_2^*) = V^p(\pi_1^*, \pi_2^*)$,*

then

$$(\pi_1, \pi_2 = \pi_1 k + \pi_{ss}(1-k)) \in \begin{cases} \mathcal{A}, & \text{if } \pi_1 \in (\pi_1^*, \pi_{ss}] \\ \mathcal{P}, & \text{if } \pi_1 \in [0, \pi_1^*] \end{cases}$$

If $\nexists$ *such a* (π_1^*, π_2^*)*, then*

$$(\pi_1, \pi_2 = \pi_1 k + \pi_{ss}(1-k)) \in \mathcal{A} \ \forall \ \pi_1 \in [0, \pi_{ss}].$$

Similarly, in the region R_{II}^2*, if for* $k \in [-1, 0]$*,* $\exists$ *a* $\pi_2^* \in [0, \pi_{ss}]$ *and* $\pi_1^* = \pi_2^* k + \pi_{ss}(1-k)$ *such that* $V^a(\pi_1^*, \pi_2^*) = V^p(\pi_1^*, \pi_2^*)$*, then*

$$(\pi_1 = \pi_2 k + \pi_{ss}(1-k), \pi_2) \in \begin{cases} \mathcal{A}, & \text{if } \pi_2 \in (\pi_1^*, \pi_{ss}] \\ \mathcal{P}, & \text{if } \pi_2 \in [0, \pi_1^*] \end{cases}$$

If $\nexists$ *such a* (π_1^*, π_2^*)*, then*

$$(\pi_1 = \pi_2 k + \pi_{ss}(1-k), \pi_2) \in \mathcal{A} \ \forall \ \pi_2 \in [0, \pi_{ss}].$$

The threshold boundary identified in the preceding proposition is characterized below.

Corollary 6. *Within region* R_{II}*, the threshold boundary is given by the upper segment of the hyperbola*

$$V^a(\pi_1, \pi_2) \ = \ W + \beta V^a(T(\pi_1), T(\pi_2))$$

where

$$\begin{aligned} V^a(x_1, x_2) \ = \ & x_1 + x_2 + \beta\big[(1-x_1)(1-x_2)V(r,r) + (1-x_1)(x_2)V(r,p) \\ & + x_1(1-x_2)V(p,r) + x_1 x_2 V(p,p), \end{aligned}$$

$T(x) = x(p-r) + r$*, and upper segment indicates the segment of the hyperbola that lies in the first quadrant around the asymptotes.*

If W is such that $V^a(\pi_{ss}, \pi_{ss}) < V^p(\pi_{ss}, \pi_{ss})$, the W-subsidy scheduling policy has the following property.

Proposition 20. *If W is such that $V^a(\pi_{ss}, \pi_{ss}) < V^p(\pi_{ss}, \pi_{ss})$, then*

(1) $(\pi_1, \pi_2) \in \mathcal{P}$, $\forall \pi_1 + \pi_2 \leq 2\pi_{ss}$

(2) (Thresholdability property) In the region R_I, if for $k \geq 0$, $\exists$ a π_1^* and $\pi_2^* = \pi_1^* k + \pi_{ss}(1-k)$ such that $V^a(\pi_1^*, \pi_2^*) = V^p(\pi_1^*, \pi_2^*)$, then

$$(\pi_1, \pi_1^* k + \pi_{ss}(1-k)) \in \begin{cases} \mathcal{A}, & \text{if } \pi_1 \in [\pi_1^*, 1] \\ \mathcal{P}, & \text{if } \pi_1 \in [\pi_{ss}, \pi_1^*) \end{cases}$$

If $\nexists$ such a (π_1^*, π_2^*), then

$$(\pi_1, \pi_2 = \pi_1 k + \pi_{ss}(1-k)) \in \mathcal{P} \ \forall \ \pi_1 \in [\pi_{ss}, 1].$$

The threshold boundary is now characterized as follows.

Corollary 7. *Within region R_I, the threshold boundary is given by the upper segment of the hyperbola*

$$V^a(\pi_1, \pi_2) \ = \ \frac{W}{1-\beta}$$

where

$$\begin{aligned} V^a(x_1, x_2) \ = \ & x_1 + x_2 + \beta\big[(1-x_1)(1-x_2)V(r,r) + (1-x_1)(x_2)V(r,p) \\ & + x_1(1-x_2)V(p,r) + x_1 x_2 V(p,p). \end{aligned}$$

The statement that the threshold boundaries identified in Corollary 6 and Corollary 7 are hyperbolas is easily verified using the definition of V^a in (5.19). The threshold boundaries are illustrated in Fig. 5.7, reproduced from Chapter 2.

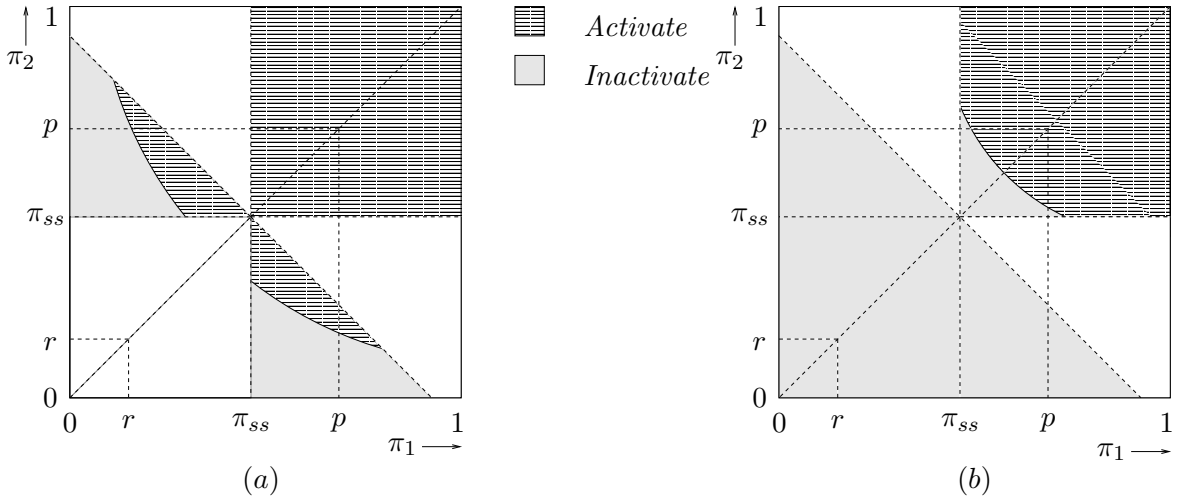

Figure 5.7: Illustration of the threshold boundaries when (a) $(\pi_{ss}, \pi_{ss}) \in \mathcal{A}$, (b) $(\pi_{ss}, \pi_{ss}) \in \mathcal{P}$.

For the case, $p + r \geq 1$ (in addition to the positive correlation condition $p > r$ that we have assumed throughout), the threshold boundary simplifies to the following form.

Corollary 8. *If $p + r \geq 1$ and if W is such that $V^a(\pi_{ss}, \pi_{ss}) < V^p(\pi_{ss}, \pi_{ss})$, the threshold boundary is given by the upper segment of the hyperbola*

$$\pi_1 + \pi_2 + \pi_1\pi_2\beta\Big[\frac{2p + \beta(1-p^2)\frac{W}{1-\beta}}{1-\beta p^2} - \frac{W}{1-\beta}\Big] = W.$$

In addition, π_2 written as a function of π_1 is convex and decreasing in π_1.

Proof. With $p + r \geq 1$,

$$\begin{aligned} p - r &\leq p^2 - r^2 \\ \Leftrightarrow (1 - (p-r))(p+r) &\leq 2r \\ \Leftrightarrow p + r &\leq \frac{2r}{1-(p-r)} = 2\pi_{ss}. \end{aligned} \tag{5.20}$$

From Proposition 20, with $(\pi_{ss}, \pi_{ss}) \in \mathcal{P}$, since $p + r \leq 2\pi_{ss}$, $(p, r) \in \mathcal{P}$. Also $(T^l(p), T^l(r)) \in \mathcal{P}$ for $l \geq 0$ since $(T^l(p), T^l(r)) \in \mathcal{L}((p, r), (\pi_{ss}, \pi_{ss}))$ from Lemma 4 in Chapter 2 and $\mathcal{L}((p, r), (\pi_{ss}, \pi_{ss})) \in \mathcal{P}$ from Proposition 20. Thus $V_{pr} = \frac{W}{1-\beta}$. Substituting V_{pr} in the threshold equation of Corollary 7, the threshold equation in the region $(\pi_1, \pi_2) \in [\pi_{ss}, 1]^2$ is given by

$$\pi_1 + \pi_2 + \pi_1\pi_2\beta(V_{pp} - \frac{W}{1-\beta}) \;\; = \;\; W \tag{5.21}$$

Note that $V_{pp} = 2p + \beta(p^2 V_{pp} + (1 - p^2)\frac{W}{1-\beta}$ where we have used $V_{pr} = V_{rr} = \frac{W}{1-\beta}$. Thus

$$V_{pp} \;\; = \;\; \frac{2p + \frac{W}{1-\beta}\beta(1 - p^2)}{1 - \beta p^2} \tag{5.22}$$

Substituting the expression for V_{pp} in threshold equation (5.21) establishes the first part of the corollary. That the threshold boundary thus identified lies on the upper segment of the hyperbola follows a proof technique similar to that of Corollaries 7 and 8. □

Recall that the notion of *Indexability* is defined by Whittle as below:

Adding the dependence on W explicitly to $\mathcal{P}$, we have, $\mathcal{P}(W)$ is the set of states for which it is optimal to inactive under passivity subsidy W. The project (the user pair chosen from the larger two-cell scheduling problem) is indexable if $\mathcal{P}(W)$ increases monotonically from $\emptyset$ to $\mathcal{S}$, as W increases from $-\infty$ to ∞

where $\emptyset$ is the empty set and $\mathcal{S}$ is the universal set of the states of the project. The monotonic increase of $\mathcal{P}(W)$ means that if a state $\pi = (\pi_1, \pi_2) \in \mathcal{P}(W_1)$, then $\pi \in \mathcal{P}(W_2)$ for $W_2 \geq W_1$. This is possible if and only if, for a state π, the optimal decision versus W plot is of the ON-OFF (activate-inactivate) type, i.e., $\exists$ a W^* such

that $V^a(\pi) = V^p(\pi)|_{W=W^*}$ and $\forall W < W^*$, it is optimal to activate at state π and $\forall W \geq W^*$, it is optimal to inactivate. If a state π has the above mentioned ON-OFF property we call it indexable with the index of state π given by $I(\pi) = W^*$. From the preceding discussion, Whittle indexability of the system is equivalent to indexability of all the states of the system.

We proceed to show that, for $p+r \geq 1$, the two user project we consider is partially indexable, i.e., indexability holds for specific subsets of the state space.

Proposition 21. *For $p+r \geq 1$, a state vector (π_1, π_2) is indexable if $(\pi_1, \pi_2) \in [\pi_{ss}, 1]^2$ or $\pi_1 + \pi_2 \geq 2p$.*

Proof. State (π_1, π_2) is indexable if $\exists$ a W^* such that $V^a(\pi_1, \pi_2) = V^p(\pi_1, \pi_2)|_{W=W^*}$ and $\forall W \leq W^*$, it is optimal to activate at state (π_1, π_2) and $\forall W > W^*$, it is optimal to inactivate. Let $(\pi_1, \pi_2) \in [\pi_{ss}, 1]^2$. Let W_0 be such that $V^a(\pi_{ss}) = V^p(\pi_{ss})$ at $W = W_0$. Thus using Corollary 8, W_0 is obtained by solving the equation

$$\pi_{ss} + \pi_{ss} + \pi_{ss}\pi_{ss}\beta\Big[\frac{2p + \beta(1-p^2)\frac{W_0}{1-\beta}}{1-\beta p^2} - \frac{W_0}{1-\beta}\Big] = W_0.$$

Hence

$$W_0 \;=\; (1-\beta)\frac{2\pi_{ss}(1+\frac{\beta p \pi_{ss}}{1-\beta p^2})}{1-\beta(1-\pi_{ss}^2) - \beta^2\frac{\pi_{ss}^2(1-p^2)}{1-\beta p^2}} \tag{5.23}$$

We now proceed to show that the threshold boundary given in Corollary 8 progressively moves outward as W increases from W_0. From Corollary 8, with $f = \beta\big(V_{pp} - \frac{W}{1-\beta}\big) = \beta\Big(\frac{2p+\beta(1-p^2)\frac{W}{1-\beta}}{1-\beta p^2} - \frac{W}{1-\beta}\Big)$, the threshold equation in the region $(\pi_1, \pi_2) \in [\pi_{ss}, 1]^2$ is rewritten as

$$\pi_1 + \pi_2 + \pi_1\pi_2 f \;=\; W \tag{5.24}$$

The preceding equation is valid if W is such that $(\pi_{ss}, \pi_{ss}) \in \mathcal{P}$. This is true for $W = W_0$. With π_1 fixed, the first derivative of π_2 with respect to W at $W = W_0$ is given from (5.24) by

$$\frac{d\pi_2}{dW}|_{W_0} = \frac{1 - \pi_1\pi_2\frac{df}{dW}}{1 + \pi_1 f}|_{W_0} \tag{5.25}$$

where $f = \beta\big(V_{pp} - \frac{W}{1-\beta}\big) \geq 0$ since $V(\pi_1, \pi_2) \geq \frac{W}{1-\beta}\ \forall\ (\pi_1, \pi_2)$ and $\frac{df}{dW} = -\frac{\beta(1-\beta(1-p^2))}{(1-\beta p^2)(1-\beta)} \leq 0$. Thus $\frac{d\pi_2}{dW}|_{W_0} \geq 0$. Thus the threshold boundary monotonically moves out as W increases from W_0 to $W_0 + \delta$, for $\delta \to 0$. Also note that, the monotonic increase in threshold boundary $\Rightarrow$ $(\pi_{ss}, \pi_{ss}) \in \mathcal{P}$ and hence the threshold equation in (5.24) is valid at $W = W_0 + \delta$. Repeating the first derivative based arguments for $W = W_0 + \delta$, recursively, we have for $W \geq W_0$, in the region $(\pi_1, \pi_2) \in [\pi_{ss}, 1]^2$, the threshold boundary monotonically moves out, i.e., if for a $W_1 \geq W_0$, $(\pi_1, \pi_2) \in \mathcal{P}$, then for $W_2 > W_1$, $(\pi_1, \pi_2) \in \mathcal{P}$. Note that, for a $(\pi_1, \pi_2) \in [\pi_{ss}, 1]^2$, the value of W for which (π_1, π_2) lies on the threshold curve is also such that $V^a(\pi_1, \pi_2) = V^p(\pi_1, \pi_2)$ at that W. With the threshold curve known to move out with W, we see that for $(\pi_1, \pi_2) \in [\pi_{ss}, 1]^2$, $\exists$ a W^* such that $\forall W \leq W^*$, it is optimal to activate at state (π_1, π_2) and $\forall W > W^*$, it is optimal to inactivate. Thus $(\pi_1, \pi_2) \in [\pi_{ss}, 1]^2$ is indexable. This establishes the first part of the proposition.

Consider $\pi_1 + \pi_2 \geq 2p$. When $W < \pi_1 + \pi_2$, it was shown in Chapter 2 that $(\pi_1, \pi_2) \in \mathcal{A}$. When $W \geq \pi_1 + \pi_2$, since $\pi_1 + \pi_2 \geq 2p$, $W \geq 2p$. Thus using Proposition 18, $(\pi_1, \pi_2) \in \mathcal{P}$. Thus (π_1, π_2) is indexable if $\pi_1 + \pi_2 \geq 2p$. The proposition thus follows. □

5.4.3 Index Policy

The threshold boundaries reported in Corollary 6 and Corollary 7 are illustrated in Fig. 5.7. The thresholdability results of the W-subsidy policy and hence the threshold boundaries were obtained using sufficient conditions that hold only in the shown regions. A tighter analysis needed to characterize the optimal policy in the whole state space appears intractable. We therefore make a set of assumptions (A) on the properties of the optimal W-subsidy policy and derive an index scheduling policy for the two-cell system. Our assumptions are stated next.

(A0) The threshold boundaries reported in Corollaries 6 and 7 in the restricted regions hold true in the entire state space. The extrapolations corresponding to these boundaries are illustrated in Fig. 5.8. Contrast this with the threshold boundaries illustrated in Fig. 5.7.

(A1) The threshold boundaries progressively move to the right, i.e., the region $\mathcal{P}$ progressively expands, as W increases. This is essentially Whittle's indexability. Recall that we have shown indexability to hold partially in Proposition 21 when $p + r \geq 1$.

We now classify the state space into four non-overlapping regions. Recall the definition of W_0: the value of W at which $V^a(\pi_{ss}, \pi_{ss})|_W = V^p(\pi_{ss}, \pi_{ss})|_W$. The state space is now classified as below:

- R_1: (π_1, π_2) such that $\pi_1 + \pi_2 \leq 2r$
- R_2: Region between the boundaries $\{(\pi_1, \pi_2) : \pi_1 + \pi_2 > 2r\}$ and $\{(\pi_1, \pi_2) : V^a(\pi_1, \pi_2)|_{W=W_0} = V^p(\pi_1, \pi_2)_{W=W_0}\}$. By definition of W_0, the second boundary passes through the steady state.

