Exploiting Channel Memory for Multi-User Wireless Scheduling without Channel Measurement: Capacity Regions and Algorithms

Chih-Ping Li, Michael J. Neely

Exploiting Channel Memory for Multi-User Wireless Scheduling without Channel Measurement: Capacity Regions and Algorithms

Chih-ping Li, *Student Member, IEEE* and Michael J. Neely, *Senior Member, IEEE*

***Abstract*—We study the fundamental network capacity of a multi-user wireless downlink under two assumptions: (1) Channels are not explicitly measured and thus instantaneous states are unknown, (2) Channels are modeled as ON/OFF Markov chains. This is an important network model to explore because channel probing may be costly or infeasible in some contexts. In this case, we can use channel memory with ACK/NACK feedback from previous transmissions to improve network throughput. Computing in closed form the capacity region of this network is difficult because it involves solving a high dimension partially observed Markov decision problem. Instead, in this paper we construct an inner and outer bound on the capacity region, showing that the bound is tight when the number of users is large and the traffic is symmetric. For the case of heterogeneous traffic and any number of users, we propose a simple queue-dependent policy that can stabilize the network with any data rates strictly within the inner capacity bound. The stability analysis uses a novel frame-based Lyapunov drift argument. The outer-bound analysis uses stochastic coupling and state aggregation to bound the performance of a restless bandit problem using a related multi-armed bandit system. Our results are useful in cognitive radio networks, opportunistic scheduling with delayed/uncertain channel state information, and restless bandit problems.**

***Index Terms*—stochastic network optimization, Markovian channels, delayed channel state information (CSI), partially observable Markov decision process (POMDP), cognitive radio, restless bandit, opportunistic spectrum access, queueing theory, Lyapunov analysis.**

I. Introduction

Due to the increasing demand of cellular network services, in the past fifteen years efficient communication over a single-hop wireless downlink has been extensively studied. In this paper we study the fundamental network capacity of a time-slotted wireless downlink under the following assumptions: (1) Channels are never explicitly probed, and thus their instantaneous states are never known, (2) Channels are modeled as two-state ON/OFF Markov chains. This network model is important because, due to the energy and timing overhead, learning instantaneous channel states by probing may be costly or infeasible. Even if this is feasible (when channel coherence time is relatively large), the time consumed by channel probing cannot be re-used for data transmission, and transmitting data without probing may achieve higher throughput [1].[1] In addition, since wireless channels can be adequately modeled as Markov chains [2], [3], we shall take advantage of channel memory to improve network throughput.

Specifically, we consider a time-slotted wireless downlink where a base station serves N users through N (possibly different) *positively correlated* Markov ON/OFF channels. Channels are never probed so that their instantaneous states are unknown. In every slot, the base station selects at most one user to which it transmits a packet. We assume every packet transmission takes exactly one slot. Whether the transmission succeeds depends on the unknown channel state. At the end of a slot, an ACK/NACK is fed back from the served user to the base station. Since channels are ON/OFF, this feedback reveals the channel state of the served user in the last slot and provides partial information of future states. Our goal is to characterize all achievable throughput vectors in this network, and to design simple throughput-achieving algorithms.

We define the *network capacity region* Λ as the closure of the set of all achievable throughput vectors. We can compute Λ by locating its boundary points. Every boundary point can be computed by formulating a partially observable Markov decision process (POMDP) [4], with information states defined as, conditioning on the channel observation history, the probabilities that channels are ON. This approach, however, is computationally prohibitive because the information state space is countably infinite (which we will show later) and grows exponentially fast with N.

The first contribution of this paper is that we construct an outer and an inner bound on Λ. The outer bound comes from analyzing a fictitious channel model in which every scheduling policy yields higher throughput than it does in the real network. The inner bound is the achievable rate region of a special class of *randomized round robin policies* (introduced in Section IV-A). These policies are simple and take advantage of channel memory. In the case of symmetric channels (that

Chih-ping Li (web: http://www-scf.usc.edu/∼chihpinl) and Michael J. Neely (web: http://www-rcf.usc.edu/∼mjneely) are with the Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089, USA.

This material is supported in part by one or more of the following: the DARPA IT-MANET program grant W911NF-07-0028, the NSF Career grant CCF-0747525, and continuing through participation in the Network Science Collaborative Technology Alliance sponsored by the U.S. Army Research Laboratory.

[1] One quick example is to consider a time-slotted channel with state space $\{\mathsf{B}, \mathsf{G}\}$. Suppose channel states are i.i.d. over slots with stationary probabilities $\Pr[\mathsf{B}] = 0.2$ and $\Pr[\mathsf{G}] = 0.8$. At state B and G, at most 1 and 2 packets can be successfully delivered in a slot, respectively. Packet transmissions beyond the capacity will all fail and need retransmissions. Channel probing can be done on each slot, which consumes 0.2 fraction of a slot. Then the policy that always probes the channel yields throughput $0.8(2 \cdot 0.8 + 1 \cdot 0.2) = 1.44$, while the policy that never probes the channel and always sends packets at rate 2 packets/slot yields throughput $2 \cdot 0.8 = 1.6 > 1.44$.

is, channels are i.i.d.) and when the network serves a large number of users, we show that as data rates are more *balanced*, or in a geometric sense as the direction of the data rate vector in the Euclidean space is closer to the 45-degree angle, the inner bound converges geometrically fast to the outer bound, and the bounds are tight. This analysis uses results in [5], [6] that derive an outer bound on the maximum sum throughput for a symmetric system.

The inner capacity bound is indeed useful. First, the structure of the bound itself shows how channel memory improves throughput. Second, we show analytically that a large class of intuitively good heuristic policies achieve throughput that is at least as good as this bound, and hence the bound acts as a (non-trivial) performance guarantee. Finally, supporting throughput outside this bound may inevitably involve solving a much more complicated POMDP. Thus, for simplicity and practicality, we may regard the inner bound as an *operational* network capacity region.

In this paper we also derive a simple queue-dependent dynamic round robin policy that stabilizes the network whenever the arrival rate vector is interior to our inner bound. This policy has polynomial time complexity and is derived by a novel *variable-length frame-based Lyapunov analysis*, first used in [7] in a different context. This analysis is important because the inner bound is based on a mixture of many different types of round robin policies, and an offline computation of the proper time average mixtures needed to achieve a given point in this complex inner bound would require solving $\Theta(2^N)$ unknowns in a linear system, which is impractical when N is large. The Lyapunov analysis overcomes this complexity difficulty with online queue-dependent decisions.

The results of this paper apply to the emerging area of opportunistic spectrum access in cognitive radio networks (see [8] and references therein), where the channel occupancy of a primary user acts as a Markov ON/OFF channel to the secondary users. Specifically, our results apply to the important case where every secondary users has a designated channel and they cooperate via a centralized controller. This paper is also a study on efficient scheduling over wireless networks with delayed/uncertain channel state information (CSI) (see [9]–[11] and references therein). The work on delayed CSI that is most closely related to ours is [10], [11], in which the capacity region and throughput-optimal policies of different wireless networks are studied, assuming that channel states are *persistently* probed but fed back *with delay*. Our paper is significantly different. Here channels are never probed, and new (delayed) CSI of a channel is only acquired when the channel is served. Implicitly, acquiring the delayed CSI of any channel is part of the control decisions in this paper. The problem formulation of this paper also applies to an important scenario in partial channel probing (see [1], [12] and references there in) where: (1) At most one channel is probed in every slot, (2) Data can be transmitted over probed channels but not on unknown channels. Specifically, we study how channel memory can improve the network throughput.