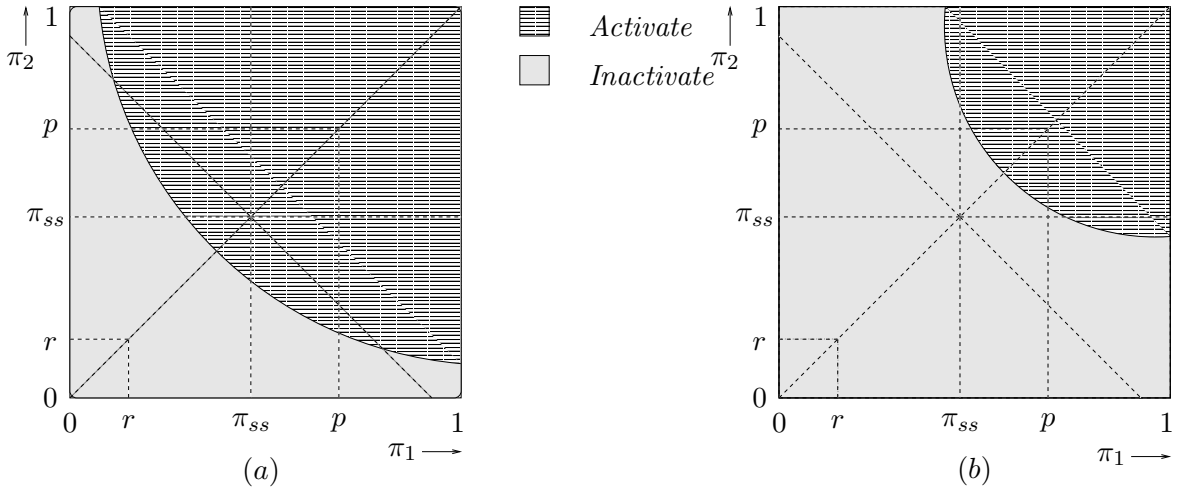

Figure 5.8: Illustration of the extrapolation of the threshold boundaries to the entire two-dimensional state space, when (a) $(\pi_{ss}, \pi_{ss}) \in \mathcal{A}$, (b) $(\pi_{ss}, \pi_{ss}) \in \mathcal{P}$.

- R_3: (π_1, π_2) such that $(\pi_1, \pi_2) \notin R_1 \cup R_2$ and $\pi_1 + \pi_2 < 2p$
- R_4: (π_1, π_2) such that $\pi_1 + \pi_2 \geq 2p$

Under assumptions (A0) and (A1), as W increases from 0 to $2r$, the threshold boundary moves progressively outwards within region R_1, with the boundary given by (π_1, π_2) such that $\pi_1 + \pi_2 = W$. As W increases from $2r$ to W_0, the threshold boundary progressively moves outward within region R_2, with the boundary given by the extrapolation of the boundary derived in Corollary 6, i.e., (π_1, π_2) such that $V^a(\pi_1, \pi_2) = W + \beta V^a(T(\pi_1), T(\pi_2))$. When W increases from W_0 to $2p$, the threshold boundary progressively moves outward within R_3 and the boundary is given from Corollary 7 by the convex curve: (π_1, π_2) such that $V^a(\pi_1, \pi_2) = \frac{W}{1-\beta}$. When W increases from $2p$ to 2, the threshold boundary progressively moves outward within region R_4 with boundary given by (π_1, π_2) such that $\pi_1 + \pi_2 = W$. This is summarized in Table 5.1.

Range of W	Threshold boundary	State space region
$W \leq 2r$	$\{(\pi_1, \pi_2) : \pi_1 + \pi_2 = W\}$	R_1
$W \in (2r, W_0]$	$\{(\pi_1, \pi_2) : V^a(\pi_1, \pi_2) = W + \beta V^a(T(\pi_1), T(\pi_2))\}$	R_2
$W \in (W_0, 2p)$	$\{(\pi_1, \pi_2) : V^a(\pi_1, \pi_2) = \frac{W}{1-\beta}\}$	R_3
$W \geq 2p$	$\{(\pi_1, \pi_2) : \pi_1 + \pi_2 = W\}$	R_4

Table 5.1: Threshold boundaries and their region affiliation for various ranges of W.

The index we employ in our policy is defined as follows: For any state (π_1, π_2), the value of W for which the threshold boundary passes through (π_1, π_2) is the index of that state. Note that if assumptions (A), were true, the index we propose is exactly the Whittle's index. Thus the index policy we propose is in fact the Whittle's index policy when (A) is true.

Note that the region of passivity, $\mathcal{P}$ lies to the left of the threshold boundary. Thus under (A), we see that, a belief vector (π_1, π_2) has an index higher than the belief vectors $(\pi_1, \pi_2 - \delta)$ and and $(\pi_1 - \delta, \pi_2)$ for $\delta \geq 0$. Call this statement (B). We now propose the index policy.

Step 0: Initialization

- For $W = 0 : \delta_W : 2$, evaluate the quantities $V(p,p)$, $V(p,r) = V(r,p)$ (thanks to the symmetry across user channels) and $V(r,r)$ using the following limit on the finite horizon Bellman equation [36]: $V(\pi_1, \pi_2) = \lim_{t\to\infty} V_t(\pi_1, \pi_2)$ where, using an appropriate measure of convergence,

$$\begin{aligned} V_t(\pi_1,\pi_2) &= \max(V_t^a(\pi_1,\pi_2), V_t^p(\pi_1,\pi_2)) \\ V_t^a(\pi_1,\pi_2) &= \pi_1+\pi_2+\beta\Big(\pi_1\pi_2 V_{t-1}(p,p)+\pi_1(1-\pi_2)V_{t-1}(p,r) \\ &\quad +(1-\pi_1)\pi_2 V_{t-1}(r,p)+(1-\pi_1)(1-\pi_2)V_{t-1}(r,r)\Big) \\ V_t^p(\pi_1,\pi_2) &= W+\beta V_{t-1}(T(\pi_1),T(\pi_2)) \end{aligned} \tag{5.26}$$

  with $V_1(\pi_1, \pi_2) = \max(\pi_1 + \pi_2, W)$.

- Evaluate W_0 as $\arg_W V^a(\pi_{ss}, \pi_{ss}) = W + \beta V^a(T(\pi_{ss}), T(\pi_{ss}))$ using the system parameters $V(p,p)$, $V(p,r)$, $V(r,p)$ and $V(r,r)$ evaluated in the previous step.

- Identify regions R_1 to R_4 using the discussion preceding before Table 1, reproduced below:

  - R_1: (π_1, π_2) such that $\pi_1 + \pi_2 \leq 2r$
  - R_2: region between the boundaries $\{(\pi_1, \pi_2) : \pi_1 + \pi_2 > 2r\}$ and $\{(\pi_1, \pi_2) : V^a(\pi_1, \pi_2)|_{W=W_0} = V^p(\pi_1, \pi_2)|_{W=W_0}\}$.
  - R_3: (π_1, π_2) such that $(\pi_1, \pi_2) \notin R_1 \cup R_2$ and $\pi_1 + \pi_2 < 2p$
  - R_4: (π_1, π_2) such that $\pi_1 + \pi_2 \geq 2p$

Index policy on belief vector $\pi = (\pi_1, \ldots, \pi_N)$

- Within each user group (n_1, f_1, n_2, f_2), identify the users that have the highest belief values. Call them n_1^*, f_1^*, n_2^* and f_2^*, respectively. From statement (B), user pair (n_1^*, f_2^*) has an index higher than any other user pair from the composite group $n_1 \times f_2$. Likewise, the user pair (f_1^*, n_2^*) has higher index than any other pair from $f_1 \times n_2$. Thus it is sufficient to compare the indices of user pairs (n_1^*, f_2^*) and (f_1^*, n_2^*).
- Calculate the index of the states corresponding to user pairs (n_1^*, f_2^*) and (f_1^*, n_2^*). Index calculation is explained as a separate step in the end.
- Schedule the user pair with the higher index.
- Receive ARQ feedback from the scheduled user pair.
- Update the belief values of all the users based on the ARQ feedback. The belief value of user i, i.e., π^i, evolves as follows: If user i was scheduled, then, if ACK feedback was received from user i, then $\pi^i \leftarrow p$, else $\pi^i \leftarrow r$. If user i was not scheduled, then $\pi^i \leftarrow T(\pi^i)$.
- Repeat the scheduling policy in the next time slot.

Index calculation for state (π_1, π_2)

- Determine the region (R_1, R_2, R_3, R_4) in which (π_1, π_2) belongs.
- Based on the identified region, identify the threshold boundary from Table 1.
- Determine the value of W for which the identified threshold boundary passes through (π_1, π_2). This can be accomplished as follows: For discretized values of $W = 0 : \delta_W : 2$, find the value of W (call it W^*) for which the threshold boundary is closest to (π_1, π_2).
- W^* is the index of the user pair. Return W^* to the Index policy.

An illustration of the proposed index policy is given in Fig. 5.9.

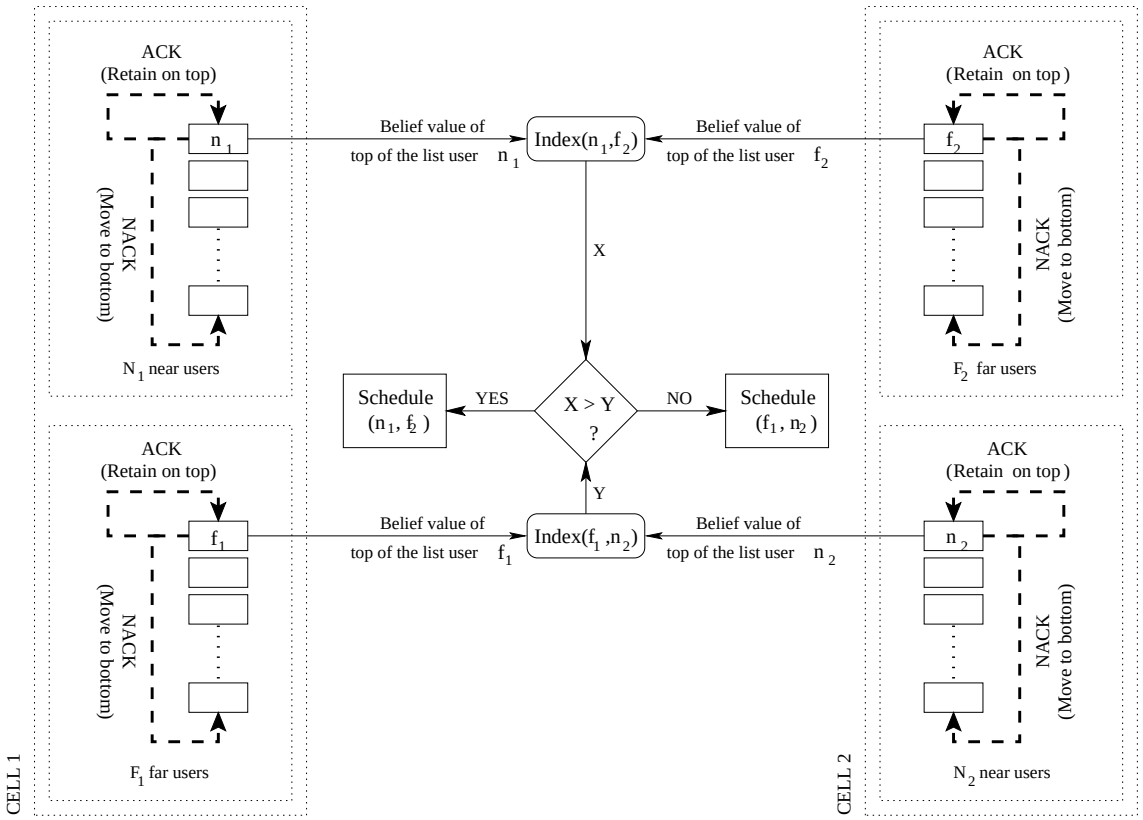

Figure 5.9: Illustration of the index policy implementation

5.5 Numerical Results and Discussion

We now proceed to report the numerical performance of the proposed index policy. Table 5.2 compares the rewards accrued by the optimal policy, V_{opt}, with that of the proposed index policy, V_{index}, when the system parameters are fixed and the initial belief values are generated randomly for each row of data. Table 5.3 compares V_{opt} and V_{index} for various randomly generated system parameters. In both tables, $p + r \geq 1$. The optimal policy is implemented by an exhaustive search over all possible $N_1 \times F_2 + F_1 \times N_2$ decisions in each time slot. The policy is repeated for increasing horizon lengths until convergence is reached. Similarly, the index policy is implemented for increasing horizon lengths until convergence. The quantity %subopt= $\frac{V_{\text{opt}} - V_{\text{index}}}{V_{\text{opt}}} \times 100\%$ quantifies the sub-optimality resulting from index policy based scheduling. The very low values of %subopt suggests that the index policy is near optimal.

To illustrate the advantage of using the ARQ feedback for scheduling, we compare V_{index} with the total reward accrued in a genie-aided system, V_{genie} in Table 5.4. The genie-aided system is defined as follows: At the end of every time slot, the scheduler learns about the channel state of *every* user in the system in that time slot. The optimal scheduling policy in the genie aided system is greedy, i.e., in each slot schedule the legitimate user pair that has the highest sum of belief values. The quantity %ARQgain= $\frac{V_{\text{index}} - V_{\text{rand}}}{V_{\text{genie}} - V_{\text{rand}}}$ quantifies the gain in reward when the ARQ feedback is used in scheduling, where V_{rand} is the reward accrued by a policy that ignores any channel feedback from the users and schedules randomly. The high values of %ARQgain in Table 5.4 underlines the significance of exploiting ARQ feedback in our scheduling setup. Consider the following two cell system with cell breathing in effect: at the beginning of each slot both the base stations learn about the instantaneous channel

$p = 0.8947,\ r = 0.3289,\ \beta = 0.2060$		
V_{opt}	V_{index}	%subopt
1.6509	1.6507	0.0097 %
1.7759	1.7759	0.0000 %
2.2293	2.2293	0.0000 %
2.0303	2.0303	0.0000 %
1.4728	1.4728	0.0000 %
1.9026	1.9026	0.0000 %
2.3118	2.3118	0.0000 %
1.9977	1.9977	0.0003 %
1.9965	1.9965	0.0000 %
1.9515	1.9514	0.0013 %

$p = 0.7452,\ r = 0.6356,\ \beta = 0.8739$		
V_{opt}	V_{index}	%subopt
5.8458	5.8457	0.0015 %
6.1100	6.1098	0.0031 %
5.8890	5.8890	0.0002 %
5.8592	5.8592	0.0002 %
5.9301	5.9300	0.0013 %
5.8352	5.8351	0.0003 %
5.6649	5.6648	0.0011 %
5.6029	5.6028	0.0014 %
5.7806	5.7805	0.0015 %
5.9786	5.9784	0.0030 %

Table 5.2: Illustration of the near optimal performance of the proposed index policy. Each table corresponds to a fixed set of system parameters. Each row within the tables correspond to randomly generated initial belief values. $N_1 = N_2 = 2$ and $F_1 = F_2 = 3$ is used throughout.

state of all the users in the system. Denote by V_{genie^*}, the optimal total discounted reward corresponding to this system. The relative values of V_{genie^*}, V_{genie}, V_{index} and V_{rand} are plotted in Fig. 5.10 against the discount factor β for various values of system parameters p, r. Note from the figure that the loss in performance when the channel state feedback is delayed by one time slot is almost the same as the loss when the channel states of only the scheduled users are fed back, with a slot delay, i.e., the ARQ feedback system.

The preceding discussion on the numerical results underlines the advantage and sufficiency of exploiting ARQ feedback for opportunistic scheduling.

$N1 = 2,\ F1 = 3,\ N2 = 2,\ F2 = 3$					
p	r	β	V_{opt}	V_{index}	%subopt
0.7638	0.3663	0.8013	4.8839	4.8839	0.0008 %
0.9504	0.5462	0.8452	5.8011	5.8011	0.0002 %
0.8476	0.4230	0.5358	3.2915	3.2915	0.0001 %
0.7452	0.6356	0.8739	6.1100	6.1098	0.0031 %
0.7825	0.5010	0.4170	2.3547	2.3547	0.0000 %
0.5546	0.4580	0.3381	2.4535	2.4535	0.0001 %
0.8536	0.6670	0.2880	2.2923	2.2923	0.0001 %
0.6688	0.4065	0.7413	3.6949	3.6949	0.0002 %
0.8947	0.3289	0.2060	1.7759	1.7759	0.0000 %
0.5387	0.4922	0.7067	3.6443	3.6443	0.0007 %
0.7309	0.3826	0.7315	4.5159	4.5159	0.0000 %
0.9994	0.7362	0.2210	1.7294	1.7294	0.0000 %
0.8914	0.8532	0.7962	5.4776	5.4776	0.0001 %
0.8022	0.6029	0.9480	6.9358	6.9357	0.0006 %
0.9238	0.6184	0.6090	4.2933	4.2933	0.0000 %

Table 5.3: Illustration of the near optimal performance of the proposed index policy. Each row corresponds to randomly generated system parameters (p, r, and β) and initial belief values. $N_1 = N_2 = 2$ and $F_1 = F_2 = 3$ is used.

$N1=2,\ F1=3,\ N2=2,\ F2=3$						
p	r	β	V_{genie}	V_{index}	V_{rand}	%ARQgain
0.7397	0.5790	0.8436	5.5785	5.5179	4.6383	93.5592 %
0.7600	0.4697	0.1551	1.6406	1.6313	1.1480	98.1181 %
0.7058	0.5223	0.7954	4.4204	4.3315	3.6053	89.0940 %
0.7801	0.6404	0.9649	7.2043	7.1424	6.2261	93.6788 %
0.7994	0.3420	0.2678	2.2562	2.2407	1.5141	97.9097 %
0.9919	0.2318	0.4784	3.1231	3.0587	2.0004	94.2659 %
0.6064	0.5657	0.2762	2.0390	2.0370	1.7046	99.3912 %
0.8446	0.5150	0.3688	2.8201	2.8091	2.0207	98.6200 %
0.7051	0.5944	0.5793	3.2111	3.1886	2.3930	97.2564 %
0.9631	0.7544	0.3493	2.6820	2.6756	2.1638	98.7781 %
0.8519	0.2541	0.4629	3.2259	3.1812	2.1514	95.8427 %
0.9644	0.6004	0.9965	8.7821	8.5872	7.6258	83.1464 %
0.8237	0.7089	0.4615	3.1647	3.1577	2.2633	99.2261 %
0.7413	0.5105	0.4637	2.8621	2.8307	2.0559	96.1071 %
0.6582	0.4114	0.2363	2.2611	2.2558	1.4567	99.3502 %

Table 5.4: Illustration of the significance of using ARQ feedback in opportunistic scheduling.

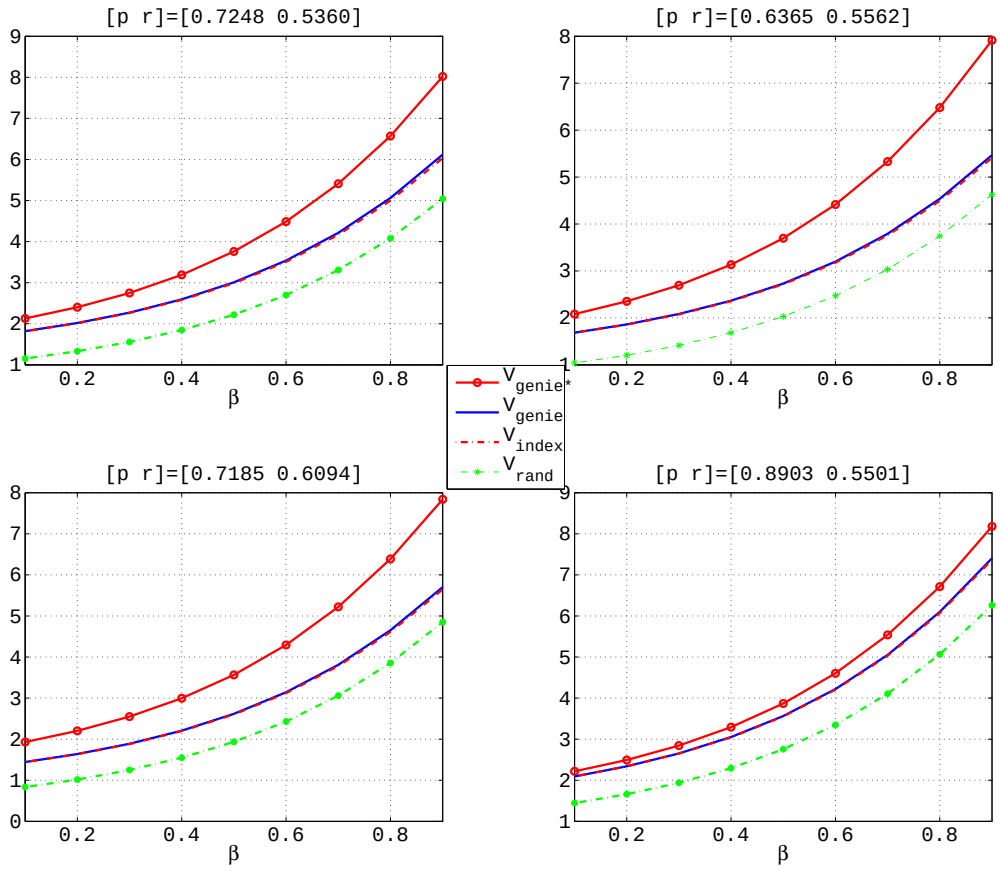

Figure 5.10: V_{genie^*}, V_{genie}, V_{index} and V_{rand} versus the discount factor β for various system parameters. Same set of initial belief values is used within each subplot.

5.6 Summary

In Chapter 3, we studied the ARQ based scheduling problem in a single cellular systems. In this chapter, we extended this analysis to the multi-cellular environment by adopting the cell breathing based ICI control mechanism. When the cooperation between the cells is asymmetric, the optimal scheduling policy has a greedy flavor and is simple to implement. Under symmetric cooperation, however, a direct optimality analysis appears difficult. We formulated the scheduling problem as a more general variant of the restless multiarmed bandit processes and studied it from the perspective of Whittle's indexability. Whittle's indexability is an important condition that is known to predispose the Whittle's index policy towards optimality in various RMAB processes. Founded on the indexability analysis of the two-cell scheduling problem, we proposed an easy-to-implement index policy that is near optimal. Upon Whittle's indexability of the two-cell scheduling problem, the policy we propose is essentially the Whittle index policy.

CHAPTER 6

CONCLUSIONS

With the ever increasing demand for limited network resources, there is a pressing need to design spectrally efficient communication techniques. Opportunistic multiuser scheduling is one among them. It is important that the channel state information required for the success of this technique be acquired in a cost-efficient manner, so that the loss involved here does not offset the gains associated with opportunistic scheduling.

6.1 Summary of Original Research

This dissertation identified mechanisms to exploit the memory inherent in fading channels to simultaneously acquire the channel state information while performing opportunistic scheduling. Thus data transmission in any time slot must take into account two potentially contradicting objectives: (1) Opportunistically schedule transmission - corresponds to immediate gain (2) Explore the channels for future scheduling purposes - corresponds to future gains. The joint scheduling is thus a dynamic program, specifically a partially observable Markov decision process that is traditionally known to be hard to solve in closed form and also computationally very expensive. Our contribution in this dissertation is to perform optimality

analysis of the joint scheduling problem in various networks, and whenever possible obtain the optimal scheduling policy in closed form. When such a closed form solution appeared intractable, founded on the optimality framework, we derived easy-to-implement scheduling policies that display near-optimal numerical performance.

Specifically, we first studied the joint scheduling problem in broadcast networks, with two-state Markov channels, when 1-bit feedback, delayed by one time slot, is available at the scheduler. The optimal policy turned out to be greedy for certain ranges of the system parameters. For the general values of the system parameters we followed an indirect approach. We first studied optimal scheduling in a two-user broadcast and established thresholdability properties of the optimal policy. Extrapolating these thresholdability properties to the general N user broadcast, we proposed a threshold policy. This policy has a polynomial complexity in the number of users and has near-optimal performance.

We then focused on using the already existing ARQ feedback mechanism for opportunistic scheduling in a single-cell downlink, with two-state Markov channels. We considered a general setup where the ARQ feedback is randomly delayed - a scenario possible in channels with severe propagation delays. The greedy policy that maximizes the immediate reward alone is optimal when the downlink has only two users. Surprisingly, this optimality result is independent of the distribution of the ARQ delay. However, when there are more than two downlink users, greedy policy is suboptimal, in general. We established this using an analytic counterexample. Numerical experiments suggest that the greedy policy has near-optimal performance. By studying the structural properties of the greedy policy, we obtained simple algorithms to implement the greedy policy. Turning our focus to the system level limits,

we obtained bounds on the capacity region of the cellular downlink with ARQ based scheduling, which we tightened for the two user case. We then proceeded to study the impact of increasing the state space of the Markov channels on the optimality properties of the greedy policy. When the Markov channel state space is increased to three, in the two-user downlink, greedy policy is not optimal. A crucial genie-equivalence that was seen in the two-state Markov modeled downlink is upset in the changed setup.

It can be expected that the dynamics of the joint channel estimation - opportunistic scheduling thus far seen in single cellular systems changes vastly in multi-cellular systems, where inter-cell interference imparts a convolved interdependence between scheduling choices in adjacent cells. We proceeded to study this changed dynamics in a two-cell system. We believe, this study could be readily extended to multi-cell systems with appropriate use of directional antennae. For the two-cell system, we studied the scheduling problem by following a two layered approach: the well established 'cell breathing' based inter-cell interference (ICI) control mechanism was adopted and assumed to be in place. On top of this layer we optimized the joint channel estimation - opportunistic scheduling based on ARQ feedback, across the cells. When the cooperation between the cells is asymmetric, the optimal policy has a greedy flavor and is simple to implement. Under symmetric cooperation, however, since a direct optimality analysis appears difficult, we formulated the scheduling problem as a more general variant of the restless multiarmed bandit processes and studied it from the perspective of Whittles indexability. By linking the indexability analysis to the broadcast scheduling problem studied in Chapter 2, we proposed an index policy that is easy to implement and has near-optimal performance.

In summary, the essential message of this dissertation is that by exploiting the memory inherent in the fading channels, significant system level performance gains can be achieved by opportunistic scheduling with minimal (delayed, 1 bit/ARQ) feedback. In addition, despite the POMDP nature of the scheduling problem, it is not necessary to perform computationally expensive optimal scheduling to realize these system level gains. Simple, practically convenient, suboptimal policies can achieve near-optimal performance.

6.2 Possible Future Research

The channel feedback is assumed to be error-free throughout this dissertation. This assumption may have limited application in realistic situations. It would be interesting to explore how the dynamics of the scheduling problem changes when the channel feedback is known to have stochastically defined imperfections.

For most part of this work, the fading channels are modeled with two-state Markov chains. With higher number of channel states, the scheduler can discriminate the channels at a finer level resulting in better scheduling gains. However, in order to limit the overhead in the feedback channel, the scheduler must be able discriminate the channels using feedback limited by the number of bits, that are probabilistically related to the actual channel states. A study of opportunistic scheduling under this setup is of practical value since this setup accommodates the use of ARQ feedback for scheduling purposes for a general size of the channel state space. We have already discussed the prevalence of ARQ in recent and upcoming wireless standards. Also note that this scenario bears similarities to the case of error-prone feedback discussed above.

It would also be an interesting exercise to study how the optimality/near-optimality properties of the greedy policy, reported in this dissertation, is affected when the fading channels are not positively correlated, or when the channels are not $i.i.d$ across users. Non-$i.i.d$ channels, in particular, is a realistic assumption that may impose Quality of Service (QoS) considerations on the scheduler, thus significantly changing the dynamics of the scheduling problem.

Random data arrivals at the network users is another important practical consideration that cannot be overlooked. With random arrivals, the scheduler has an added mandate of maintaining the stability of the user queues, likely without the knowledge of the arrival rates. It would be an interesting exercise to study how the buffer occupancy influences the scheduling decisions that have so far been purely based on the channel belief values.

Another potential direction for this research topic is when the assumption on TDMA-styled scheduling is removed. The TDMA model, i.e., one and only one user is scheduled in each slot, may have limited relevance in practical scenarios, compared to a constraint on the average number of users scheduled per slot. In the RMAB literature, it is common wisdom [23] that, Whittles index policy is optimal under an average constraint on the number of projects activated in a slot. Thus, under the 'average number of users' constraint, one may be able to earn a better understanding of the scheduling problem, by properly linking it to the existing results on RMAB processes.

APPENDIX A

PROOFS FOR CHAPTER 2

A.1 Proof of Lemma 3

Along the lines of proof of lemma 2, we proceed by reinterpreting the infinite horizon total discounted reward as a limit [36] on the finite horizon reward as below.

$$V(\pi_1, \pi_2) = \lim_{t \to \infty} V_t(\pi_1, \pi_2) \tag{A.1}$$

with

$$\begin{aligned} V_t(\pi_1, \pi_2) &= \max\{V_t^a(\pi_1, \pi_2), V_t^p(\pi_1, \pi_2)\} \\ V_t^a(\pi_1, \pi_2) &= \pi_1 + \pi_2 + \beta \sum_{j=0}^{2^N - 1} P_j(\pi) V_{t-1}(\Pi_j) \\ V_t^p(\pi_1, \pi_2) &= W + \beta V_{t-1}(T(\pi)). \end{aligned} \tag{A.2}$$

We have used the convention of decreasing time index up to the horizon at $t = 1$. The terminal reward, i.e., the reward at the horizon, is given by

$$V_1(\pi_1, \pi_2) = \max\{\pi_1 + \pi_2, W\}. \tag{A.3}$$

The finite horizon equivalent of the quantities γ and α are defined as

$$\begin{aligned} \gamma_t &= V_t(p, p) + V_t(r, r) - 2V_t(p, r) \\ \alpha_t &= V_t(p, r) - V_t(r, r). \end{aligned} \tag{A.4}$$

In the rest of this proof, we will study the reward functions over the two dimensional state space by sweeping over $\pi_1 \in [0,1]$ with π_2 along specific *directions/axes* given by $\pi_2 = \pi_1 k + c$ for $k, c \in \mathbb{R}$. For ease of notation, we denote the axis $\pi_2 = \pi_1 k + c$ by (k, c) and use π_2 and (k, c) interchangeably as arguments in reward functions. We now proceed to establish the proposition using induction.

Assume the following holds (induction hypothesis (H_1)): *For $t \geq 2$, $V_{t-1}(\pi_1, \pi_2)$ is convex and increasing in π_1 along (k, c) for $k \geq 0$. Along (k, c) for $k < 0$, V_{t-1} is piecewise concave. When $k = -1$, V_{t-1} attains a unique maximum at $\pi_1 = \pi_2 = \frac{c}{2}$.*

Along the axis $(k = 1, c = 0)$, i.e., $\pi_1 = \pi_2$, V_{t-1} is convex. Thus $V_{t-1}(p,p) + V_{t-1}(r,r) \geq 2V_{t-1}(\frac{p+r}{2}, \frac{p+r}{2})$. Since V_{t-1} has a maximum at $\pi_1 = \pi_2 = \frac{c}{2}$ along $\pi_2 = -\pi_1 + c$, we have, with $c = p + r$, $V_{t-1}(\frac{p+r}{2}, \frac{p+r}{2}) \geq V_{t-1}(p,r)$. Thus $\gamma_{t-1} \triangleq V_{t-1}(p,p) + V_{t-1}(r,r) - 2V_{t-1}(p,r) \geq 0$. Also, $\alpha_{t-1} = V_{t-1}(p,r) - V_{t-1}(r,r) \geq 0$ since V_{t-1} is increasing in π_1 along (k, c) for $k \geq 0$. From (A.2), V_t^a along the axis (k, c)) is given by,

$$
\begin{aligned}
& V_t^a(\pi_1, (k,c)) \\
&= \pi_1(1+k) + c + \beta\Big(\pi_1(\pi_1 k + c)V_{t-1}(p,p) + \pi_1(1 - \pi_1 k - c)V_{t-1}(p,r) \\
&\quad + (1-\pi_1)(\pi_1 k + c)V_{t-1}(r,p) + (1-\pi_1)(1 - \pi_1 k - c)V_{t-1}(r,r)\Big) \\
&= \pi_1(1+k) + c + \beta\Big(\pi_1^2 k\gamma_{t-1} + \pi_1\big(c\gamma_{t-1} + (k+1)\alpha_{t-1}\big) + c\alpha_{t-1} + V_{t-1}(r,r)\Big).
\end{aligned}
\tag{A.5}
$$

The second derivative of V_t^a is given by $2\beta k\gamma_{t-1}$. With $\beta \geq 0$ and $\gamma_{t-1} \geq 0$, V_t^a is convex in π_1 along (k, c) for $k \geq 0$ and concave in π_1 for $k < 0$. Also the first derivative of V_t^a with respect to π_1 along $k = 0, c \in [0,1]$, i.e., along $\pi_2 =$ constant is

given by

$$\begin{aligned}\frac{dV_t^a(\pi_1,(k,c))}{d\pi_1}|_{k=0,c\in[0,1]} &= (1+k)+\beta\Big(2\pi_1 k\gamma_{t-1}+\big(c\gamma_{t-1}+(k+1)\alpha_{t-1}\big)\Big)|_{k=0,c\in[0,1]} \\ &= 1+\beta(c\gamma_{t-1}+\alpha_{t-1})|_{c\in[0,1]}. \end{aligned} \tag{A.6}$$

Since $\alpha_{t-1}, \gamma_{t-1} \geq 0$ and $c \in [0,1]$, the first derivative of V_t^a, when π_2 is fixed, is positive and hence V_t^a is increasing in π_1 when π_2 is constant. Likewise, V_t^a is increasing in π_2 when π_1 is constant. Together, we have, V_t^a is increasing in π_1 along the axis (k,c) for $k \geq 0$.

From (A.2), V_t^p along (k,c) is given by,

$$\begin{aligned} V_t^p(\pi_1,(k,c)) &= W+\beta V_{t-1}\big(T(\pi_1),T(\pi_1 k+c)\big) \\ &= W+\beta V_{t-1}\big(T(\pi_1),T(\pi_1)k+c^*\big) \\ &= W+\beta V_{t-1}\big(T(\pi_1),(k,c^*)\big) \end{aligned} \tag{A.7}$$

where $c^* = c(p-r)+r(1-k)$. From the relationship between π_1 and $T(\pi_1)$, V_t^p along (k,c) is one-on-one and sequentially mapped[15] to V_{t-1} along the parallel axis (k,c^*). The second derivative of V_t^p is given by

$$\frac{d^2V_t^p(\pi_1,(k,c))}{d\pi_1^2}|_{\pi_1} = \beta(p-r)^2\frac{d^2V_{t-1}(\pi_1,(k,c^*))}{d{\pi_1}^2}|_{\pi_1(p-r)+r}. \tag{A.8}$$

When $k \geq 0$, since V_{t-1} is convex, $\frac{d^2V_{t-1}(\pi_1,(k,c^*))}{d{\pi_1}^2} \geq 0$. Thus V_t^p is convex in π_1 along (k,c) for $k \geq 0$. The first derivative is given by

$$\frac{dV_t^p(\pi_1,(k,c))}{d\pi_1}|_{\pi_1} = \beta(p-r)\frac{dV_{t-1}(\pi_1,(k,c^*))}{d\pi_1}|_{\pi_1(p-r)+r}. \tag{A.9}$$

Thus, V_{t-1} is increasing in π_1 along (k,c) for $k \geq 0$, V_t^p is also increasing in π_1 along (k,c) for $k \geq 0$.