This paper is organized as follows. The network model is given in Section II, inner and outer bounds are constructed in Sections III and IV, and compared in Section V in the case of symmetric channels. Section VI gives the queue-dependent policy to achieve the inner bound.

II. Network Model

Consider a base station transmitting data to N users through N Markov ON/OFF channels. Suppose time is slotted with normalized slots t in $\{0, 1, 2, \ldots\}$. Each channel is modeled as a two-state ON/OFF Markov chain (see Fig. 1). The state

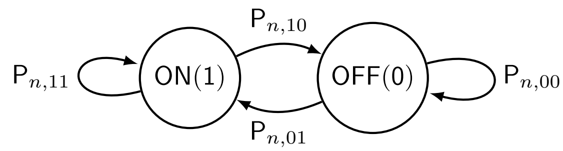

Fig. 1. A two-state Markov ON/OFF chain for channel $n \in \{1, 2, \ldots, N\}$.

evolution of channel $n \in \{1, 2, \ldots, N\}$ follows the transition probability matrix

$$\mathbf{P}_n = \begin{bmatrix} \mathsf{P}_{n,00} & \mathsf{P}_{n,01} \\ \mathsf{P}_{n,10} & \mathsf{P}_{n,11} \end{bmatrix},$$

where state ON is represented by 1 and OFF by 0, and $\mathsf{P}_{n,ij}$ denotes the transition probability from state i to j. We assume $\mathsf{P}_{n,11} < 1$ for all n so that no channel is constantly ON. Incorporating constantly ON channels like wired links is easy and thus omitted in this paper. We suppose channel states are fixed in every slot and may only change at slot boundaries. We assume all channels are positively correlated, which, in terms of transition probabilities, is equivalent to assuming $\mathsf{P}_{n,11} > \mathsf{P}_{n,01}$ or $\mathsf{P}_{n,01} + \mathsf{P}_{n,10} < 1$ for all n.[2] We suppose the base station keeps N queues of infinite capacity to store exogenous packet arrivals destined for the N users. At the beginning of every slot, the base station attempts to transmit a packet (if there is any) to a selected user. We suppose the base station has no channel probing capability and must select users oblivious of the current channel states. If a user is selected and its current channel state is ON, one packet is successfully delivered to that user. Otherwise, the transmission fails and zero packets are served. At the end of a slot in which the base station serves a user, an ACK/NACK message is fed back from the selected user to the base station through an independent error-free control channel, according to whether the transmission succeeds. Failing to receive an ACK is regarded as a NACK. Since channel states are either ON or OFF, such feedback reveals the channel state of the selected user in the last slot.

Conditioning on all past channel observations, define the N-dimensional *information state vector* $\boldsymbol{\omega}(t) = (\omega_n(t) : 1 \leq n \leq N)$ where $\omega_n(t)$ is the conditional probability that channel n is ON in slot t. We assume initially $\omega_n(0) = \pi_{n,\mathsf{ON}}$ for all n, where $\pi_{n,\mathsf{ON}}$ denotes the stationary probability that channel

[2]Assumption $\mathsf{P}_{n,11} > \mathsf{P}_{n,01}$ yields that the state $s_n(t)$ of channel n has auto-covariance $\mathbb{E}\left[(s_n(t) - \mathbb{E}s_n(t))(s_n(t+1) - \mathbb{E}s_n(t+1))\right] > 0$. In addition, we note that the case $\mathsf{P}_{n,11} = \mathsf{P}_{n,01}$ corresponds to a channel having i.i.d. states over slots. Although we can naturally incorporate i.i.d. channels into our model and all our results still hold, we exclude them in this paper because we shall show how throughput can be improved by channel memory, which i.i.d. channels do not have. The degenerate case where all channels are i.i.d. over slots is fully solved in [1].

n is ON. As discussed in [4, Chapter 5.4], vector $\boldsymbol{\omega}(t)$ is a *sufficient statistic*. That is, instead of tracking the whole system history, the base station can act optimally only based on $\boldsymbol{\omega}(t)$. The base station shall keep track of the $\{\boldsymbol{\omega}(t)\}$ process.

We assume transition probability matrices $\mathbf{P}_n$ for all n are known to the base station. We denote by $s_n(t) \in \{\mathsf{OFF}, \mathsf{ON}\}$ the state of channel n in slot t. Let $n(t) \in \{1, 2, \dots, N\}$ denote the user served in slot t. Based on the ACK/NACK feedback, vector $\boldsymbol{\omega}(t)$ is updated as follows. For $1 \leq n \leq N$,

$$\omega_n(t+1) = \begin{cases} \mathsf{P}_{n,01}, & \text{if } n = n(t),\ s_n(t) = \mathsf{OFF} \\ \mathsf{P}_{n,11}, & \text{if } n = n(t),\ s_n(t) = \mathsf{ON} \\ \omega_n(t)\mathsf{P}_{n,11} + (1 - \omega_n(t))\mathsf{P}_{n,01}, & \text{if } n \neq n(t). \end{cases} \tag{1}$$

If in the most recent use of channel n, we observed (through feedback) its state was $i \in \{0, 1\}$ in slot $(t-k)$ for some $k \leq t$, then $\omega_n(t)$ is equal to the k-step transition probability $\mathsf{P}_{n,i1}^{(k)}$. In general, for any fixed n, probabilities $\omega_n(t)$ take values in the countably infinite set $\mathcal{W}_n = \{\mathsf{P}_{n,01}^{(k)}, \mathsf{P}_{n,11}^{(k)} : k \in \mathbb{N}\} \cup \{\pi_{n,\mathsf{ON}}\}$. By eigenvalue decomposition on $\mathbf{P}_n$ [13, Chapter 4], we can show the k-step transition probability matrix $\mathbf{P}_n^{(k)}$ is

$$\begin{aligned} \mathbf{P}_n^{(k)} &\triangleq \begin{bmatrix} \mathsf{P}_{n,00}^{(k)} & \mathsf{P}_{n,01}^{(k)} \\ \mathsf{P}_{n,10}^{(k)} & \mathsf{P}_{n,11}^{(k)} \end{bmatrix} = (\mathbf{P}_n)^k \\ &= \frac{1}{x_n} \begin{bmatrix} \mathsf{P}_{n,10} + \mathsf{P}_{n,01}(1 - x_n)^k & \mathsf{P}_{n,01}\,(1 - (1 - x_n)^k) \\ \mathsf{P}_{n,10}(1 - (1 - x_n)^k) & \mathsf{P}_{n,01} + \mathsf{P}_{n,10}(1 - x_n)^k \end{bmatrix}, \end{aligned} \tag{2}$$

where we have defined $x_n \triangleq \mathsf{P}_{n,01} + \mathsf{P}_{n,10}$. Assuming that channels are positively correlated, i.e., $x_n < 1$, by (2) we have the following lemma.