[15]In addition to the passivity reward, W and the discount factor, β.

When $k < 0$, V_{t-1} is piecewise concave in π_1 along (k, c^*). Consider any interval $\pi_1 \in [x, y]$ in which $V_{t-1}(\pi_1, (k, c^*))$ is concave in π_1. From the second derivative relationship in (A.8), $V_t^p(\pi_1, (k, c))$ is concave in the interval $\pi_1 \in [T^{-1}(x), T^{-1}(y)]$, where $T^{-1}(z) = \frac{z-r}{p-r}$ is the inverse of the (one-step) evolution operator under *idle* decision, with $T^{-(k+1)}(z) = T^{-1}(T^{-k}(z))$. Thus, since V_t^p along (k, c) is one-on-one and sequentially mapped to V_{t-1} along the parallel axis (k, c^*), V_{t-1} is piecewise concave in π_1 along $(k, c^*) \Rightarrow V_t^p$ is also piecewise concave in π_1 along (k, c) for $k < 0$.

When $k = -1$, the mapping between V_t^p and V_{t-1} can be shown to be symmetric about the axis $\pi_1 = \pi_2$, i.e.,

$$\begin{aligned} V_t^p(\pi_1 = \frac{c}{2} + \delta, (-1, c)) &= W + \beta V_{t-1}(\pi_1 = \frac{c^*}{2} + \delta^*, (-1, c^*)) \\ V_t^p(\pi_1 = \frac{c}{2} - \delta, (-1, c)) &= W + \beta V_{t-1}(\pi_1 = \frac{c^*}{2} - \delta^*, (-1, c^*)) \end{aligned} \tag{A.10}$$

where $\delta, \delta^* \geq 0$ and $\delta = 0 \Leftrightarrow \delta^* = 0$. Thus, since $V_{t-1}(\pi_1, (-1, c^*))$ attains the unique maximum at $\pi_1 = \pi_2$, $V_t^p(\pi_1, (-1, c))$ also attains the maximum (also unique) at $\pi_1 = \pi_2 = \frac{c}{2}$.

Having obtained the structural properties of V_t^a and V_t^p under hypothesis (H_1), we now proceed to show that V_t satisfies the properties assumed for V_{t-1} in (H_1).

Recall the finite horizon total discounted reward function expression from (A.2);

$$V_t(\pi_1, \pi_2) = \max\{V_t^a(\pi_1, \pi_2), V_t^p(\pi_1, \pi_2)\}. \tag{A.11}$$

Along the axis (k, c), for $k \geq 0$, V_t^a and V_t^p are both convex and increasing in π_1. Thus V_t is also convex and increasing in π_1 for $k \geq 0$. For $k < 0$, V_t^a is concave and V_t^p is piecewise concave. Thus V_t is piecewise concave for $k < 0$ since the maximum of a concave and a piecewise concave function is piecewise concave. For $k = -1$,

V_t^a, being a concave and symmetric function in π_1, attains the unique maximum at $\pi_1 = \pi_2 = \frac{c}{2}$. Since V_t^p also attains the unique maximum at $\pi_1 = \pi_2 = \frac{c}{2}$, V_t attains the unique maximum at $\pi_1 = \pi_2 = \frac{c}{2}$. Now consider the terminal reward $V_1(\pi_1, \pi_2) = \max\{\pi_1 + \pi_2, W\}$. It can be readily seen that V_1 satisfies the induction hypothesis (H_1). It follows that, using induction, $V_t(\pi_1, \pi_2)$, for any t, satisfies the properties in (H_1). Using (A.1), the infinite horizon total discounted reward satisfies the properties identified in the lemma. Using the properties of V thus established and using arguments along the lines of the preceding induction based proof, the properties of V^a, V^p, γ and α identified in the lemma statement can also be established.

A.2 Proof of Proposition 3

Recall (proof of Lemma 3) the notion of studying reward functions by sweeping over π_1 with π_2 along specific *axes* $\pi_2 = \pi_1 k + c$ for $k, c \in \mathbb{R}$. We repeat this approach in this proof.

Consider the region R_I. This region can be covered by sweeping over $\pi_1 \geq \pi_{ss}$ along the axes $\pi_2 = \pi_1 k + \pi_{ss}(1-k)$, for $k \geq 0$. Note that these axes pass through the steady state (π_{ss}, π_{ss}). Note from lemma 6 that the evolution $(T(.))$ of a state along an axis passing through the steady state maps to a state on the same axis. Thus, since the axis $\pi_2 = \pi_1 k + \pi_{ss}(1-k)$ passes through the steady state, $(T(\pi_1), T(\pi_1 k + \pi_{ss}(1-k))) = (T(\pi_1), T(\pi_1)k + \pi_{ss}(1-k))$, i.e., the axis $(k, c) = (k, \pi_{ss}(1-k))$ is preserved upon $T(.)$ operation. This property is crucial in establishing the proposition. Call this property (E). Fix the axis $\pi_2 = \pi_1 k + \pi_{ss}(1-k)$, for $k \geq 0$, until noted otherwise. Now, by definition, $(\pi_{ss}, \pi_{ss}) \in \mathcal{A}$ iff $V^a(\pi_{ss}, \pi_{ss}) \geq V^p(\pi_{ss}, \pi_{ss})$. Denote (π_{ss}, π_{ss}) simply by π_{ss}. Consider the case $V(\pi_{ss}) = V^a(\pi_{ss}) = V^p(\pi_{ss})$. With

$V^p(\pi_{ss}) = W + \beta V(\pi_{ss}) = W + \beta V^p(\pi_{ss})$, we have $V^p(\pi_{ss}) = \frac{W}{1-\beta}$. Therefore $V(\pi_{ss}) = V^a(\pi_{ss}) = \frac{W}{1-\beta}$. We now proceed to show that $\exists$ a $\delta > 0$ such that, for $\pi_1 \in [\pi_{ss}, \pi_{ss}+\delta]$, $V^a(\pi_1, \pi_1 k + \pi_{ss}(1-k)) \geq V^p(\pi_1, \pi_1 k + \pi_{ss}(1-k))$. Assume the preceding statement is not true. Then $\exists$ a $\delta^* > 0$ such that, for $\pi_1 \in [\pi_{ss}, \pi_{ss} + \delta^*]$, $V^a(\pi_1, \pi_1 k + \pi_{ss}(1-k)) < V^p(\pi_1, \pi_1 k + \pi_{ss}(1-k))$. Therefore, for $\pi_1 \in [\pi_{ss}, \pi_{ss} + \delta^*]$, $V^p(\pi_1, \pi_1 k + \pi_{ss}(1-k)) = W + \beta V(T(\pi_1), T(\pi_1 k + \pi_{ss}(1-k))) = W + \beta V^p(T(\pi_1), T(\pi_1 k + \pi_{ss}(1-k)))$ since, from Lemma 4, $\pi_{ss} \leq T(\pi_1) \leq \pi_1 \leq \pi_{ss} + \delta^*$. Applying property (E) here, we have $V^p(\pi_1, \pi_1 k + \pi_{ss}(1-k)) = W + \beta V^p(T(\pi_1), T(\pi_1) k + \pi_{ss}(1-k))$. In the rest of this proof, since the axis $(k, \pi_{ss}(1-k))$ is preserved upon $T(.)$ operation, we skip the second argument in the reward functions. Now, recall that $V^a(\pi_1)$ is increasing in π_1 along the axis considered, i.e., when $k \geq 0$. Thus $V^a(\pi_1) \geq \frac{W}{1-\beta}$, $\forall \pi_1 \in [\pi_{ss}, 1]$. Thus $V^p(\pi_1) > \frac{W}{1-\beta}$ for $\pi_1 \in [\pi_{ss}, \pi_{ss} + \delta^*]$. This implies $V^p(T(\pi_1)) > \frac{W}{1-\beta}$ since $V^p(\pi_1) = W + \beta V^p(T(\pi_1))$. Using this argument recursively, $V^p(T^k(\pi_1)) > \frac{W}{1-\beta}$, for $k \geq 0$ when $\pi_1 \in [\pi_{ss}, \pi_{ss} + \delta^*]$. Thus $V^p(\pi_{ss}) > \frac{W}{1-\beta}$. This is a contradiction to the fact $V^p(\pi_{ss}) = \frac{W}{1-\beta}$. Thus, when $V^a(\pi_{ss}) = V^p(\pi_{ss})$, $\exists$ a $\delta > 0$ such that, for $\pi_1 \in [\pi_{ss}, \pi_{ss} + \delta]$, $V^a(\pi_1) \geq V^p(\pi_1)$. Now, consider the case $V(\pi_{ss}) = V^a(\pi_{ss}) > V^p(\pi_{ss})$. It can be trivially shown that V^a, V^p and V are continuous functions in the state space $(\pi_1, \pi_2) \in [0,1]^2$. Thus, with $V^a(\pi_{ss}) > V^p(\pi_{ss})$, $\exists$ a $\delta > 0$ such that, for $\pi_1 \in [\pi_{ss}, \pi_{ss} + \delta]$, $V^a(\pi_1) > V^p(\pi_1)$. Together, when $(\pi_{ss}) \in \mathcal{A}$, $\exists$ a $\delta > 0$ such that, for $\pi_1 \in [\pi_{ss}, \pi_{ss} + \delta]$, $V^a(\pi_1) \geq V^p(\pi_1)$. Note that $V^p(\pi_1) = W + \beta V(T(\pi_1))$. For $\pi_1 \in [\pi_{ss}, T^{-1}(\pi_{ss} + \delta)]$, we have $V^p(\pi_1) = W + \beta V(T(\pi_1)) = W + \beta V^a(T(\pi_1))$. Thus, for $\pi_1 \in [\pi_{ss}, T^{-1}(\pi_{ss} + \delta)]$, $V^p(\pi_1) = W + \beta V^a(T(\pi_1))$. The first derivative of V^p

with respect to π_1, for $\pi_1 \in [\pi_{ss}, T^{-1}(\pi_{ss}+\delta)]$, is now given by

$$\begin{aligned}
\frac{dV^p(\pi_1)}{d\pi_1}|_{\pi_1} &= \beta(p-r)\frac{d(V^a(\pi_1))}{d\pi_1}|_{\pi_1(p-r)+r} \\
&\leq \beta(p-r)\frac{d(V^a(\pi_1))}{d\pi_1}|_{\pi_1} \\
&\leq \frac{d(V^a(\pi_1))}{d\pi_1}|_{\pi_1}. \qquad \text{(A.12)}
\end{aligned}$$

Note that, by convexity of V^a in the axis considered (lemma 5), slope of $V^a(\pi_1)$ increases with π_1. This, along with the relation $\pi_{ss} \leq T(\pi_1) \leq \pi_1$ (lemma 6), is used in the second inequality.

Thus for $\pi_1 \in [\pi_{ss}, T^{-1}(\pi_{ss}+\delta)]$, $\frac{dV^p(\pi_1)}{d\pi_1} \leq \frac{dV^a(\pi_1)}{d\pi_1}$. This along with $V^a(\pi_{ss}) \geq V^p(\pi_{ss})$ leads to $V^a(\pi_1) \geq V^p(\pi_1)$ for $\pi_1 \in [\pi_{ss}, T^{-1}(\pi_{ss}+\delta)] = [\pi_{ss}, \frac{\pi_{ss}+\delta-r}{p-r}] = [\pi_{ss}, \pi_{ss}+\frac{\delta}{p-r}]$. By repeating the preceding arguments, for $\pi_1 \in [\pi_{ss}, T^{-l}(\pi_{ss}+\frac{\delta}{p-r})] = [\pi_{ss}, \pi_{ss}+\frac{\delta}{(p-r)^l}]$, recursively, for $l = 2, 3\ldots$, we have $V^a(\pi_1) \geq V^p(\pi_1)$, $\forall \pi_1 \geq \pi_{ss}$ along the axis $\pi_2 = \pi_1 k + \pi_{ss}(1-k)$, $k \geq 0$, i.e., $V^a(\pi_1) \geq V^p(\pi_1)$, $\forall (\pi_1, \pi_2) \in [\pi_{ss}, 1]^2$. This proves the first part of the proposition.

Consider the region R^1_{II}. We proceed to show thresholdability in this region when $\pi_{ss} \in \mathcal{A}$. Note that R^1_{II} is covered by sweeping over $\pi_1 \in [0, \pi_{ss}]$ along the axis $\pi_2 = \pi_1 k + \pi_{ss}(1-k)$, for $k \in [-1, 0]$. Consider one such axis, i.e., fix a $k \in [-1, 0]$ and $\pi_2 = \pi_1 k + \pi_{ss}(1-k)$ until noted otherwise. As before, since the axis $\pi_2 = \pi_1 k + \pi_{ss}(1-k)$ passes through the steady state, the evolution of any state along this axis maps to a state on the same axis. We therefore do not explicitly refer to π_2 or the axis in the subsequent analysis. The reward function V^a is now given as

$$\begin{aligned}
V^a(\pi_1) &= \pi_1(1+k) + \pi_{ss}(1-k) \\
&\quad + \beta\Big(\pi_1^2 k\gamma + \pi_1(\pi_{ss}(1-k)\gamma + \alpha(1+k)) + \pi_{ss}(1-k)\alpha + V_{rr}\Big)
\end{aligned} \qquad \text{(A.13)}$$

where the quantities γ and α are defined in section B. The first derivative of V^a with respect to π_1 is given as

$$\frac{dV^a(\pi_1)}{d\pi_1} = (1+k)[1+\beta\alpha] + \beta\gamma[\pi_{ss}(1-k) + 2\pi_1 k] \tag{A.14}$$

Note that since $\alpha \geq 0$, $\gamma \geq 0$ (lemma 5) and $k \in [-1, 0]$, $\frac{dV^a}{d\pi_1} \geq 0$ when $\pi_1 \leq \pi_{ss}$. With $\pi_{ss} \in \mathcal{A}$, we have either $V^a(\pi_{ss}) = V^p(\pi_{ss})$ or $V^a(\pi_{ss}) > V^p(\pi_{ss})$. In the former case, $V(\pi_{ss}) = V^a(\pi_{ss}) = V^p(\pi_{ss}) = \frac{W}{1-\beta}$. Since $\frac{dV^a(\pi_1)}{d\pi_1} \geq 0$, $V^a(\pi_1) \leq \frac{W}{1-\beta}$, for $\pi_1 \in [0, \pi_{ss}]$. Note that $V(\pi_1) \geq \frac{W}{1-\beta}$ for any π_1.[16] Thus $V^p(\pi_1) = W + \beta V(T(\pi_1)) \geq \frac{W}{1-\beta}$. Therefore, when $V^a(\pi_{ss}) = V^p(\pi_{ss})$, $\nexists$ a $\pi_1^* \in [0, \pi_{ss})$ such that $V^a(\pi_1^*) > V^p(\pi_1^*)$. Thus $V^a(\pi_1) < V^p(\pi_1)\ \forall\ \pi_1 \in [0, \pi_{ss})$. Thus $R_{II}^1 \in \mathcal{P}$ when $V^a(\pi_{ss}) = V^p(\pi_{ss})$. Arguing along similar lines, we have $R_{II}^2 \in \mathcal{P}$ when $V^a(\pi_{ss}) = V^p(\pi_{ss})$. This establishes the second part of the proposition.