Lemma 1. *For a positively correlated Markov* ON/OFF *channel with transition probability matrix* $\mathbf{P}_n$*, we have*

1) *The stationary probability* $\pi_{n,\mathsf{ON}} = \mathsf{P}_{n,01}/x_n$.
2) *The k-step transition probability* $\mathsf{P}_{n,01}^{(k)}$ *is nondecreasing in k and* $\mathsf{P}_{n,11}^{(k)}$ *nonincreasing in k. Both* $\mathsf{P}_{n,01}^{(k)}$ *and* $\mathsf{P}_{n,11}^{(k)}$ *converge to* $\pi_{n,\mathsf{ON}}$ *as* $k \to \infty$.

As a corollary of Lemma 1, it follows that

$$\mathsf{P}_{n,11} \geq \mathsf{P}_{n,11}^{(k_1)} \geq \mathsf{P}_{n,11}^{(k_2)} \geq \pi_{n,\mathsf{ON}} \geq \mathsf{P}_{n,01}^{(k_3)} \geq \mathsf{P}_{n,01}^{(k_4)} \geq \mathsf{P}_{n,01} \tag{3}$$

for any integers $k_1 \leq k_2$ and $k_3 \geq k_4$ (see Fig. 2). To maximize network throughput, (3) has some fundamental implications. We note that $\omega_n(t)$ represents the transmission success probability over channel n in slot t. Thus we shall keep serving a channel whenever its information state is $\mathsf{P}_{n,11}$, for it is the best state possible. Second, given that a channel was OFF in its last use, its information state improves as long as the channel remains idle. Thus we shall wait as long as possible before reusing such a channel. Actually, when channels are symmetric ($\mathbf{P}_n = \mathbf{P}$ for all n), it is shown that a myopic policy with this structure maximizes the sum throughput of the network [6].

III. A Round Robin Policy

For any integer $M \in \{1, 2, \dots, N\}$, we present a special round robin policy $\mathsf{RR}(M)$ serving the first M users

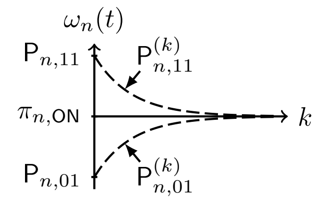

Fig. 2. Diagram of the k-step transition probabilities $\mathsf{P}_{n,01}^{(k)}$ and $\mathsf{P}_{n,11}^{(k)}$ of a positively correlated Markov ON/OFF channel.

$\{1, 2, \dots, M\}$ in the network. The M users are served in the circular order $1 \to 2 \to \cdots \to M \to 1 \to \cdots$. In general, we can use this policy to serve any subset of users. This policy is the fundamental building block of all the results in this paper.

A. The Policy

Round Robin Policy $\mathsf{RR}(M)$:

1) At time 0, the base station starts with channel 1. Suppose initially $\omega_n(0) = \pi_{n,\mathsf{ON}}$ for all n.
2) Suppose at time t, the base station switches to channel n. Transmit a *data* packet to user n with probability $\mathsf{P}_{n,01}^{(M)}/\omega_n(t)$ and a *dummy* packet otherwise. In both cases, we receive ACK/NACK information at the end of the slot.
3) At time $(t+1)$, if a dummy packet is sent at time t, switch to channel $(n \mod M) + 1$ and go to Step 2. Otherwise, keep transmitting data packets over channel n until we receive a NACK. Then switch to channel $(n \mod M) + 1$ and go to Step 2. We note that dummy packets are only sent on the first slot every time the base station switches to a new channel.
4) Update $\boldsymbol{\omega}(t)$ according to (1) in every slot.

Step 2 of $\mathsf{RR}(M)$ only makes sense if $\omega_n(t) \geq \mathsf{P}_{n,01}^{(M)}$, which we prove in the next lemma.

Lemma 2. *Under* $\mathsf{RR}(M)$*, whenever the base station switches to channel* $n \in \{1, 2, \dots, M\}$ *for another round of transmission, its current information state satisfies* $\omega_n(t) \geq \mathsf{P}_{n,01}^{(M)}$.

Proof of Lemma 2: Due to page limit please see [14]. ■

We note that policy $\mathsf{RR}(M)$ is very conservative and not throughput-optimal. For example, we can improve the throughput by always sending data packets but no dummy ones. Also, it does not follow the guidelines we provide at the end of Section II for maximum throughput. Yet, we will see that, in the case of symmetric channels, throughput under $\mathsf{RR}(M)$ is close to optimal when M is large. Moreover, the underlying analysis of $\mathsf{RR}(M)$ is tractable so that we can mix such round robin policies over different subsets of users to form a non-trivial inner capacity bound. The tractability of $\mathsf{RR}(M)$ is because it is equivalent to the following *fictitious* round robin policy (which can be proved as a corollary of Lemma 3 provided later).

Equivalent Fictitious Round Robin:

1) At time 0, start with channel 1.
2) When the base station switches to channel n, *set its*

current information state to $\mathsf{P}_{n,01}^{(M)}$.[3] Keep transmitting data packets over channel n until we receive a NACK. Then switch to channel $(n \mod M) + 1$ and repeat Step 2.

For any round robin policy that serves channels in the circular order $1 \to 2 \to \cdots \to M \to 1 \to \cdots$, the technique of resetting the information state to $\mathsf{P}_{n,01}^{(M)}$ creates a system with an information state that is *worse* than the information state under the actual system. To see this, since in the actual system channels are served in the circular order, after we switch away from serving a particular channel n, we serve the other $(M-1)$ channels for at least one slot each, and so we return to channel n after at least M slots. Thus, its starting information state is always at least $\mathsf{P}_{n,01}^{(M)}$ (the proof is similar to that of Lemma 2). Intuitively, since information states represent the packet transmission success probabilities, resetting them to lower values degrades throughput. This is the reason why our inner capacity bound constructed later using $\mathsf{RR}(M)$ provides a throughput lower bound for a large class of policies.

B. Network Throughput under $\mathsf{RR}(M)$

Next we analyze the throughput vector achieved by $\mathsf{RR}(M)$.

1) General Case: Under $\mathsf{RR}(M)$, let L_{kn} denote the duration of the kth time the base station stays with channel n. A sample path of the $\{L_{kn}\}$ process is

$$(\underbrace{L_{11}, L_{12}, \ldots, L_{1M}}_{\text{round } k=1}, \underbrace{L_{21}, L_{22}, \ldots, L_{2M}}_{\text{round } k=2}, L_{31}, \ldots). \quad (4)$$

The next lemma presents useful properties of L_{kn}, which serve as the foundation of the throughput analysis in the rest of the paper.

Lemma 3. *For any integer k and $n \in \{1, 2, \ldots, M\}$,*

1) *The probability mass function of L_{kn} is independent of k, and is*

$$L_{kn} = \begin{cases} 1 & \text{with prob. } 1 - \mathsf{P}_{n,01}^{(M)} \\ j \geq 2 & \text{with prob. } \mathsf{P}_{n,01}^{(M)} \left(\mathsf{P}_{n,11}\right)^{(j-2)} \mathsf{P}_{n,10}. \end{cases}$$

As a result, for all $k \in \mathbb{N}$ we have

$$\mathbb{E}\left[L_{kn}\right] = 1 + \frac{\mathsf{P}_{n,01}^{(M)}}{\mathsf{P}_{n,10}} = 1 + \frac{\mathsf{P}_{n,01}(1-(1-x_n))^M}{x_n \mathsf{P}_{n,10}}.$$

2) *The number of data packets served in L_{kn} is $(L_{kn}-1)$.*
3) *For every fixed channel n, time durations L_{kn} are* i.i.d. *random variables over all k.*

Proof of Lemma 3:

1) Note that $L_{kn} = 1$ if, on the first slot of serving channel n, either a dummy packet is transmitted or a data packet is transmitted but the channel is OFF. This event occurs with probability

$$\left(1 - \frac{\mathsf{P}_{n,01}^{(M)}}{\omega_n(t)}\right) + \frac{\mathsf{P}_{n,01}^{(M)}}{\omega_n(t)} (1 - \omega_n(t)) = 1 - \mathsf{P}_{n,01}^{(M)}.$$

Next, $L_{kn} = j \geq 2$ if in the first slot a data packet is successfully served, and this is followed by $(j-2)$ consecutive ON slots and one OFF slot. This occurs with probability $\mathsf{P}_{n,01}^{(M)} \left(\mathsf{P}_{n,11}\right)^{(j-2)} \mathsf{P}_{n,10}$. The expectation of L_{kn} can be computed from its probability mass function.

2) We can observe that one data packet is served in every slot of L_{kn} except for the last one (when a dummy packet is sent over channel n, we have $L_{kn} = 1$ and zero data packets are served).
3) At the beginning of every L_{kn}, we observe from the equivalent fictitious round robin policy that $\mathsf{RR}(M)$ effectively fixes $\mathsf{P}_{n,01}^{(M)}$ as the current information state, regardless of the true current state $\omega_n(t)$. Neglecting $\omega_n(t)$ is to discard all system history, including all past $L_{k'n}$ for all $k' < k$. Thus L_{kn} are i.i.d.. Specifically, for any $k' < k$ and integers $l_{k'}$ and l_k we have

$$\Pr\left[L_{kn} = l_k \mid L_{k'n} = l_{k'}\right] = \Pr\left[L_{kn} = l_k\right].$$

■

Now we can derive the throughput vector supported by $\mathsf{RR}(M)$. Fix an integer $K > 0$. By Lemma 3, the time average throughput over channel n after all channels finish their Kth rounds, which we denote by $\mu_n(K)$, is

$$\mu_n(K) \triangleq \frac{\sum_{k=1}^K (L_{kn} - 1)}{\sum_{k=1}^K \sum_{n=1}^M L_{kn}}.$$

Passing $K \to \infty$, we get

$$\begin{aligned} &\lim_{K\to\infty} \mu_n(K) \\ &= \lim_{K\to\infty} \frac{\sum_{k=1}^K (L_{kn} - 1)}{\sum_{k=1}^K \sum_{n=1}^M L_{kn}} \\ &= \lim_{K\to\infty} \frac{(1/K)\sum_{k=1}^K (L_{kn} - 1)}{\sum_{n=1}^M (1/K)\sum_{k=1}^K L_{kn}} \\ &\overset{(a)}{=} \frac{\mathbb{E}\left[L_{1n}\right] - 1}{\sum_{n=1}^M \mathbb{E}\left[L_{1n}\right]} \\ &\overset{(b)}{=} \frac{\mathsf{P}_{n,01}(1-(1-x_n)^M)/(x_n \mathsf{P}_{n,10})}{M + \sum_{n=1}^M \mathsf{P}_{n,01}(1-(1-x_n)^M)/(x_n \mathsf{P}_{n,10})}, \end{aligned} \quad (5)$$

where (a) is by the Law of Large Numbers (noting by Lemma 3 that L_{kn} are i.i.d. over k), and (b) is by Lemma 3.

2) Symmetric Case: We are particularly interested in the sum throughput under $\mathsf{RR}(M)$ when channels are symmetric, that is, all channels have the same statistics $\mathbf{P}_n = \mathbf{P}$ for all n. In this case, by channel symmetry every channel has the same throughput. From (5), we can show the sum throughput is

$$\sum_{n=1}^M \lim_{K\to\infty} \mu_n(K) = \frac{\mathsf{P}_{01}(1-(1-x)^M)}{x\,\mathsf{P}_{10} + \mathsf{P}_{01}(1-(1-x)^M)},$$

where in the last term the subscript n is dropped due to channel symmetry. It is handy to define a function $c_{(\cdot)} : \mathbb{N} \to \mathbb{R}$ as

$$c_M \triangleq \frac{\mathsf{P}_{01}(1-(1-x)^M)}{x\,\mathsf{P}_{10} + \mathsf{P}_{01}(1-(1-x)^M)}, \quad x \triangleq \mathsf{P}_{01} + \mathsf{P}_{10}, \quad (6)$$

and define $c_\infty \triangleq \lim_{M\to\infty} c_M = \mathsf{P}_{01}/(x\mathsf{P}_{10} + \mathsf{P}_{01})$ (note that $x < 1$ because every channel is positively correlated over time

[3] In reality we cannot *set* the information state of a channel, and therefore the policy is fictitious.

slots). The function $c_{(\cdot)}$ will be used extensively in this paper. We summarize the above derivation in the next lemma.

Lemma 4. *Policy* $\mathsf{RR}(M)$ *serves channel* $n \in \{1, 2, \dots, M\}$ *with throughput*

$$\frac{\mathsf{P}_{n,01}(1-(1-x_n)^M)/(x_n\mathsf{P}_{n,10})}{M+\sum_{n=1}^{M}\mathsf{P}_{n,01}(1-(1-x_n)^M)/(x_n\mathsf{P}_{n,10})}.$$

In particular, in symmetric channels the sum throughput under $\mathsf{RR}(M)$ *is* c_M *defined as*

$$c_M = \frac{\mathsf{P}_{01}(1-(1-x)^M)}{x\,\mathsf{P}_{10}+\mathsf{P}_{01}(1-(1-x)^M)}, \quad x = \mathsf{P}_{01}+\mathsf{P}_{10},$$

and every channel has throughput c_M/M.

We remark that the sum throughput c_M of $\mathsf{RR}(M)$ in the symmetric case is nondecreasing in M, and thus can be improved by serving more channels. Interestingly, here we see that the sum throughput is improved by having *multiuser diversity* in the network, even though instantaneous channel states are never known.

C. How Good is $\mathsf{RR}(M)$?

Next, in symmetric channels, we quantify how close the sum throughput c_M is to optimal. The following lemma presents a useful upper bound on the maximum sum throughput.

Lemma 5 ([5], [6])**.** *In symmetric channels, any scheduling policy that confines to our model has sum throughput less than or equal to* c_∞.[4]

By Lemma 4 and 5, the loss of the sum throughput of $\mathsf{RR}(M)$ is no larger than $c_\infty - c_M$. Define $\widetilde{c}_M$ as

$$\widetilde{c}_M \triangleq \frac{\mathsf{P}_{01}(1-(1-x)^M)}{x\mathsf{P}_{10}+\mathsf{P}_{01}} = c_\infty(1-(1-x)^M)$$

and note that $\widetilde{c}_M \le c_M \le c_\infty$. It follows

$$c_\infty - c_M \le c_\infty - \widetilde{c}_M = c_\infty(1-x)^M. \tag{7}$$

The last term of (7) decreases to zero geometrically fast as M increases. This indicates that $\mathsf{RR}(M)$ yields near-optimal sum throughput even when it only serves a moderately large number of channels.