Now, consider the case $V^a(\pi_{ss}) > V^p(\pi_{ss})$. Consider the case when $\exists$ a $\pi_1 \in [0, \pi_{ss})$ such that $V^a(\pi_1) = V^p(\pi_1)$. Let $\pi_1^* = \arg\max_{\pi_1 \in [0, \pi_{ss})}(V^a(\pi_1) = V^p(\pi_1))$. Then, since V^a and V^p are continuous functions, $V^a(\pi_1) \geq V^p(\pi_1)$, for $\pi_1 \in [\pi_1^*, \pi_{ss})$. Consider $\pi_1 \in [T^{-1}(\pi_1^*), \pi_{ss})$. The first derivative of V^p in this region is given by

$$\begin{aligned}
\frac{dV^p(\pi_1)}{d\pi_1} &= \beta(p-r)\frac{d(V^a(\pi_1))}{d\pi_1}|_{\pi_1(p-r)+r} \\
&\leq \beta(p-r)\frac{d(V^a(\pi_1))}{d\pi_1} \\
&\leq \frac{d(V^a(\pi_1))}{d\pi_1}.
\end{aligned} \tag{A.15}$$

Note that the first equality uses $V(T(\pi_1)) = V^a(T(\pi_1))$ when $\pi_1 \in [T^{-1}(\pi_1^*), \pi_{ss})$. The second inequality uses the fact that $\pi_1(p-r)+r \geq \pi_1$ when $\pi_1 \leq \pi_{ss}$ in addition to the concavity property of V^a along the axis considered (note $k \in [-1, 0]$). The

[16] Since the reward corresponding to the decision: inactivate at all times $= \frac{W}{1-\beta}$.

third inequality follows from the discussion alongside (A.14) that $\frac{d(V^a(\pi_1))}{d\pi_1} \geq 0$ along the considered axis. Thus $\frac{dV^p(\pi_1)}{d\pi_1} \leq \frac{d(V^a(\pi_1))}{d\pi_1}$ for $\pi_1 \in [T^{-1}(\pi_1^*), \pi_{ss})$. This, along with $V^a(\pi_1^*) = V^p(\pi_1^*) \Rightarrow V^a(\pi_1) \leq V^p(\pi_1)$ when $\pi_1 \in [T^{-1}(\pi_1^*), \pi_1^*]$. Now, consider $\pi_1 \in [T^{-2}(\pi_1^*), T^{-1}(\pi_1^*)]$. Thus $V^a(T(\pi_1) \leq V^p(T(\pi_1))$ since $T(\pi_1) \in [T^{-1}(\pi_1^*), \pi_1^*]$. Thus, the first derivative of V^p for $\pi_1 \in [T^{-2}(\pi_1^*), T^{-1}(\pi_1^*)]$ is given by

$$\begin{aligned}
\frac{dV^p(\pi_1)}{d\pi_1} &= \beta \frac{dV(T(\pi_1))}{d\pi_1} = \beta \frac{dV^p(T(\pi_1))}{d\pi_1} \\
&= \beta(p-r) \frac{d(V^p(\pi_1))}{d\pi_1}|_{\pi_1(p-r)+r} \\
&\leq \beta(p-r) \frac{d(V^a(\pi_1))}{d\pi_1}|_{\pi_1(p-r)+r} \\
&\leq \frac{d(V^a(\pi_1))}{d\pi_1}|_{\pi_1(p-r)+r} \\
&\leq \frac{d(V^a(\pi_1))}{d\pi_1} \qquad \text{(A.16)}
\end{aligned}$$

where we have used the results $V^a(x) \leq V^p(x)$ when $x \in [T^{-1}(\pi_1^*), \pi_1^*]$ in the first equality, $\frac{dV^p(\pi_1)}{d\pi_1} \leq \frac{d(V^a(\pi_1))}{d\pi_1}$ for $\pi_1 \in [T^{-1}(\pi_1^*), \pi_{ss})$ in the third inequality and the concavity property of V^a in the last inequality. Thus $\frac{dV^p(\pi_1)}{d\pi_1} \leq \frac{dV^a(\pi_1)}{d\pi_1}$ for $\pi_1 \in [T^{-2}(\pi_1^*), T^{-1}(\pi_1^*)]$. Since $V^a(\pi_1) \leq V^p(\pi_1)$ when $\pi_1 \in [T^{-1}(\pi_1^*), \pi_1^*]$, we now have $V^a(\pi_1) \leq V^p(\pi_1)$ when $\pi_1 \in [T^{-2}(\pi_1^*), T^{-1}(\pi_1^*)]$. Repeating the preceding arguments for $\pi_1 \in [T^{-(l+1)}(\pi_1^*), T^{-l}(\pi_1^*)]$ for $l = 2, 3 \ldots$, recursively, we have $V^a(\pi_1) \leq V^p(\pi_1)$ when $\pi_1 \in [0, \pi_1^*]$ and $V^a(\pi_1) > V^p(\pi_1)$ when $\pi_1 \in [\pi_1^*, \pi_{ss})$. If $\nexists$ a $\pi_1 \in [0, \pi_{ss})$ such that $V^a(\pi_1) = V^p(\pi_1)$, then since $V^a(\pi_{ss}) > V^p(\pi_{ss})$ and since V^a and V^p are continuous functions, $V^a(\pi_1) > V^p(\pi_1)$ for all $\pi_1 \in [0, \pi_{ss}]$. Region R_{II}^2 can be characterized along similar lines when $V^a(\pi_{ss}) > V^p(\pi_{ss})$.

The proposition thus follows.

A.3 Proof of Proposition 4

Along the lines of the proof of Proposition 3, we will traverse the two-dimensional state space by sweeping over π_1 with π_2 along specific axes $\pi_2 = \pi_1 k + \pi_{ss}(1-k)$ that pass through the steady state (π_{ss}, π_{ss}). Since the $T(.)$ operator preserves the axis, we skip the second argument in the reward functions throughout this proof. Also, denote the steady state simply by π_{ss}. Now, since the two-user broadcast is Type-II, we have $V^a(\pi_{ss}) < V^p(\pi_{ss})$. This implies $V(\pi_{ss}) = V^p(\pi_{ss}) = \frac{W}{1-\beta}$. Consider the axis $\pi_2 = \pi_1 k + \pi_{ss}(1-k)$, for $k \geq 0$. Along this axis, from Lemma 3, $V^a(\pi_1)$ is increasing in π_1. Thus, with $V^a(\pi_{ss}) < V^p(\pi_{ss}) = \frac{W}{1-\beta}$, we have, for $\pi_1 \leq \pi_{ss}$, $V^a(\pi_1) < \frac{W}{1-\beta}$. Since $V^p(\pi_1) \geq \frac{W}{1-\beta}$ for any π_1, we have $V^p(\pi_1) > V^a(\pi_1)$, $\forall\ \pi_1 \in [0, \pi_{ss}]$. Thus $(\pi_1, \pi_2) \in \mathcal{P}$, for $\pi_1 \leq \pi_{ss}$, $\pi_2 \leq \pi_{ss}$. Consider the axis $\pi_2 = \pi_1 k + \pi_{ss}(1-k)$, for $k \in [-1, 0]$. We now show that V^a is increasing in π_1 for $\pi_1 \in [0, \pi_{ss}]$. From the proof of lemma 5, the first derivative of V^a along an axis (k, c) is given by

$$\frac{dV^a(\pi_1, (k,c))}{d\pi_1} \;=\; (1+k) + \beta\Big(2\pi_1 k\gamma + \big(c\gamma + (k+1)\alpha\big)\Big) \tag{A.17}$$

Thus along $\pi_2 = \pi_1 k + \pi_{ss}(1-k)$, the slope, obtained by substituting $c = \pi_{ss}(1-k)$, is $(1+k) + \beta\big(\gamma(2k\pi_1 + \pi_{ss}(1-k)) + (1+k)\alpha\big)$. The slope is positive if $2k\pi_1 + \pi_{ss}(1-k) \geq 0$. This is true since $k \in [-1, 0]$ and $\pi_1 \in [0, \pi_{ss}]$. Thus, along the axis $\pi_2 = \pi_1 k + \pi_{ss}(1-k)$, for $k \in [-1, 0]$, $V^a(\pi_1)$ is increasing in π_1 for $\pi_1 \in [0, \pi_{ss}]$. Therefore, arguing along the lines of the $k \geq 0$ case, since $V^a(\pi_{ss}) < V^p(\pi_{ss})$, we have $V^a(\pi_1) < V^p(\pi_1)$ for $\pi_1 \in [0, \pi_{ss}]$ along the axis $\pi_2 = \pi_1 k + \pi_{ss}(1-k)$, for $k \in [-1, 0]$, i.e., when $\pi_1 \in [0, \pi_{ss}]$ and $\pi_{ss} \leq \pi_2 \leq -\pi_1 + 2\pi_{ss}$. By symmetry, $V^a(\pi_1) < V^p(\pi_1)$ for $\pi_2 \in [0, \pi_{ss}]$ and $\pi_{ss} \leq \pi_1 \leq -\pi_2 + 2\pi_{ss}$. This proves the first part of the proposition.

Consider the region R_I. Fix an axis $k \geq 0$ with $\pi_2 = \pi_1 k + \pi_{ss}(1-k)$. With $V^a(\pi_{ss}) < V^p(\pi_{ss})$, consider the case when $\exists$ a $\pi_1 \geq \pi_{ss}$ such that $V^a(\pi_1) = V^p(\pi_1)$. Let $\pi_1^* = \arg\min_{\pi_1 \geq \pi_{ss}}(V^a(\pi_1) = V^p(\pi_1))$. Therefore, with $V^a(\pi_{ss}) < V^p(\pi_{ss})$, $V^a(\pi_1) \leq V^p(\pi_1)$ for $\pi_1 \in [\pi_{ss}, \pi_1^*]$. Since $V(\pi_{ss}) = V^p(\pi_{ss})$, $V^p(\pi_{ss}) = \frac{W}{1-\beta}$. Thus $V^p(\pi_1) = \frac{W}{1-\beta}$ for $\pi_1 \in [\pi_{ss}, \pi_1^*]$. Consider $\pi_1 \in [\pi_1^*, T^{-1}(\pi_1^*)]$. Since $T(\pi_1) \in [\pi_{ss}, \pi_1^*]$, $V^p(\pi_1) = W + \beta V^p(T(\pi_1)) = \frac{W}{1-\beta}$. Since V^a is increasing in π_1 along the axis considered, with $V^a(\pi_1^*) = V^p(\pi_1^*)$ and $V^p(\pi_1) = V^p(\pi_1^*)$ for $\pi_1 \in [\pi_1^*, T^{-1}(\pi_1^*)]$, we have $V^a(\pi_1) \geq V^p(\pi_1)$ for $\pi_1 \in [\pi_1^*, T^{-1}(\pi_1^*)]$. Consider the first derivative of $V^p(\pi_1)$ for $\pi_1 \in [T^{-1}(\pi_1^*), T^{-2}(\pi_1^*)]$.

$$\begin{aligned} \frac{dV^p(\pi_1)}{d\pi_1}|_{\pi_1} &= \beta(p-r)\frac{d(V^a(\pi_1))}{d\pi_1}|_{\pi_1(p-r)+r} \\ &\leq \beta(p-r)\frac{d(V^a(\pi_1))}{d\pi_1}|_{\pi_1} \\ &\leq \frac{d(V^a(\pi_1))}{d\pi_1}|_{\pi_1} \end{aligned} \tag{A.18}$$

where the first equality comes from $T(\pi_1) \in [\pi_1^*, T^{-1}(\pi_1^*)]$ for $\pi_1 \in [\pi_1^*, T^{-1}(\pi_1^*)]$ and $V^a(\pi_1) \geq V^p(\pi_1)$ for $\pi_1 \in [\pi_1^*, T^{-1}(\pi_1^*)]$. The second inequality uses the convexity property of V^a in the axis considered. Inequality (A.18) along with $V^a(T^{-1}(\pi_1^*)) \geq V^p(T^{-1}(\pi_1^*))$ gives $V^a(\pi_1) \geq V^p(\pi_1)$ for $\pi_1 \in [T^{-1}(\pi_1^*), T^{-2}(\pi_1^*)]$. Repeating the preceding arguments for $\pi_1 \in [T^{-l}(\pi_1^*), T^{-(l+1)}(\pi_1^*)]$ for $l = 2, 3, \ldots$ recursively, we have $V^a(\pi_1) \geq V^p(\pi_1)$ for $\pi_1 \in [\pi_1^*, 1]$ and $V^a(\pi_1) < V^p(\pi_1)$ for $\pi_1 \in [\pi_{ss}, \pi_1^*)$. If $\nexists$ such a $\pi_1^* \in [\pi_{ss}, 1]$, then, since V^a and V^p are continuous functions with $V^a(\pi_{ss}) < V^p(\pi_{ss})$, we have $V^a(\pi_1) < V^p(\pi_1)\ \forall\ \pi_1 \in [\pi_{ss}, 1]$. The proposition thus follows.

APPENDIX B

PROOFS FOR CHAPTER 3

B.1 Proof of Lemma 5

Recall the definition of the $u-$step belief evolution operator: $T^u(x) = T(T^{(u-1)}(x)) = T^{(u-1)}(T(x))$ with $T(x) = xp + (1-x)r = x(p-r) + r$ and $T^0(x) = x$ for $x \in [0,1]$ and $u \in \{0,1,2,\ldots\}$. For $u \in \{1,2,\ldots\}$, $x \in [0,1]$,

$$\begin{aligned} T^u(p) &= T^{(u-1)}(p)p + (1 - T^{(u-1)}(p))r \\ T^{(u+1)}(x) &= T^u(x)p + (1 - T^u(x))r \\ T^u(p) - T^{(u+1)}(x) &= (p-r)(T^{(u-1)}(p) - T^u(x)). \end{aligned} \tag{B.1}$$

Thus if, for $u \in \{1,2,\ldots\}$, $T^{(u-1)}(p) - T^u(x) \geq 0$, then, since $p > r$, we have $T^u(p) - T^{(u+1)}(x) \geq 0$. By induction, using $p \geq T(x) = xp + (1-x)r$ for any $x \in [0,1]$, we have $T^u(p) \geq T^{u+1}(x)$ for any $u \in \{0,1,2,\ldots\}$ and $x \in [0,1]$. The second inequality in the lemma can be proved along the same lines using $r \leq T(x) = xp + (1-x)r$.

Consider the third inequality. By definition, for any $x, y \in [0,1]$, $T^u(x) - T^u(y) = (p-r)(T^{(u-1)}(x) - T^{(u-1)}(y))$. Thus, if $T^{(u-1)}(x) - T^{(u-1)}(y)$, then $T^u(x) - T^u(y) \geq 0$. When $x \geq y$, by induction, $T^u(x) - T^u(y) \geq 0$ for any $u \in \{0,1,2,\ldots\}$. This establishes the third inequality.

Considering the last inequality, the belief evolution operator can be expressed as

$$\begin{aligned} T^u(x) &= T(T^{(u-1)}(x)) = T(T(T^{(u-2)}(x))) \\ &= x(p-r)^u + r(\frac{1-(p-r)^u}{1-(p-r)}) \\ &= \frac{r}{1-(p-r)} + (p-r)^u\big(x - \frac{r}{1-(p-r)}\big) \end{aligned} \tag{B.2}$$

for $u \in \{0,1,2,\ldots\}$ and $x \in [0,1]$. Thus $T^u(p) = \frac{r}{1-(p-r)} + (p-r)^u\big[\frac{(p-r)(1-p)}{1-(p-r)}\big]$. Note that, since $p > r$, $T^u(p) \geq \frac{r}{1-(p-r)}$. Also, $T^u(r) = \frac{r}{1-(p-r)} - (p-r)^u\big[\frac{(p-r)r}{1-(p-r)}\big] \leq \frac{r}{1-(p-r)}$. This establishes the last inequality in the lemma.

B.2 Proof of Proposition 6

Let $N = 3$ users. Assume a deterministic ARQ delay of one time slot, i.e., $P_D(d=1) = 1$ and $P_D(d \neq 1) = 0$. Let $m = 4$ and the users be indexed in decreasing order of their initial beliefs, i.e., $\pi_m(1) \geq \pi_m(2) \geq \pi_m(3)$. The net expected reward corresponding to the greedy policy is given by

$$\begin{aligned} V_4(\pi_4, \{\hat{\mathfrak{A}}_k\}_{k=1}^4) &= \pi_4(1) + T(\pi_4(1)) \\ &\quad + \mathrm{E}_{f_4|\pi_4,a_4=1}[\hat{R}_2] + \mathrm{E}_{f_3,f_4|\pi_4,a_4=1,a_3=1}[\hat{R}_1] \end{aligned} \tag{B.3}$$

Note that since the delay is one slot, the first ARQ feedback comes at the end of slot 3. Thus, the greedy decision in both slots 4 and 3 is user 1. Also, the greedy scheduler has access to feedback f_4 only, at the beginning of slot 2 and both feedback f_4 and f_3, at the beginning of slot 1. Therefore, $\hat{R}_2$ is averaged over f_4 and $\hat{R}_1$ is averaged over f_4 *and* f_3. The average total reward under greedy policy can thus be evaluated by averaging over all realizations of f_4 and f_3. Table B.1 lists the belief values of the three users in slots 2 and 1 for various values of $\{f_4, f_3\}$ along with the greedy decisions and immediate rewards in slots 2 and 1. Note from the table that

the belief value π_2 at slot 2 is a function of f_4 only, while π_1 at slot 1 is a function of both f_4 and f_3, consistent with the preceding discussion.

The probabilities of occurrence of the various realizations of $\{f_4, f_3\}$ are summarized below

$$P(f_4, f_3) = \begin{cases} \pi_4(1)p, & \text{if } \{f_4, f_3\} = \{1, 1\} \\ \pi_4(1)(1-p), & \text{if } \{f_4, f_3\} = \{1, 0\} \\ (1-\pi_4(1))r, & \text{if } \{f_4, f_3\} = \{0, 1\} \\ (1-\pi_4(1))(1-r), & \text{if } \{f_4, f_3\} = \{0, 0\}. \end{cases} \tag{B.4}$$

Thus the net expected reward under the greedy policy is given by

$$\begin{aligned} V_4(\pi_4, \{\hat{\mathfrak{A}}_k\}_{k=1}^4) &= \pi_4(1) + T(\pi_4(1)) + \pi_4(1)p\big(2T(p)\big) \\ &\quad + \pi_4(1)(1-p)\big(T(p) + T^3(\pi_4(2))\big) \\ &\quad + (1-\pi_4(1))r\big(T^2(\pi_4(2)) + T(p)\big) \\ &\quad + (1-\pi_4(1))(1-r)\big(T^2(\pi_4(2)) + T^3(\pi_4(2))\big) \end{aligned} \tag{B.5}$$

Now, with a_k^* indicating the optimal decision in slot k, consider the following policy $\tilde{\mathfrak{A}}_k$ such that $\tilde{a}_4 = 1, \tilde{a}_3 = 2, \tilde{a}_2 = a_2^*, \tilde{a}_1 = a_1^*$. Since the ARQ delay is deterministic and equals one slot, the decision in slot 2 does not affect the reward in slot 1. Thus the greedy policy is optimal in slot 2. Trivially, greedy policy is optimal in slot 1, as well. Thus $a_2^* = \hat{a}_2$, $a_1^* = \hat{a}_1$. The average total reward under $\tilde{\mathfrak{A}}_k$ is given by

$$\begin{aligned} V_4(\pi_4, \{\tilde{\mathfrak{A}}_k\}_{k=1}^4) &= \pi_4(1) + T(\pi_4(2)) + \mathrm{E}_{f_4|\pi_4, a_4=1}[\tilde{R}_2] \\ &\quad + \mathrm{E}_{f_3, f_4|\pi_4, a_4=1, a_3=2}[\tilde{R}_1] \\ &= \pi_4(1) + T(\pi_4(2)) + \mathrm{E}_{f_4|\pi_4, a_4=1}[\hat{R}_2] \\ &\quad + \mathrm{E}_{f_3, f_4|\pi_4, a_4=1, a_3=2}[\hat{R}_1] \end{aligned} \tag{B.6}$$

We evaluate $V_4(\pi_4, \{\tilde{\mathfrak{A}}_k\}_{k=1}^4)$ along the lines of the greedy net expected reward evaluation. Table B.2 summarizes the beliefs, scheduling decision $\tilde{a}_k$ and immediate rewards

$\{f_4, f_3\}$	π_2 at slot 2	$\hat{a}_2$	$\hat{R}_2$	π_1 at slot 1	$\hat{a}_1$	$\hat{R}_1$
{1,1}	$\begin{bmatrix} T(p) \\ T^2(\pi_4(2)) \\ T^2(\pi_4(3)) \end{bmatrix}$	1	$T(p)$	$\begin{bmatrix} T(p) \\ T^3(\pi_4(2)) \\ T^3(\pi_4(3)) \end{bmatrix}$	1	$T(p)$
{1,0}	$\begin{bmatrix} T(p) \\ T^2(\pi_4(2)) \\ T^2(\pi_4(3)) \end{bmatrix}$	1	$T(p)$	$\begin{bmatrix} T(r) \\ T^3(\pi_4(2)) \\ T^3(\pi_4(3)) \end{bmatrix}$	2	$T^3(\pi_4(2))$
{0,1}	$\begin{bmatrix} T(r) \\ T^2(\pi_4(2)) \\ T^2(\pi_4(3)) \end{bmatrix}$	2	$T^2(\pi_4(2))$	$\begin{bmatrix} T(p) \\ T^3(\pi_4(2)) \\ T^3(\pi_4(3)) \end{bmatrix}$	1	$T(p)$
{0,0}	$\begin{bmatrix} T(r) \\ T^2(\pi_4(2)) \\ T^2(\pi_4(3)) \end{bmatrix}$	2	$T^2(\pi_4(2))$	$\begin{bmatrix} T(r) \\ T^3(\pi_4(2)) \\ T^3(\pi_4(3)) \end{bmatrix}$	2	$T^3(\pi_4(2))$

Table B.1: Belief values, scheduling decisions, immediate rewards in slots 2 and 1 for various realizations of ARQ feedback under the greedy policy.