IV. Randomized Round Robin Policy, Inner and Outer Capacity Bound

A. Randomized Round Robin Policy

Lemma 4 specifies the throughput vector achieved by implementing $\mathsf{RR}(M)$ over a particular collection of M channels. Here we are interested in the set of throughput vectors achievable by randomly mixing $\mathsf{RR}(M)$-like policies over different channel subsets and allowing a different round-robin ordering on each subset. To generalize the $\mathsf{RR}(M)$ policy, first let Φ denote the set of all N-dimensional binary vectors excluding the all-zero vector $(0, 0, \dots, 0)$. For any binary vector $\boldsymbol{\phi} = (\phi_1, \phi_2, \dots, \phi_N)$ in Φ, we say channel n is *active* in $\boldsymbol{\phi}$ if $\phi_n = 1$. Each vector $\boldsymbol{\phi} \in \Phi$ represents a different subset of active channels. We denote by $M(\boldsymbol{\phi})$ the number of active channels in $\boldsymbol{\phi}$.

For each $\boldsymbol{\phi} \in \Phi$, consider the following round robin policy $\mathsf{RR}(\boldsymbol{\phi})$ that serves active channels in $\boldsymbol{\phi}$ in every round.

Dynamic Round Robin Policy $\mathsf{RR}(\boldsymbol{\phi})$:

1) *Deciding the service order in each round*:
   At the beginning of each round, we denote by τ_n the time duration between the last use of channel n and the beginning of the current round. Active channels in $\boldsymbol{\phi}$ are served in the decreasing order of τ_n in this round (in other words, the active channel that is *least recently used* is served first).
2) *On each active channel in a round*:
   a) Suppose at time t the base station switches to channel n. Transmit a *data* packet to user n with probability $\mathsf{P}_{n,01}^{(M(\boldsymbol{\phi}))}/\omega_n(t)$ and a *dummy* packet otherwise. In both cases, we receive ACK/NACK information at the end of the slot.
   b) At time $(t+1)$, if a dummy packet is sent at time t, switch to the next active channel following the order given in Step 1. Otherwise, keep transmitting data packets over channel n until we receive a NACK. Then switch to the next active channel and go to Step 2a. We note that dummy packets are only sent on the first slot every time the base station switches to a new channel.
3) Update $\boldsymbol{\omega}(t)$ according to (1) in every slot.

Using $\mathsf{RR}(\boldsymbol{\phi})$ as building blocks, we consider the following class of *randomized round robin policies*.

Randomized Round Robin Policy RandRR:

1) Pick $\boldsymbol{\phi} \in \Phi$ with probability $\alpha_{\boldsymbol{\phi}}$, where $\sum_{\boldsymbol{\phi}\in\Phi} \alpha_{\boldsymbol{\phi}} = 1$.
2) Run policy $\mathsf{RR}(\boldsymbol{\phi})$ for one round. Then go to Step 1.

Note that active channels may be served in different order in different rounds, according to the least-recently-used service order. This allows more time for OFF channels to return to better information states (note that $\mathsf{P}_{n,01}^{(k)}$ is nondecreasing in k) and thus improves throughput. The next lemma guarantees the feasibility of executing any $\mathsf{RR}(\boldsymbol{\phi})$ policy in RandRR (similar to Lemma 2, whenever the base station switches to a new channel n, we need $\omega_n(t) \ge \mathsf{P}_{n,01}^{(M(\boldsymbol{\phi}))}$ in Step 2a of $\mathsf{RR}(\boldsymbol{\phi})$).

Lemma 6. *When* $\mathsf{RR}(\boldsymbol{\phi})$ *is chosen by* RandRR *for a new round of transmission, every active channel* n *in* $\boldsymbol{\phi}$ *starts with information state no worse than* $\mathsf{P}_{n,01}^{(M(\boldsymbol{\phi}))}$.

Proof of Lemma 6: See [14]. ■

Although RandRR randomly selects subsets of users and serves them in an order that depends on previous choices, we can surprisingly analyze its throughput. This is done by using the throughput analysis of $\mathsf{RR}(M)$, as shown in the following corollary to Lemma 3:

Corollary 1. *For each policy* $\mathsf{RR}(\boldsymbol{\phi})$, $\boldsymbol{\phi} \in \Phi$*, within time periods in which* $\mathsf{RR}(\boldsymbol{\phi})$ *is executed by* RandRR*, denote by*

[4] We note that the throughput analysis in [5] makes a minor assumption on the existence of some limiting time average. Using similar ideas of [5], in Theorem 2 of Section IV-C we will construct an upper bound on the maximum sum throughput for general positively correlated Markov ON/OFF channels. When restricted to the symmetric case, we get the same upper bound without any assumption (see Corollary 3).

L_{kn}^{ϕ} *the duration of the kth time the base station stays with active channel* n*. Then:*

1) *The probability mass function of* L_{kn}^{ϕ} *is independent of* k*, and is*

$$L_{kn}^{\phi} = \begin{cases} 1 & \textit{with prob. } 1 - \mathsf{P}_{n,01}^{(M(\phi))} \\ j \geq 2 & \textit{with prob. } \mathsf{P}_{n,01}^{(M(\phi))} \left(\mathsf{P}_{n,11}\right)^{(j-2)} \mathsf{P}_{n,10}. \end{cases}$$

As a result, for all $k \in \mathbb{N}$ *we have*

$$\mathbb{E}\left[L_{kn}^{\phi}\right] = 1 + \frac{\mathsf{P}_{n,01}^{(M(\phi))}}{\mathsf{P}_{n,10}}. \tag{8}$$

2) *The number of data packets served in* L_{kn}^{ϕ} *is* $(L_{kn}^{\phi} - 1)$*.*
3) *For every fixed* ϕ *and every fixed active channel* n *in* ϕ*,* L_{kn}^{ϕ} *are i.i.d. random variables over all* k*.*

B. Achievable Network Capacity — An Inner Capacity Bound

Using Corollary 1, next we present the achievable rate region of the class of RandRR policies. For each RR(ϕ) policy, define an N-dimensional vector $\boldsymbol{\eta}^{\phi} = (\eta_1^{\phi}, \eta_2^{\phi}, \dots, \eta_N^{\phi})$ where

$$\eta_n^{\phi} \triangleq \begin{cases} \frac{\mathbb{E}\left[L_{1n}^{\phi}\right]-1}{\sum_{n:\phi_n=1} \mathbb{E}\left[L_{1n}^{\phi}\right]} & \text{if channel } n \text{ is active in } \phi, \\ 0 & \text{otherwise,} \end{cases} \tag{9}$$

where $\mathbb{E}\left[L_{1n}^{\phi}\right]$ is given in (8). Intuitively, by the analysis prior to Lemma 4, round robin policy RR(ϕ) yields throughput η_n^{ϕ} over channel n for each $n \in \{1, 2, \dots, N\}$. Incorporating all possible random mixtures of RR(ϕ) policies for different ϕ, RandRR can support any data rate vector that is entrywise dominated by a convex combination of vectors $\{\boldsymbol{\eta}^{\phi}\}_{\phi \in \Phi}$ as shown by the next theorem.

Theorem 1 (Generalized Inner Capacity Bound)**.** *The class of* RandRR *policies supports all data rate vectors* $\boldsymbol{\lambda}$ *in the set* Λ_{int} *defined as*

$$\Lambda_{\mathit{int}} \triangleq \left\{ \boldsymbol{\lambda} \mid \mathbf{0} \leq \boldsymbol{\lambda} \leq \boldsymbol{\mu},\ \boldsymbol{\mu} \in \operatorname{conv}\left(\left\{ \boldsymbol{\eta}^{\phi} \right\}_{\phi \in \Phi} \right) \right\},$$

where $\boldsymbol{\eta}^{\phi}$ *is defined in* (9)*,* $\operatorname{conv}(A)$ *denotes the convex hull of set* A*, and* $\leq$ *is taken entrywise.*

Proof of Theorem 1: See [14]. ∎

Next is a corollary to Theorem 1 for symmetric channels.

Corollary 2 (Inner Capacity Bound for Symmetric Channels)**.** *In symmetric channels, the class of* RandRR *policies supports all rate vectors* $\boldsymbol{\lambda} \in \Lambda_{\mathit{int}}$ *where*

$$\Lambda_{\mathit{int}} = \left\{ \boldsymbol{\lambda} \mid \mathbf{0} \leq \boldsymbol{\lambda} \leq \boldsymbol{\mu},\, \boldsymbol{\mu} \in \operatorname{conv}\left(\left\{ \frac{c_{M(\phi)}}{M(\phi)} \phi \right\}_{\phi \in \Phi} \right) \right\},$$

where $c_{M(\phi)}$ *is defined in* (6)*.*

An example of the inner capacity bound and a simple queue-dependent dynamic policy that supports all data rates within this nontrivial inner bound will be provided later.

C. Outer Capacity Bound

We construct an outer bound on Λ using several novel ideas. First, by state aggregation, we transform the information state process $\{\omega_n(t)\}$ for each channel n into non-stationary two-state Markov chains (in Fig. 4 provided later). Second, we create a set of bounding stationary Markov chains (in Fig. 5 provided later), which has the structure of a multi-armed bandit system. Finally, we create an outer capacity bound by relating the bounding model to the original non-stationary Markov chains using stochastic coupling. We note that since the control of the set of information state processes $\{\omega_n(t)\}$ for all n can be viewed as a restless bandit problem [15], it is interesting to see how we bound the optimal performance of a restless bandit problem by a related multi-armed bandit system.

We first *map* channel information states $\omega_n(t)$ into *modes* for each $n \in \{1, 2, \dots, N\}$. Inspired by (3), we observe that each channel n must be in one of the following two modes:

- M1 The last observed state is ON, and the channel has not been seen (through feedback) to turn OFF. In this mode the information state $\omega_n(t) \in [\pi_{n,\mathsf{ON}}, \mathsf{P}_{n,11}]$.
- M2 The last observed state is OFF, and the channel has not been seen to turned ON. Here $\omega_n(t) \in [\mathsf{P}_{n,01}, \pi_{n,\mathsf{ON}}]$.

On channel n, recall that $\mathcal{W}_n$ is the state space of $\omega_n(t)$, and define a map $f_n : \mathcal{W}_n \to \{\mathsf{M1}, \mathsf{M2}\}$ where

$$f_n(\omega_n(t)) = \begin{cases} \mathsf{M1} & \text{if } \omega_n(t) \in (\pi_{n,\mathsf{ON}}, \mathsf{P}_{n,11}], \\ \mathsf{M2} & \text{if } \omega_n(t) \in [\mathsf{P}_{n,01}, \pi_{n,\mathsf{ON}}]. \end{cases}$$

This mapping is illustrated in Fig. 3.

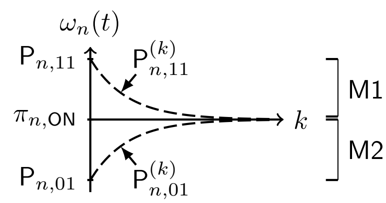

Fig. 3. The mapping f_n from information states $\omega_n(t)$ to modes $\{\mathsf{M1}, \mathsf{M2}\}$.

For any information state process $\{\omega_n(t)\}$ (controlled by some scheduling policy), the corresponding mode transition process under f_n can be represented by the Markov chains shown in Fig. 4. Specifically, when channel n is served in a slot, the associated mode transition follows the upper non-stationary chain of Fig. 4. When channel n is idled in a slot, the mode transition follows the lower stationary chain of Fig. 4. In the upper chain of Fig. 4, regardless what the current mode is, mode M1 is visited in the next slot if and only if channel n is ON in the current slot, which occurs with probability $\omega_n(t)$. In the lower chain of Fig. 4, when channel n is idled, its information state changes from a k-step transition probability to the $(k+1)$-step transition probability with the same most recent observed channel state. Therefore, the next mode stays the same as the current mode. We emphasize that, in the upper chain of Fig. 4, at mode M1 we always have $\omega_n(t) \leq \mathsf{P}_{n,11}$, and at mode M2 it is $\omega_n(t) \leq \pi_{n,\mathsf{ON}}$. A packet is served if and only if M1 is visited in the upper chain of Fig. 4.

To upper bound throughput, we compare Fig. 4 to the mode transition diagrams in Fig. 5 that corresponds to a fictitious model for channel n. This fictitious channel has constant information state $\omega_n(t) = \mathsf{P}_{n,11}$ whenever it is in mode M1,

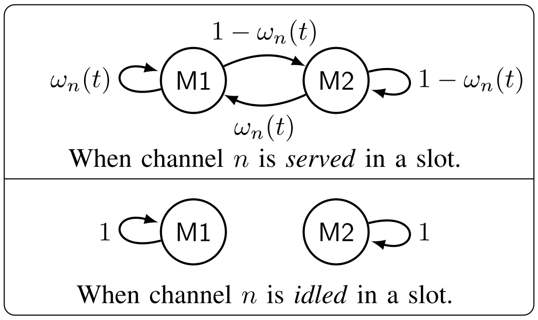

Fig. 4. Mode transition diagrams for the real channel n.

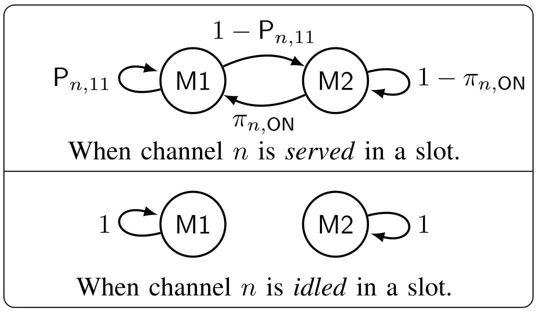

Fig. 5. Mode transition diagrams for the fictitious channel n.

and $\omega_n(t) = \pi_{n,\mathsf{ON}}$ whenever it is in M2. In other words, when the fictitious channel n is in mode M1 (or M2), it sets its current information state to be the best state possible when the corresponding real channel n is in the same mode. It follows that, when both the real and the fictitious channel n are served, the probabilities of transitions M1 $\to$ M1 and M2 $\to$ M1 in the upper chain of Fig. 5 are greater than or equal to those in Fig. 4, respectively. In other words, the upper chain of Fig. 5 is *more likely* to go to mode M1 and serve packets than that of Fig. 4. Therefore, intuitively, if we serve both the real and the fictitious channel n in the same infinite sequence of time slots, the fictitious channel n will yield higher throughput for all n. This observation is made precise by the next lemma.