$\{f_4, f_3\}$	π_2 at slot 2	$\tilde{a}_2$	$\tilde{R}_2$	π_1 at slot 1	$\tilde{a}_1$	$\tilde{R}_1$
{1,1}	$\begin{bmatrix} T(p) \\ T^2(\pi_4(2)) \\ T^2(\pi_4(3)) \end{bmatrix}$	1	$T(p)$	$\begin{bmatrix} T^2(p) \\ T(p) \\ T^3(\pi_4(3)) \end{bmatrix}$	2	$T(p)$
{1,0}	$\begin{bmatrix} T(p) \\ T^2(\pi_4(2)) \\ T^2(\pi_4(3)) \end{bmatrix}$	1	$T(p)$	$\begin{bmatrix} T^2(p) \\ T(r) \\ T^3(\pi_4(3)) \end{bmatrix}$	1	$T^2(p)$
{0,1}	$\begin{bmatrix} T(r) \\ T^2(\pi_4(2)) \\ T^2(\pi_4(3)) \end{bmatrix}$	2	$T^2(\pi_4(2))$	$\begin{bmatrix} T^2(r) \\ T(p) \\ T^3(\pi_4(3)) \end{bmatrix}$	2	$T(p)$
{0,0}	$\begin{bmatrix} T(r) \\ T^2(\pi_4(2)) \\ T^2(\pi_4(3)) \end{bmatrix}$	2	$T^2(\pi_4(2))$	$\begin{bmatrix} T^2(r) \\ T(r) \\ T^3(\pi_4(3)) \end{bmatrix}$	3	$T^3(\pi_4(3))$

Table B.2: Belief values, scheduling decisions, immediate rewards in slots 2 and 1 for various realizations of ARQ feedback under policy $\tilde{\mathfrak{A}}_k$.

in slots 2 and 1 for all the realizations of $\{f_4, f_3\}$ when $\{\tilde{a}_4, \tilde{a}_3\} = \{1, 2\}$. Users are once again ordered according to their initial belief values, i.e., $\pi_4(1) \geq \pi_4(2) \geq \pi_4(3)$. Note from the table that the belief value π_2 at slot 2 is a function of f_4 only, while π_1 at slot 1 is a function of both f_4 and f_3, consistent with the ARQ delay profile.

The probabilities of occurrence of the various realizations of $\{f_4, f_3\}$ when $a_4 = 1, a_3 = 2$, are summarized below.

$$P(f_4, f_3) = \begin{cases} \pi_4(1)T(\pi_4(2)), & \text{if } \{f_4, f_3\} = \{1, 1\} \\ \pi_4(1)(1 - T(\pi_4(2))), & \text{if } \{f_4, f_3\} = \{1, 0\} \\ (1 - \pi_4(1))T(\pi_4(2)), & \text{if } \{f_4, f_3\} = \{0, 1\} \\ (1 - \pi_4(1))(1 - T(\pi_4(2))), & \text{if } \{f_4, f_3\} = \{0, 0\}. \end{cases} \tag{B.7}$$

p	r	π_4	$V_4(\pi_4, \{\tilde{\mathfrak{A}}\}_{k=1}^4)$	$V_4(\pi_4, \{\hat{\mathfrak{A}}\}_{k=1}^4)$	$V_4(\pi_4, \{\tilde{\mathfrak{A}}\}_{k=1}^4)$ $-V_4(\pi_4, \{\hat{\mathfrak{A}}\}_{k=1}^4)$
0.9308	0.1797	$\begin{bmatrix}0.5216\\0.5130\\0.3305\end{bmatrix}$	2.6368	2.6141	0.0227
0.8875	0.0186	$\begin{bmatrix}0.3416\\0.3310\\0.2648\end{bmatrix}$	1.6155	1.5454	0.0701

Table B.3: Sample system parameters when the greedy policy is suboptimal. Number of users $N = 3$, deterministic delay $D = 1$, horizon $m = 4$ is used.

Thus, the net expected reward under policy $\tilde{\mathfrak{A}}_k$ is given by

$$\begin{aligned} V_4(\pi_4, \{\tilde{\mathfrak{A}}\}_{k=1}^4) &= \pi_4(1) + T(\pi_4(2)) + \pi_4(1)T(\pi_4(2))\big(2T(p)\big) \\ &\quad +\pi_4(1)(1 - T(\pi_4(2)))\big(T(p) + T^2(p)\big) \\ &\quad +(1 - \pi_4(1))T(\pi_4(2))\big(T^2(\pi_4(2)) + T(p)\big) \\ &\quad +(1 - \pi_4(1))(1 - T(\pi_4(2)))\big(T^2(\pi_4(2)) + T^3(\pi_4(3))\big) \end{aligned} \tag{B.8}$$

We now proceed to show that, for $N = 3$, deterministic ARQ delay $D = 1$ and horizon $m = 4$, $\exists\ p, r, \pi_4$ such that the net expected reward corresponding to policy $\tilde{\mathfrak{A}}_k$ is strictly higher than that of the greedy policy. The difference in reward, after

algebraic manipulations is given by

$$\begin{aligned} & V_4(\pi_4, \{\tilde{\mathfrak{A}}\}_{k=1}^4) - V_4(\pi_4, \{\hat{\mathfrak{A}}\}_{k=1}^4) \\ = \;& (p-r)\Big(\pi_4(2) - \pi_4(1) \\ & +(p-r)^2(1-\pi_4(1))\pi_4(3)\big(1-r-(p-r)\pi_4(2)\big)\Big). \end{aligned} \tag{B.9}$$

For the special case $\pi_4(1) = \pi_4(2) = \frac{1}{2}$, we have

$$\begin{aligned} & V_4(\pi_4, \{\tilde{\mathfrak{A}}\}_{k=1}^4) - V_4(\pi_4, \{\hat{\mathfrak{A}}\}_{k=1}^4) \\ = \;& \frac{(p-r)^3}{2}\Big(1 - \frac{p+r}{2}\Big)\pi_4(3) \end{aligned} \tag{B.10}$$

For any $p < 1$, since $p > r$, $V_4(\pi_4, \{\tilde{\mathfrak{A}}\}_{k=1}^4) > V_4(\pi_4, \{\hat{\mathfrak{A}}\}_{k=1}^4)\ \forall\ \pi_4(3) > 0$. With the net expected reward of the optimal policy being no less than $V_4(\pi_4, \{\tilde{\mathfrak{A}}\}_{k=1}^4)$, we see that the greedy policy is not in general optimal. Table B.3 lists a few other values of p, r, π_4 for which the greedy policy is suboptimal. This establishes the proposition.

APPENDIX C

PROOFS FOR CHAPTER 4

C.1 Proof of Lemma 6

Let $\beta = [\beta_1\ \beta_2\ \beta_3]^T$, with $\beta_1 \leq \beta_2 \leq \beta_3$. Consider the inequality $\mathbf{p}_3\beta \geq \mathbf{p}_2\beta$. This can be rewritten as,

$$\begin{aligned} \beta_1 p_{31} + \beta_2 p_{32} + \beta_3 p_{33} &\geq \beta_1 p_{21} + \beta_2 p_{22} + \beta_3 p_{23} \\ \Leftrightarrow \beta_1(p_{31} - p_{21}) &\geq \beta_2(p_{22} - p_{32}) + \beta_3(p_{23} - p_{33}) \\ \Leftrightarrow \beta_1(p_{21} - p_{31}) &\leq -\beta_2(p_{22} - p_{32}) + \beta_3(p_{33} - p_{23}) \end{aligned} \tag{C.1}$$

Since $\beta_2 \geq \beta_1$, it is now sufficient to prove $\beta_2(p_{21} - p_{31} + p_{22} - p_{32}) \leq \beta_3(p_{33} - p_{23})$, i.e., $\beta_2(p_{33} - p_{23}) \leq \beta_3(p_{33} - p_{23})$ which is indeed true. Consider the inequality $\mathbf{p}_2\beta \geq \mathbf{p}_1\beta$,

$$\begin{aligned} \beta_1 p_{21} + \beta_2 p_{22} + \beta_3 p_{23} &\geq \beta_1 p_{11} + \beta_2 p_{12} + \beta_3 p_{13} \\ \Leftrightarrow \beta_2 + p_{23}(\beta_3 - \beta_2) - p_{21}(\beta_2 - \beta_1) &\geq \beta_2 + p_{13}(\beta_3 - \beta_2) - p_{11}(\beta_2 - \beta_1) \end{aligned} \tag{C.2}$$

The last inequality is indeed true, since $p_{23} \geq p_{13}$, $p_{21} \leq p_{11}$ and $\beta_3 \geq \beta_2 \geq \beta_1$. Thus if $\beta_1 \leq \beta_2 \leq \beta_3$ and $\beta = [\beta_1\ \beta_2\ \beta_3]^T$,

$$\mathbf{p}_3\beta \geq \mathbf{p}_2\beta \geq \mathbf{p}_1\beta \tag{C.3}$$

We can write, for $i \in 1,2,3$, $\mathbf{p}_i P^{k+1}\alpha = \mathbf{p}_i[\mathbf{p}_1 P^k\alpha \ \mathbf{p}_2 P^k\alpha \ \mathbf{p}_3 P^k\alpha]^T$. Thus if $\mathbf{p}_1 P^k\alpha \le \mathbf{p}_2 P^k\alpha \le \mathbf{p}_3 P^k\alpha$, we have, using (C.3), $\mathbf{p}_1 P^{k+1}\alpha \le \mathbf{p}_2 P^{k+1}\alpha \le \mathbf{p}_3 P^{k+1}\alpha$. Since $\alpha_1 = 0 \le \alpha_2 \le \alpha_3 = 1$, the lemma is established using induction.

C.2 Proof of Lemma 7 and Lemma 8

Consider $\mathbf{p}_3 P^{k+1}\alpha = p_{31}\mathbf{p}_1 P^k\alpha + p_{32}\mathbf{p}_2 P^k\alpha + p_{33}\mathbf{p}_3 P^k\alpha$. Since $\mathbf{p}_1 P^k\alpha \le \mathbf{p}_2 P^k\alpha \le \mathbf{p}_3 P^k\alpha$ from Lemma 6, we have $\mathbf{p}_3 P^{k+1}\alpha \le \mathbf{p}_3 P^k\alpha$. Lemma 8 can be proved similarly.

C.3 Proof of Lemma 9

Let $\mathbf{p}_2 P^k[001]^T \le \mathbf{p}_2 P^{k-1}[001]^T$. Multiplying both sides by p_{22} and adding to both sides $p_{21}\mathbf{p}_1 P^{k-1}[001]^T + p_{23}\mathbf{p}_3 P^{k-1}[001]^T$,

$$p_{21}\mathbf{p}_1 P^{k-1}[001]^T + p_{22}\mathbf{p}_2 P^k[001]^T + p_{23}\mathbf{p}_3 P^{k-1}[001]^T \ \le \ \mathbf{p}_2 P^k[001]^T \quad \text{(C.4)}$$

If we show that $p_{21}\mathbf{p}_1 P^k[001]^T + p_{23}\mathbf{p}_3 P^k[001]^T \le p_{21}\mathbf{p}_1 P^{k-1}[001]^T + p_{23}\mathbf{p}_3 P^{k-1}[001]^T$, then using (C.4), $p_{21}\mathbf{p}_1 P^k[001]^T + p_{22}\mathbf{p}_2 P^k[001]^T + p_{23}\mathbf{p}_3 P^k[001]^T \le \mathbf{p}_2 P^k[001]^T$, i.e, $\mathbf{p}_2 P^{k+1}[001]^T \le \mathbf{p}_2 P^k[001]^T$. Consider the inequality

$$\begin{aligned}
p_{21}\mathbf{p}_1 P^k[001]^T + p_{23}\mathbf{p}_3 P^k[001]^T &\le p_{21}\mathbf{p}_1 P^{k-1}[001]^T + p_{23}\mathbf{p}_3 P^{k-1}[001]^T \\
\Leftrightarrow \mathbf{p}_2 P^{k+1}[001]^T - p_{22}\mathbf{p}_2 P^k[001]^T &\le \mathbf{p}_2 P^k[001]^T - p_{22}\mathbf{p}_2 P^{k-1}[001]^T \\
\Leftrightarrow \mathbf{p}_2(P^{k+1}[001]^T - P^k[001]^T) &\le p_{22}(\mathbf{p}_2 P^k[001]^T - \mathbf{p}_2 P^{k-1}[001]^T) \\
\Rightarrow \mathbf{p}_2 P^{k+1}[001]^T &\le \mathbf{p}_2 P^k[001]^T \qquad \text{(C.5)}
\end{aligned}$$

where the last inequality is from the initial assumption that $\mathbf{p}_2 P^k[001]^T - \mathbf{p}_2 P^{k-1}[001]^T \le 0$.

With $\mathbf{p}_2 P^1[001]^T \le \mathbf{p}_2 P^0[001]^T$, i.e, $\mathbf{p}_2 P[001]^T \le p_{23}$, using induction, we have the $\mathbf{p}_2 P^{k+1}[001]^T \le \mathbf{p}_2 P^k[001]^T \ \forall k \ge 0$. Since steady state exists, by the definition

of steady state, $\lim_{k\to\infty} P^k = \begin{bmatrix} \pi_{ss} \\ \pi_{ss} \\ \pi_{ss} \end{bmatrix}$. Thus $\mathbf{p}_2 \lim_{k\to\infty} P^k[001]^T = \pi_{ss}(3)$ and $\pi_{ss}(3) \leq p_{23}$ by the monotonic decrease property of $\mathbf{p}_2 P^k[001]^T$. Also note that the direction of the inequalities throughout this proof can be changed and we can prove that $\mathbf{p}_2 P^k[001]^T$ monotonically *increases* to $\pi_{ss}(3)$ as $k \to \infty$ if $\mathbf{p}_2 P[001]^T \geq p_{23}$. This establishes that $\mathbf{p}_2 P[001]^T \leq p_{23}$ is a necessary condition for the first part of the Lemma to hold.

C.4 Proof of Proposition 15

Let the probability transition matrix satisfy the following conditions:

$$p_{12} = p_{22} = p_{32} \tag{C.6}$$

$$p_{23}p_{31} \geq p_{21}p_{13} \tag{C.7}$$

The preceding inequality along with condition (C.6) is equivalent to condition (A) in Lemma 9. Thus under (C.6) and (C.7), both Lemma 9 and Lemma 10 hold true. From Lemma 9, $p_{23} \geq \pi_{ss}(3)$. From (C.6), $\pi_{ss}(2) = p_{22}$. Thus $p_2\alpha - \pi_{ss}\alpha = p_{22}\alpha_2 + p_{23} - \pi_{ss}(2)\alpha_2 - \pi_{ss}(3) = p_{23} - \pi_{ss}(3) \geq 0$. The system is thus type I.

Consider a control interval $m > 1$ with belief vectors $\pi_{m,1}$, $\pi_{m,2}$ and action a_m. If we can show for any m that, assuming the greedy policy will be implemented in all the future control intervals, the greedy policy is optimal in control interval m, then using induction from interval 1, where greedy is indeed optimal, we could establish the long term optimality of the greedy policy. Let $\{\mathfrak{A}_k\}_{k\leq m-1} = \{\widehat{\mathfrak{A}}_k\}_{k\leq m-1}$and let S_k be the state vector such that $S_k(i)$ is the state of the channel of user i in interval k.