Lemma 7. *Consider two discrete-time Markov chains $\{X(t)\}$ and $\{Y(t)\}$ both with state space $\{0,1\}$. Suppose $\{X(t)\}$ is stationary and ergodic with transition probability matrix*

$$\mathbf{P} = \begin{bmatrix} \mathsf{P}_{00} & \mathsf{P}_{01} \\ \mathsf{P}_{10} & \mathsf{P}_{11} \end{bmatrix},$$

and $\{Y(t)\}$ is non-stationary with

$$\mathbf{Q}(t) = \begin{bmatrix} \mathsf{Q}_{00}(t) & \mathsf{Q}_{01}(t) \\ \mathsf{Q}_{10}(t) & \mathsf{Q}_{11}(t) \end{bmatrix}.$$

Assume $\mathsf{P}_{01} \geq \mathsf{Q}_{01}(t)$ and $\mathsf{P}_{11} \geq \mathsf{Q}_{11}(t)$ for all t. In $\{X(t)\}$, let $\pi_X(1)$ denote the stationary probability of state 1; $\pi_X(1) = \mathsf{P}_{01}/(\mathsf{P}_{01}+\mathsf{P}_{10})$. In $\{Y(t)\}$, define

$$\pi_Y(1) \triangleq \limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1} Y(t)$$

as the limiting fraction of time $\{Y(t)\}$ stays at state 1. Then we have $\pi_X(1) \geq \pi_Y(1)$.

Proof of Lemma 7: See [14]. ∎

We note that executing a scheduling policy in the network is to generate a sequence of channel selection decisions. By Lemma 7, if we apply the same sequence of channel selection decisions of some scheduling policy to the set of fictitious channels, we will get higher throughput on every channel. A direct consequence of this is that the maximum sum throughput over the fictitious channels is greater than or equal to that over the real channels.

Lemma 8. *The maximum sum throughput over the set of fictitious channels is no more than*

$$\max_{n\in\{1,2,\dots,N\}} \{c_{n,\infty}\}, \quad c_{n,\infty} \triangleq \frac{\mathsf{P}_{n,01}}{x_n \mathsf{P}_{n,10} + \mathsf{P}_{n,01}}.$$

Proof of Lemma 8: Finding the maximum sum throughput over fictitious channels in Fig. 5 is equivalent to solving a multi-armed bandit problem [16] with each channel acting as an arm (see Fig. 5 and note that a channel can change mode only when it is served), and one unit of reward is earned if a packet is delivered (recall that a packet is served if and only if mode M1 is visited in the upper chain of Fig. 5). The optimal solution to the multi-armed bandit system is to always play the arm (channel) with the largest average reward (throughput). The average reward over channel n is equal to the stationary probability of mode M1 in the upper chain of Fig. 5, which is

$$\frac{\pi_{n,\mathsf{ON}}}{\mathsf{P}_{n,10} + \pi_{n,\mathsf{ON}}} = \frac{\mathsf{P}_{n,01}}{x_n \mathsf{P}_{n,10} + \mathsf{P}_{n,01}}.$$

This finishes the proof. ∎

Together with the fact that throughput over any real channel n cannot exceed its stationary ON probability $\pi_{n,\mathsf{ON}}$, we have constructed an outer bound on the network capacity region Λ (the proof follows the above discussions and thus is omitted).

Theorem 2. *(Generalized Outer Capacity Bound): Any supportable throughput vector $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \dots, \lambda_N)$ necessarily satisfies $\lambda_n \leq \pi_{n,\mathsf{ON}}$ for all $n \in \{1,\dots,N\}$ and*

$$\sum_{n=1}^{N} \lambda_n \leq \max_{n\in\{1,2,\dots,N\}} \{c_{n,\infty}\},\ c_{n,\infty} = \frac{\mathsf{P}_{n,01}}{x_n \mathsf{P}_{n,10} + \mathsf{P}_{n,01}}.$$

These $(N+1)$ hyperplanes form an outer bound Λ_{out} on Λ.

Corollary 3 (Outer Capacity Bound for Symmetric Channels)**.** *In symmetric channels with $\mathbf{P}_n = \mathbf{P}$ for all n, we have*

$$\Lambda_{out} = \left\{ \boldsymbol{\lambda} \mid \sum_{n=1}^{N} \lambda_n \leq c_\infty,\ 0 \leq \lambda_n \leq \pi_{\mathsf{ON}}\ \forall\, n \right\}, \qquad (10)$$

where the subscript n is dropped due to channel symmetry.

D. A Two-User Example on Symmetric Channels

Here we consider a two-user example on symmetric channels. For simplicity we will drop the subscript n in notations. From Corollary 3, we have the outer bound

$$\Lambda_{\text{out}} = \left\{ \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} \;\middle|\; \begin{array}{c} 0 \leq \lambda_n \leq \mathsf{P}_{01}/x, \text{ for } 1 \leq n \leq 2, \\ \lambda_1 + \lambda_2 \leq \mathsf{P}_{01}/(x\mathsf{P}_{10} + \mathsf{P}_{01}), \\ x = \mathsf{P}_{01} + \mathsf{P}_{10} \end{array} \right\}.$$

For the inner bound Λ_{int}, we note that policy RandRR can execute three round robin policies $\mathsf{RR}(\phi)$ for $\phi \in \Phi = \{(1,1),(0,1),(1,0)\}$. From Corollary 2, we have

$$\Lambda_{\text{int}} = \left\{ \begin{bmatrix}\lambda_1\\ \lambda_2\end{bmatrix} \;\middle|\; \begin{array}{c} 0 \le \lambda_n \le \mu_n, \text{ for } 1 \le n \le 2, \\ \begin{bmatrix}\mu_1\\ \mu_2\end{bmatrix} \in \text{conv}\left(\left\{\begin{bmatrix}c_2/2\\ c_2/2\end{bmatrix}, \begin{bmatrix}c_1\\ 0\end{bmatrix}, \begin{bmatrix}0\\ c_1\end{bmatrix}\right\}\right)\end{array} \right\}.$$

Under the special case $\mathsf{P}_{01} = \mathsf{P}_{10} = 0.2$, the two bounds λ_{int} and Λ_{out} are shown in Fig. 6.

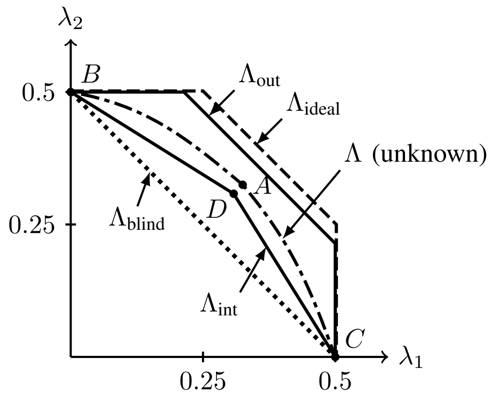

Fig. 6. Comparison of rate regions under different assumptions.

In Fig. 6, we also compare Λ_{int} and Λ_{out} with other rate regions. Set Λ_{ideal} is the ideal capacity region when instantaneous channel states are known without causing any (timing) overhead [17]. Next, work [5] shows that the maximum sum throughput in this network is achieved at point $A = (0.325, 0.325)$. The (unknown) network capacity region Λ is bounded between Λ_{int} and Λ_{out}, and has boundary points B, A, and C. Since the boundary of Λ is a concave curve connecting B, A, and C, we envision that Λ shall contain but be very close to Λ_{int}.