We rewrite the net expected reward as follows

$$V_m(\pi_{m,1}, \pi_{m,2}, \{a_m, \{\widehat{\mathfrak{A}}_k\}_{k \leq m-1}\}) = \pi_{m,a_m}\alpha + \sum_{S_m} P_{S_m|\pi_{m,1},\pi_{m,2}}(S_m|\pi_{m,1}, \pi_{m,2})\hat{V}_{m-1}(S_m, \hat{a}_{m-1}),$$

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 $k \leq m-1$. $P_{S_m|\pi_{m,1},\pi_{m,2}}(S_m|\pi_{m,1}, \pi_{m,2})$ is the conditional probability of the current state vector S_m given the belief vectors $\pi_{m,1}$, $\pi_{m,2}$. The scheduling decision in the next control interval, $\hat{a}_{m-1}$, is based on the greedy policy and is a function of the ARQ feedback received in the current control interval k, i.e., $S_m(a_m)$. The decision logic was summarized in Proposition 10. We now proceed to compare the net expected reward when $a_m = 1$ and $a_m = 2$. The net expected reward when $a_m = 1$ is written as follows,

$$
\begin{aligned}
&V_m(\pi_{m,1}, \pi_{m,2}, \{a_m = 1, \{\widehat{\mathfrak{A}}_k\}_{k \le m-1}\}) \\
&= \pi_{m,1}\alpha + P_{S_m|\pi_{m,1},\pi_{m,2}}(S_m = [1\ 1]|\pi_{m,1}, \pi_{m,2})\hat{V}_{m-1}\big(S_m = [1\ 1], \hat{a}_{m-1} = 2\big) \\
&\quad + P_{S_m|\pi_{m,1},\pi_{m,2}}(S_m = [1\ 2]|\pi_{m,1}, \pi_{m,2})\hat{V}_{m-1}\big(S_m = [1\ 2], \hat{a}_{m-1} = 2\big) \\
&\quad + P_{S_m|\pi_{m,1},\pi_{m,2}}(S_m = [1\ 3]|\pi_{m,1}, \pi_{m,2})\hat{V}_{m-1}\big(S_m = [1\ 3], \hat{a}_{m-1} = 2\big) \\
&\quad + P_{S_m|\pi_{m,1},\pi_{m,2}}(S_m = [2\ 1]|\pi_{m,1}, \pi_{m,2})\hat{V}_{m-1}\big(S_m = [2\ 1], \hat{a}_{m-1} = 1\big) \\
&\quad + P_{S_m|\pi_{m,1},\pi_{m,2}}(S_m = [2\ 2]|\pi_{m,1}, \pi_{m,2})\hat{V}_{m-1}\big(S_m = [2\ 2], \hat{a}_{m-1} = 1\big) \\
&\quad + P_{S_m|\pi_{m,1},\pi_{m,2}}(S_m = [2\ 3]|\pi_{m,1}, \pi_{m,2})\hat{V}_{m-1}\big(S_m = [2\ 3], \hat{a}_{m-1} = 1\big) \\
&\quad + P_{S_m|\pi_{m,1},\pi_{m,2}}(S_m = [3\ 1]|\pi_{m,1}, \pi_{m,2})\hat{V}_{m-1}\big(S_m = [3\ 1], \hat{a}_{m-1} = 1\big) \\
&\quad + P_{S_m|\pi_{m,1},\pi_{m,2}}(S_m = [3\ 2]|\pi_{m,1}, \pi_{m,2})\hat{V}_{m-1}\big(S_m = [3\ 2], \hat{a}_{m-1} = 1\big) \\
&\quad + P_{S_m|\pi_{m,1},\pi_{m,2}}(S_m = [3\ 3]|\pi_{m,1}, \pi_{m,2})\hat{V}_{m-1}\big(S_m = [3\ 3], \hat{a}_{m-1} = 1\big) \qquad \text{(C.8)}
\end{aligned}
$$

Note that the scheduler uses the information of the state of the scheduled user (user 1) alone in the scheduling decisions, consistent with the problem setup. Also note that when $S_m(1) = 2$, the schedule is retained. This is consistent with the implementation structure of the greedy policy seen in Proposition 10, where the scheduler *retains the scheduling choice even F_2 is received*. As was discussed in the same proposition, this is a greedy decision only if an user was never dropped in the past for giving feedback F_3. Since we are restricting to the class of schedulers that retains the schedule when F_3 is satisfied[17], this is indeed a greedy decision. Since the Markov channel statistics are identical across the users, we have $\hat{V}_k\big(S_{k+1} = [x\ y], \hat{a}_k = 1]\big) = \hat{V}_k\big(S_{k+1} = [y\ x], \hat{a}_k = 2]\big)$. Expanding the net expected reward when $a_m = 2$ along the lines of (C.8) and

[17]This is the only instance in the proof where we constrain the search space.

using the preceding symmetry property, we have,

$$
\begin{aligned}
& V_m(\pi_{m,1}, \pi_{m,2}, \{a_m = 1, \{\widehat{\mathfrak{A}}_k\}_{k \le m-1}\}) - V_m(\pi_{m,1}, \pi_{m,2}, \{a_m = 2, \{\widehat{\mathfrak{A}}_k\}_{k \le m-1}\}) \\
&= \pi_{m,1}\alpha - \pi_{m,2}\alpha \\
&\quad + \Big[\hat{V}_{m-1}\big(S_m = [3\ 2], \hat{a}_{m-1} = 1\big) - \hat{V}_{m-1}\big(S_m = [2\ 3], \hat{a}_{m-1} = 1\big)\Big] \times \\
&\quad \Big[\pi_{m,1}(3)\pi_{m,2}(2) - \pi_{m,1}(2)\pi_{m,2}(3)\Big] \qquad \text{(C.9)}
\end{aligned}
$$

Let $\hat{a}_m$ indicate the greedy choice among the users in the current control interval, i.e., $\hat{a}_m = \arg\max_{i \in 1,2} R_m(\pi_{m,i})$. Let $\tilde{a}_m$ indicate the other user. The net expected reward can now be rewritten as,

$$
\begin{aligned}
& V_m(\pi_{m,1}, \pi_{m,2}, \{a_m = \hat{a}_m, \{\widehat{\mathfrak{A}}_k\}_{k \le m-1}\}) - V_m(\pi_{m,1}, \pi_{m,2}, \{a_m = \tilde{a}_m, \{\widehat{\mathfrak{A}}_k\}_{k \le m-1}\}) \\
&= \pi_{m,\hat{a}_m}\alpha - \pi_{m,\tilde{a}_m}\alpha \\
&\quad + \Big[\hat{V}_{m-1}\big(S_m = [3\ 2], \hat{a}_{m-1} = 1\big) - \hat{V}_{m-1}\big(S_m = [2\ 3], \hat{a}_{m-1} = 1\big)\Big] \times \\
&\quad \Big[\pi_{m,\hat{a}_m}(3)\pi_{m,\tilde{a}_m}(2) - \pi_{m,\hat{a}_m}(2)\pi_{m,\tilde{a}_m}(3)\Big] \qquad \text{(C.10)}
\end{aligned}
$$

where, by definition, $\pi_{m,\hat{a}_m}\alpha \ge \pi_{m,\tilde{a}_m}\alpha$. We now proceed to show that the quantity $\hat{V}_{m-1}\big(S_m = [3\ 2], \hat{a}_{m-1} = 1\big) - \hat{V}_{m-1}\big(S_m = [2\ 3], \hat{a}_{m-1} = 1\big)$ is non-negative. With $\hat{V}_k\big(S_{k+1} = [x\ y]\big) := \hat{V}_k\big(S_{k+1} = [x\ y], \hat{a}_k = 1\big)$, and expanding $\hat{V}_{m-1}\big(S_m = [x\ y]\big)$ along the lines of (C.8) with $\pi_{m-1,1} = \mathbf{p}_x$ and $\pi_{m-1,2} = \mathbf{p}_y$ and $a_{m-1} = 1$, we have the following.

$$
\begin{aligned}
& \hat{V}_{m-1}\big(S_m = [3\ 2]\big) - \hat{V}_{m-1}\big(S_m = [2\ 3]\big) \\
&= \mathbf{p}_3\alpha - \mathbf{p}_2\alpha + \Big[\hat{V}_{m-2}\big(S_{m-1} = [3\ 2]\big) - \hat{V}_{m-2}\big(S_{m-1} = [2\ 3]\big)\Big](p_{33}p_{22} - p_{23}p_{32}) \\
& \qquad \text{(C.11)}
\end{aligned}
$$

By the property of the P matrix, $p_{33} \ge p_{23}$ and $p_{22} \ge p_{32}$. Also, we have seen in Lemma3 that $r_3 \ge r_2 \ge r_1$. Thus if $\hat{V}_{m-2}\big(S_{m-1} = [3\ 2]\big) - \hat{V}_{m-2}\big(S_{m-1} = [2\ 3]\big) \ge$

0, then $\hat{V}_{m-1}\big(S_m = [3\ 2]\big) - \hat{V}_{m-1}\big(S_m = [2\ 3]\big) \geq 0$. Expanding $\hat{V}_{m-2}\big(S_{m-1} = [3\ 2]\big) - \hat{V}_{m-2}\big(S_{m-1} = [2\ 3]\big) \geq 0$ along the lines of (C.11) repeatedly and using $\hat{V}_1\big(S_m = [3\ 2]\big) - \hat{V}_1\big(S_m = [2\ 3]\big) = r_3 - r_2 \geq 0$, by induction, we can show that $\hat{V}_{m-2}\big(S_{m-1} = [3\ 2]\big) - \hat{V}_{m-2}\big(S_{m-1} = [2\ 3]\big) \geq 0$. Thus $\hat{V}_{m-1}\big(S_m = [3\ 2]\big) - \hat{V}_{m-1}\big(S_m = [2\ 3]\big) \geq 0$. Applying this inequality in (C.10), we see that the optimality of the greedy policy (in the specified class of policies) can be established if we show that the following condition (condition (S)) holds:

$$\pi_{m,\hat{a}_m}(3)\pi_{m,\tilde{a}_m}(2) \geq \pi_{m,\hat{a}_m}(2)\pi_{m,\tilde{a}_m}(3). \tag{C.12}$$

It appears that the preceding condition is too generic to hold true. However, by constraining the belief vectors to the set of values that will be encountered in the ARQ based scheduling problem, we will now show that, (C.12) indeed holds true.

We first introduce the following result: From Lemma 9, $\mathbf{p}_2 P^k [001]^T$ monotonically decreases to $\pi_{ss}[001]^T = \pi_{ss}(3)$ as k increases. Since $\mathbf{p}_2 P^k [010] = p_{22} = \pi_{ss}(2)$, the expected reward from an user given the channel of the user was in state 2 $k+1$ intervals earlier, given by, $\mathbf{p}_2 P^k \alpha = \alpha_(2)\pi_{ss}(2) + \mathbf{p}_2 P^k [001]^T$ monotonically decreases to $\pi_{ss}\alpha$.

We proceed with studying the sufficient condition under various belief vectors encountered in the ARQ based scheduling problem. Assume the scheduling process has begun in a control interval earlier than m and is performed uninterrupted till the horizon, i.e, control interval 1 - assumption (A)[18]. The belief vector of the greedy

[18]Note that there is no loss of generality in this assumption for the following reason: The problem setup and the optimality analysis of any policy implicitly assumes uninterrupted scheduling until the horizon. This is to be in tune with the interval to interval evolution of the underlying Markov chains. Thus when the uninterrupted scheduling process begins at a control interval M, for all $m < M$ condition (A) is satisfied automatically. In the control interval M, however, part of the condition, i.e, *scheduling process began earlier*, does not hold. But at the origin, i.e., the control

choice $\hat{a}_m$ and the other user $\tilde{a}_m$, for the type I system under consideration, falls under one of the following cases.

- 1. User $\hat{a}_m$ was scheduled in the previous control interval, $m+1$, and had given a feedback F_3. The belief vector $\pi_{m,\hat{a}_m} = \mathbf{p}_3$. The other user was either scheduled in $k+1$ control intervals earlier (with $k \in 1, 2, \ldots$) with any of the three possible feedback or was never scheduled in the past. Thus the belief vector of $\tilde{a}_m$ is of the form $\mathbf{p}_i P^k$ with $i \in 1, 2, 3$ and $k \in 1, 2, \ldots$. Note that if $\tilde{a}_m$ was never scheduled in the past, then $\pi_{m,\tilde{a}_m} = \pi_{ss}$ which still falls in the preceding form.

- 2. User $\tilde{a}_m$ was scheduled in the previous control interval and had given a feedback F_1. User $\hat{a}_m$ was either scheduled $k+1$ control intervals earlier (with $k \in 1, 2, \ldots$) with any of the three possible feedbacks or was never scheduled in the past. The belief vectors are given by $\pi_{m,\tilde{a}_m} = \mathbf{p}_1$ and $\pi_{m,\hat{a}_m} = \mathbf{p}_i P^k$ with $i \in 1, 2, 3$ and $k \in 1, 2, \ldots$.

- 3. User $\hat{a}_m$ was scheduled in the previous control interval and had given a feedback F_2. User $\tilde{a}_m$ was scheduled $k+1$ control intervals earlier (with $k \in 1, 2, \ldots$) with feedback F_1 or was never scheduled in the past. The belief vectors are given by $\pi_{m,\hat{a}_m} = \mathbf{p}_2$ $\pi_{m,\tilde{a}_m} = \mathbf{p}_1 P^k$ with $k \in 1, 2, \ldots$.

- 4. User $\hat{a}_m$ was scheduled in the previous control interval and had given a feedback F_2. User $\tilde{a}_m$ was scheduled $k+1$ control intervals earlier (with $k \in 1, 2, \ldots$) with feedback F_2. The belief vectors are given by $\pi_{m,\hat{a}_m} = \mathbf{p}_2$ $\pi_{m,\tilde{a}_m} = \mathbf{p}_2 P^k$ with $k \in 1, 2, \ldots$.

interval M, the belief vectors of all the users take the steady state value, π_{ss}. Thus, by all symmetry, the question of what scheduling decision to make and hence the question of the optimality of the greedy policy at M becomes irrelevant.

- 5. User $\hat{a}_m$ was scheduled in the previous control interval and had given a feedback F_2. User $\tilde{a}_m$ was scheduled $L+1$ or more control intervals earlier with feedback F_3. L is the number of coherence intervals such that, reward expected from an user that was observed to be in state 2 in the previous control interval is higher than the reward expected from an user that was observed in state 3 $k+1$ control intervals earlier iff $k \geq L$. Mathematically, L is such that,

$$\mathbf{p}_2\alpha \geq \mathbf{p}_3 P^k \alpha \; if \; k \geq L \mathbf{p}_2\alpha < \mathbf{p}_3 P^k \alpha \; if \; k < L \tag{C.13}$$

  Note that such an L exists since $\mathbf{p}_2\alpha \leq \mathbf{p}_3\alpha$ and both $\mathbf{p}_2 P^k\alpha$ and $\mathbf{p}_3 P^k\alpha$ monotonically decreases (with k) to $\pi_{ss}\alpha \leq \mathbf{p}_2\alpha$. The belief vectors are hence given as $\pi_{m,\hat{a}_m} = \mathbf{p}_2$, $\pi_{m,\tilde{a}_m} = \mathbf{p}_3 P^k$ with $k \geq L$.

- 6. User $\tilde{a}_m$ was scheduled in the previous control interval and had given a feedback F_2. User $\hat{a}_m$ was scheduled $k+1$ control intervals earlier with feedback F_3 with $k < L$. The belief vectors are as follows: $\pi_{m,\hat{a}_m} = \mathbf{p}_3 P^k$ with $k < L$ and $\pi_{m,\tilde{a}_m} = \mathbf{p}_2$.

The above list is exhaustive. In fact, cases 5 and 6 will never appear since we are considering the class of schedulers that never drop an user when it sends an F_3. However, we will show that even for these cases the sufficient condition is satisfied. In all the above 6 cases, $R_m(\hat{a}_m) \geq R_m(\tilde{a}_m)$ as required by the definition of $\hat{a}_m$. We now focus on the sufficient condition (S) for each of the above cases.

- 1. Sufficient condition (S) is given as follows:

$$\begin{aligned} \pi_{m,\hat{a}_m}(3)\pi_{m,\tilde{a}_m}(2) &\geq \pi_{m,\hat{a}_m}(2)\pi_{m,\tilde{a}_m}(3) \\ i.e., \; p_{33}\mathbf{p}_i P^k[010]^T &\geq p_{32}\mathbf{p}_i P^k[001]^T, \forall i \in 1,2,3, \; k \in 1,2,\ldots \end{aligned} \tag{C.14}$$

Since $p_{12} = p_{22} = p_{32}$, we have

$$\mathbf{p}_i P^k [010]^T = p_{12} = p_{22} = p_{32} \forall i \in 1, 2, 3,\ k \in 1, 2, \ldots \tag{C.15}$$

Also, $\mathbf{p}_i P^k [001]^T = \mathbf{p}_i P^{k-1} P [001]^T = \mathbf{p}_i P^{k-1} [p_{13}\ p_{23}\ p_{33}]^T \leq p_{33}$, since $p_{33} \geq p_{23} \geq p_{13}$ by the property of the P matrix. Thus (S) holds for case 1.

- 2. (S) is as follows: $\mathbf{p}_i P^k [001]^T p_{12} \geq \mathbf{p}_i P^k [010]^T p_{13}, \forall i \in 1, 2, 3,\ k \in 1, 2, \ldots$.

  From the symmetry property (C.15), $p_{12} = \mathbf{p}_i P^k [010]^T$. Also since $p_{13} \leq p_{23} \leq p_{33}$ we can show $\mathbf{p}_i P^k [001]^T \geq p_{13}$. Thus (S) is satisfied for case 2.

- 3. (S): $p_{23} \mathbf{p}_1 P^k [010]^T \geq p_{22} \mathbf{p}_1 P^k [001]^T$. From Lemma 10, $p_1 P^k [001]^T$ monotonically increases to $\pi_s s(3)$ as k increases as $0, 1, 2, \ldots$. Since $p_{23} \geq \pi_{ss}(3)$ (using Lemma 9), we have $p_{23} \geq \mathbf{p}_1 P^k [001]^T$. Also, $\mathbf{p}_1 P^k [010]^T = p_{22}$ from the symmetry property in (C.15). Thus (S) holds for case 3.

- 4. (S): $p_{23} \mathbf{p}_2 P^k [010]^T \geq p_{22} \mathbf{p}_2 P^k [001]^T$. From Lemma 9, $\mathbf{p}_2 P^k [001]^T$ monotonically decreases from p_{23} to $\pi_{ss}(3)$ as k increases as $0, 1, 2, \ldots$. Thus $p_{23} \geq \mathbf{p}_2 P^k [001]^T$. This inequality along with the symmetry property (C.15) establishes (S) for case 4.

- 5. (S): $p_{23} \mathbf{p}_3 P^k [010]^T \geq p_{22} \mathbf{p}_3 P^k [001]^T$ with $k \geq L$. Note that for all $k \geq L$,

$$\begin{aligned} \mathbf{p}_2 \alpha &\geq \mathbf{p}_3 P^k \alpha \\ \Rightarrow \alpha_2 p_{22} + p_{23} &\geq \alpha_2 \mathbf{p}_3 P^k [010]^T + \mathbf{p}_3 P^k [001]^T \\ \Rightarrow p_{23} &\geq \mathbf{p}_3 P^k [001]^T \end{aligned} \tag{C.16}$$

where we have used the symmetry property $p_{22} = \mathbf{p}_3 P^k [010]^T$ in obtaining the last inequality. (S) is established by using the symmetry property along with the preceding inequality.

- 6. (S): $\mathbf{p}_3 P^k [001]^T p_{22} \geq \mathbf{p}_3 P^k [010]^T p_{23}$ with $k < L$. For $k < L$, $\mathbf{p}_2 \alpha < \mathbf{p}_3 P^k \alpha$. Expanding both the sides along the lines of case 5 and using the symmetry property of (C.15), (S) can be established for case 6.

Thus the sufficient condition for the constrained search space optimality of the greedy policy is satisfied.

APPENDIX D

PROOFS FOR CHAPTER 5

D.1 Proof of Lemma 12

Let condition (D.2) be true. Let $m > 1$ be fixed. Since, by assumption, $\pi_{t_m}(n) \geq \pi_{t_m}(n+1)\ \forall n \in \{1 \dots F_2 - 1\}$, we have from (5.13),

$$V_{t_m}(\pi_{t_m}, \{a_{t_m} = n, \{\widehat{\mathfrak{A}}_{t_k}\}_{k \leq m-1}\}) \quad \geq \quad V_{t_m}(\pi_{t_m}, \{a_{t_m} = n+1, \{\widehat{\mathfrak{A}}_{t_k}\}_{k \leq m-1}\}).$$

Therefore,

$$\begin{aligned} & V_{t_m}(\pi_{t_m}, \{a_{t_m} = \arg\max_i \pi_{t_m}(i) = 1, \{\widehat{\mathfrak{A}}_{t_k}\}_{k \leq m-1}\}) \\ \geq \quad & V_{t_m}(\pi_{t_m}, \{a_{t_m} \in \{2 \dots F_2\}, \{\widehat{\mathfrak{A}}_{t_k}\}_{k \leq m-1}\}). \end{aligned}$$

We now have the following statement:

If $\forall \pi_{t_{m-1}} \in [0,1]^{F_2}$,

$$\{\widehat{\mathfrak{A}}_{t_k}\}_{k \leq m-1} = \arg \max_{\{\mathfrak{A}_{t_k}\}_{k \leq m-1}} V_{t_{m-1}}(\pi_{t_{m-1}}, \{\mathfrak{A}_{t_k}\}_{k \leq m-1}),$$

then $\forall \pi_{t_m} \in [0,1]^{F_2}$,

$$\{\widehat{\mathfrak{A}}_{t_k}\}_{k \leq m} = \arg \max_{\{\mathfrak{A}_{t_k}\}_{k \leq m}} V_{t_m}(\pi_{t_m}, \{\mathfrak{A}_{t_k}\}_{k \leq m}). \tag{D.1}$$

Since $\widehat{\mathfrak{A}}_{t_1} = \arg\max_{\mathfrak{A}_{t_1}} V_{t_1}(\pi_{t_1}, \mathfrak{A}_{t_1}), \forall \pi_{t_1} \in [0,1]^{F_2}$, using (D.1), by induction, we have

$$\{\widehat{\mathfrak{A}}_{t_k}\}_{k \le m} \;=\; \arg \max_{\{\mathfrak{A}_{t_k}\}_{k \le m}} V_{t_m}(\pi_{t_m}, \{\mathfrak{A}_{t_k}\}_{k \le m}) \quad \forall \mathbf{n} \ge m \ge 1, \pi_{t_m} \in [0,1]^{F_2}.$$

The lemma thus follows.

D.2 Proof of Proposition 16

We begin by establishing that the sufficient condition for the optimality of the greedy policy on the sporadic time axis $\mathbf{t_n}$ in fact holds. Consider a realization of the channel states of the F_2 users on the time axis $\mathbf{t}_{m-1}$, $m \le n$. Denote it by $\{\mathcal{R}, i, j\}$, where i, j indicate the channel state of users $n+1$ and F_2, respectively, at time t_{m-1} with $\mathcal{R}$ indicating the rest of the channel state realization. We can rewrite the second quantity of the sufficient condition as follows.

$$\hat{V}_{t_{m-1}}\big([1\ Y\ 0\ X], [1 \dots F_2]\big) \;=\; \hat{V}_{t_{m-1}}\big([Y\ 0\ X\ 1], [F_2, 1 \dots F_2 - 1]\big)$$

Define $V_a(\{\mathcal{R}, i, j\})$ as the reward accrued from time t_{m-1} on the sporadic axis when the channel states have a realization $\{\mathcal{R}, i, j\}$ and the greedy policy is implemented in the order $[1 \dots F_2]$ from slot t_{m-1}. Let $V_b(\{\mathcal{R}, i, j\})$ be similarly defined with the order given by $[F_2, 1 \dots F_2 - 1]$. Let $P(\{i,j\}|\{k,l\}) = P(S_{t_{m-1}}(n+1) = i, S_{t_{m-1}}(F_2) =$

$j|S_{t_m}(n+1) = k, S_{t_m}(F_2) = l)$. The sufficient condition can now be rewritten as below.

$$
\begin{aligned}
&\hat{V}_{t_{m-1}}\big([Y\ 1\ X\ 0], [1 \dots F_2]\big) - \hat{V}_{t_{m-1}}\big([1\ Y\ 0\ X], [1 \dots F_2]\big) \\
&= \sum_{\mathcal{R}} P(\mathcal{R}|S_{t_m}(1) \dots S_{t_m}(n) = Y, S_{t_m}(n+2) \dots S_{t_m}(F_2) = X) \times \\
&\quad \Big(P(\{1,0\}|\{1,0\})V_a(\{\mathcal{R},1,0\}) - P(\{0,1\}|\{0,1\})V_b(\{\mathcal{R},0,1\}) \\
&\quad + P(\{0,1\}|\{1,0\})V_a(\{\mathcal{R},1,0\}) - P(\{1,0\}|\{0,1\})V_b(\{\mathcal{R},0,1\}) \\
&\quad + P(\{0,0\}|\{1,0\})V_a(\{\mathcal{R},1,0\}) - P(\{0,0\}|\{0,1\})V_b(\{\mathcal{R},0,1\}) \\
&\quad + P(\{1,1\}|\{1,0\})V_a(\{\mathcal{R},1,0\}) - P(\{1,1\}|\{0,1\})V_b(\{\mathcal{R},0,1\})\Big) \\
&= p(1-r)(V_a(\{\mathcal{R},1,0\}) - V_b(\{\mathcal{R},0,1\})) \\
&\quad + (1-p)(r)(V_a(\{\mathcal{R},0,1\}) - V_b(\{\mathcal{R},1,0\})) \\
&\quad + (1-p)(1-r)(V_a(\{\mathcal{R},0,0\}) - V_b(\{\mathcal{R},0,0\})) \\
&\quad + pr(V_a(\{\mathcal{R},1,1\}) - V_b(\{\mathcal{R},1,1\})). \qquad \text{(D.2)}
\end{aligned}
$$

It has been shown in [45] that when greedy policy is implemented in orders $[1 \dots F_2]$ and $[F_2, 1 \dots F_2 - 1]$, the difference in reward accrued, for any fixed realization, is upper bounded by 1. The sample path argument used in the proof works for the non-sporadic axis as well, as long as $\beta_k \geq \beta_l$ for $k \geq l$, which is indeed true. Thus $V_a(\{\mathcal{R}, i, j\}) - V_b(\{\mathcal{R}, i, j\}) \leq 1$ for any $\{\mathcal{R}, i, j\}$. Notice that since the realization is fixed and since $V_a(\{\mathcal{R}, i, j\})$ schedules user 1 first, the value of j does not affect V_a. Thus $V_a(\{\mathcal{R}, i, 1\}) = V_a(\{\mathcal{R}, i, 0\})$. Similarly $V_b(\{\mathcal{R}, 1, j\}) = V_b(\{\mathcal{R}, 0, j\})$. Using these observations in (D.2), we show the sufficient condition holds. Similarly, $\hat{\mathfrak{A}}^n$ is optimal on the sporadic axis $\{h, h-1, \dots 1\} - \mathbf{t_n}$. The proposition is thus established from Lemmas 11 and 12.

BIBLIOGRAPHY

[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] 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.

[3] 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.

[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] 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.

[7] 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.

[8] H.S. Wang and P.-C. Chang, On verifying the first-order Markovian assumption for a Rayleigh fading channel model, *IEEE Trans. Vehicular Technology,* vol. 45, pp. 353-357, May 1996.

[9] Q. Zhang and S. A. Kassam, "Finite-state Markov model for Rayleigh fading channels," *IEEE Transactions on Communications,* vol. 47, no. 11, Nov. 1999.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

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

[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] E. J. Sondik, "The optimal control of partially observable Markov processes," *PhD Thesis*, Stanford University, 1971.

[18] R. D. Smallwood and E. J. Sondik, "The optimal control of partially observable Markov processes over a finite horizon," *Operations Research,* Sep. 1973.

[19] S. Christian Albright, "Structural results for partially observable Markov decision processes," *Operations Research,* vol. 27, no. 5, pp. 1041-1053, Sep.-Oct. 1979.

[20] C. C. White and W. Scherer, "Solution procedures for partially observed Markov decision processes," *Operations Research,* pp. 791797, 1985.

[21] G. E. Monahan, "A survey of partially observable Markov decision processes: Theory, Models, and Algorithms," *Management Science,* vol. 28, no. 1, pp. 116, Jan. 1982.

[22] N. Meuleau, K. Kim, L. P. Kaelbling, A. R. Cassandra, "Solving POMDPs by searching the space of finite policies," *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence,* pp. 417-426, 1999.

[23] P. Whittle, "Restless bandits: Activity allocation in a changing world," *Journal of Applied Probability,* 25A:287 - 298, 1988.

[24] Andrew Tanenbaum, *Computer Networks,* Prentice Hall, ed. 4, 2003.

[25] I. Chlamtac and S. Kutten, "On broadcasting in radio networks - problem analysis and protocol design," *IEEE Transactions on Communications,* COM-33 (12):1240-1246, Dec. 1985.

[26] A Durresi, V. K. Paruchuri, S. S. Iyengar, R. Kannan, "Optimized broadcast protocol for sensor networks," *IEEE Transactions on Computers,* vol. 54 , issue 8, Aug. 2005.

[27] M. Agarwal, J. H. Cho, L. Gao and J. Wu, "Energy efficient broadcast in wireless ad hoc networks with Hitch-hiking," *Proc. IEEE INFOCOM,* 2004.

[28] S. Singh, C. Raghavendra and J. Stepanek, "Power-aware broadcasting in mobile ad hoc networks," *Proc. IEEE INFOCOM*, Sept. 1999.

[29] A. J. Goldsmith and S. B. Wicker, "Design challenges for energy constrained ad hoc wireless networks," *IEEE Wireless Communications*, vol. 9, no. 4, pp. 827, Aug. 2002.

[30] A. Ephremides, "Energy concerns in wireless networks," *IEEE Wireless Communications,* vol. 9, no. 4, pp. 4859, Aug. 2002.

[31] R. Min, M. Bhardwaj, S. Cho, N. Ickes, E. Shih, A. Sinha, A. Wang, and A. Chandrakasan, "Energy-centric enabling technologies for wireless sensor networks," *IEEE Wireless Communications,* vol. 9, no. 4, pp. 2839, Aug. 2002.

[32] V. Srivastava and M. Motani, "Cross-layer design: a survey and the road ahead," *IEEE Communications Magazine,* vol. 43, no. 12, pp. 112119, Dec. 2005.

[33] X Lin, N. B. Shroff, R. Srikant, "A tutorial on cross-layer optimization in wireless networks," *IEEE Journal on Selected Areas in Communications,* vol. 24, no. 8, pp. 1452-1463, Aug. 2006.

[34] G. Miao, N. Himayat, Y. Li, A. Swami, "Cross-layer optimization for energy-efficient wireless communications: a survey," *Wireless Communications and Mobile Computing,* vol. 9, no. 4, pp. 529-542, Apr. 2009.

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

[36] D. P. Bertsekas, *Dynamic programming and optimal control,* Athena Scientific, vol. 1, ed. 3, 2007.

[37] T. M. Cover, J. A. Thomas, *Elements of Information Theory,* Wiley-Interscience; ed.1, 1991.

[38] C. H. Papadimitriou, J. N. Tsitsiklis, "The complexity of Markov decision processes," *Mathematics of Operations Research,* 12, 441450, 1987.

[39] 0. Madani, S. Hanks, A. Condon, "On the undecidability of probabilistic planning and infinite-horizon partially observable Markov decision processes," *Proceedings of the Sixteenth National Conference on Artificial Intelligence,* 1999.

[40] George Lawton, "What lies ahead for cellular technology?," *Computer: IEEE Computer Society,* vol. 38, no. 6, pp. 14-17, June 2005.

[41] S. Lin, D. Costello, and M. Miller, "Automatic-repeat-request error control schemes," *IEEE Communications Magazine,* vol. 22, pp. 517, Dec. 1984.

[42] 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.

[43] 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.

[44] 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.

[45] S. H. Ahmad, M. Liu, T. Javidi, Q. Zhao and B. Krishnamachari, "Optimality of myopic sensing in multi-channel opportunistic access," *IEEE Transactions on Information Theory*, vol. 55, No. 9, pp. 4040-4050, September, 2009.

[46] V. H. Mac Donald, "The Cellular Concept," *The Bell System Technical Journal*, Vol. 58, No. 1, pp. 15-41, Jan 1979.

[47] F. Borgonovo, M. Zorzi, L. Fratta, V. Trecordi, and G. Bianchi, "Capture-division packet access for wireless personal communications," *IEEE J. Select. Areas Commun.*, vol. 14, pp. 609-622, May 1996.

[48] J. Li and N. B. Shroff and E. K. P. Chong, "A Reduced-power channel reuse scheme for wireless packet cellular networks," *IEEE/ACM Trans. on Networking*, vol. 7, no. 6, pp. 818-832, Dec 1999.

[49] X. Wu, A. Das, J. Li and R. Laroia, "Fractional power reuse in cellular networks," *Proc. Allerton Conf. on Communication, Control, and Computing,* (Monticello, IL), Oct. 2006.

[50] S. Jing, D. N. C. Tse, J. Hou, J. B. Soriaga, J. E. Smee, R. Padovani, "Multi-cell downlink capacity with coordinated processing," *Proc. Information Theory and Applications Workshop (ITA),*" San Diego, CA, Jan 2007.

[51] J. B. Andersen, T. S. Rappaport, and S. Yoshida, "Propagation measurements and models for wireless communications channels," *IEEE Communication Mag.,* vol. 33, no. 1, Jan. 1995

[52] W. Rudin, *Principles of Mathematical Analysis,* McGraw-Hill Education, ed. 3, 1977.

[53] J. C. Gittins, "Multi-armed bandit allocation indices," *John Wiley and Sons,* New York, NY, 1989.

[54] J. C. Gittins and D. M. Jones, "A dynamic allocation index for sequential design of experiments," *Progress in Statistics,* Euro. Meet. Statis., 1:241-266, 1972.

[55] R. R. Weber and G. Weiss, "On an index policy for restless bandits," *Journal of Applied Probability,* 27, 637 - 648, Sep. 1990.

[56] D. Bertsimas and J. Nino-Mora, "Restless bandits, linear programming relaxations, and a primal-dual index heuristic," *Operations Research,* 48:80-90, 2000.

[57] K. D. Glazebrook, J. Nino Mora, and P. S. Ansell, "Index policies for a class of discounted restless bandits," *Advances in Applied Probability,* 34(4):754-774, 2002.

[58] J. Nino-Mora, "Restless bandits, partial conservation laws, and indexability," *Advances in Applied Probability,* 33:76-98, 2001.

[59] J. Nino-Mora, "Dynamic allocation indices for restless projects and queuing admission control: a polyhedral approach," *Mathematical Programming,* Series A, 93:361-413, 2002.

[60] K. Liu, Q. Zhao, "Indexability of restless bandit problems and optimality of Whittle's index for dynamic multichannel access," to appear in IEEE Transactions on Information Theory, 2010.

[61] K. D. Glazebrook, H. M. Mitchell, "An index policy for a stochastic scheduling model with improving/deteriorating jobs," *Naval Research Logistics,* 2002.