Finally, the rate region Λ_{blind} is rendered by completely neglecting channel memory and treating the channels as i.i.d. over slots [1]. We observe the throughput gain $\Lambda_{\text{int}} \setminus \Lambda_{\text{blind}}$, as much as 23% in this example, is achieved by incorporating channel memory. In general, if channels are symmetric and treated as i.i.d. over slots, the maximum sum throughput in the network is $\pi_{\mathsf{ON}} = c_1$. Then the maximum throughput gain of RandRR using channel memory is $c_N - c_1$, which as $N \to \infty$ converges to

$$c_\infty - c_1 = \frac{\mathsf{P}_{01}}{x\mathsf{P}_{10} + \mathsf{P}_{01}} - \frac{\mathsf{P}_{01}}{\mathsf{P}_{01} + \mathsf{P}_{10}},$$

which is controlled by the factor $x = \mathsf{P}_{01} + \mathsf{P}_{10}$.

V. Proximity of the Inner Bound to the True Capacity Region — Symmetric Case

Next we bound the closeness of the boundaries of Λ_{int} and Λ in the case of symmetric channels. In Section III-C, by choosing $M = N$, we have provided such analysis for the boundary point in the direction $(1,1,\ldots,1)$. Here we generalize to all boundary points. Define

$$\mathcal{V} \triangleq \left\{ (v_1, v_2, \ldots, v_N) \;\middle|\; \begin{array}{c} v_n \ge 0 \text{ for } 1 \le n \le N, \\ v_n > 0 \text{ for at least one } n \end{array} \right\}$$

as a set of directional vectors. For any $\boldsymbol{v} \in \mathcal{V}$, let $\boldsymbol{\lambda}^{\text{int}} = (\lambda_1^{\text{int}}, \lambda_2^{\text{int}}, \ldots, \lambda_N^{\text{int}})$ and $\boldsymbol{\lambda}^{\text{out}} = (\lambda_1^{\text{out}}, \lambda_2^{\text{out}}, \ldots, \lambda_N^{\text{out}})$ be the boundary point of Λ_{int} and Λ_{out} in the direction of $\boldsymbol{v}$, respectively. It is useful to compute $\sum_{n=1}^N (\lambda_n^{\text{out}} - \lambda_n^{\text{int}})$, because it upper bounds the loss of the sum throughput of Λ_{int} from Λ in the direction of $\boldsymbol{v}$. We note that computing $\boldsymbol{\lambda}^{\text{int}}$ in an arbitrary direction is difficult. Thus we will find an upper bound on $\sum_{n=1}^N (\lambda_n^{\text{out}} - \lambda_n^{\text{int}})$.

Definition 1. *For any $\boldsymbol{v} \in \mathcal{V}$, we say $\boldsymbol{v}$ is d-*user diverse *if $\boldsymbol{v}$ can be written as a* positive *combination of vectors in Φ_d, where Φ_d denotes the set of N-dimensional binary vectors having d entries be 1. Define*

$$d(\boldsymbol{v}) \triangleq \max_{1 \le d \le N} \{d \mid \boldsymbol{v} \text{ is } d\text{-user diverse}\},$$

and we shall say $\boldsymbol{v}$ is maximally $d(\boldsymbol{v})$-user diverse.

The notion of $d(\boldsymbol{v})$ is well-defined because every $\boldsymbol{v}$ must be 1-user diverse.[5] Definition 1 is the most useful to us through the next lemma.

Lemma 9. *The boundary point of Λ_{int} in the direction of $\boldsymbol{v} \in \mathcal{V}$ has sum throughput at least $c_{d(\boldsymbol{v})}$, where*

$$c_{d(\boldsymbol{v})} \triangleq \frac{\mathsf{P}_{01}(1-(1-x)^{d(\boldsymbol{v})})}{x\,\mathsf{P}_{10} + \mathsf{P}_{01}(1-(1-x)^{d(\boldsymbol{v})})}, \quad x \triangleq \mathsf{P}_{01} + \mathsf{P}_{10}.$$

Proof of Lemma 9: See [14]. ■

Fig. 7 provides an example of Lemma 9 in the two-user symmetric system in Section IV-D. We observe that

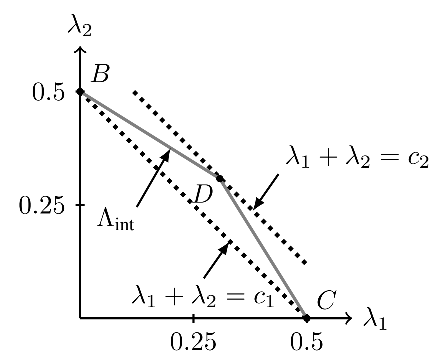

Fig. 7. An example for Lemma 9 in the two-user symmetric network. Point B and C achieve sum throughput $c_1 = \pi_{\mathsf{ON}} = 0.5$, and the sum throughput at D is $c_2 \approx 0.615$. Any other boundary point of Λ_{int} has sum throughput between c_1 and c_2.

direction $(1,1)$, the one that passes point D in Fig. 7, is the only direction that is maximally 2-user diverse. The sum throughput c_2 is achieved at D. For all the other directions, they are maximally 1-user diverse and, from Fig. 7, only sum throughput c_1 is guaranteed along those directions. In

[5]The set $\Phi_1 = \{\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_N\}$ is the collection of unit coordinate vectors where $\boldsymbol{e}_n$ has its nth entry be 1 and 0 otherwise. Any vector $\boldsymbol{v} \in \mathcal{V}$, $\boldsymbol{v} = (v_1, v_2, \ldots, v_N)$, can be written as $\boldsymbol{v} = \sum_{v_n > 0} v_n \boldsymbol{e}_n$.

general, geometrically we can show that a maximally d-user diverse vector, say $\boldsymbol{v}_d$, forms a smaller angle with the all-1 vector $(1, 1, \ldots, 1)$ than a maximally d'-user diverse vector, say $\boldsymbol{v}_{d'}$, does if $d' < d$. In other words, data rates along $\boldsymbol{v}_d$ are *more balanced* than those along $\boldsymbol{v}_{d'}$. Lemma 9 states that we guarantee to support higher sum throughput if the user traffic is more balanced.

A. Proximity Analysis

We use the notion of $d(\boldsymbol{v})$ to upper bound $\sum_{n=1}^{N}(\lambda_n^{\text{out}} - \lambda_n^{\text{int}})$ in any direction $\boldsymbol{v} \in \mathcal{V}$. Let $\boldsymbol{\lambda}^{\text{out}} = \theta\boldsymbol{\lambda}^{\text{int}}$ (i.e., $\lambda_n^{\text{out}} = \theta\lambda_n^{\text{int}}$ for all n) for some $\theta \geq 1$. By (10), the boundary of Λ_{out} is characterized by the interaction of the $(N+1)$ hyperplanes $\sum_{n=1}^{N} \lambda_n = c_\infty$ and $\lambda_n = \pi_{\mathsf{ON}}$ for each $n \in \{1, 2, \ldots, N\}$. Specifically, in any given direction, if we consider the cross points on all the hyperplanes in that direction, the boundary point $\boldsymbol{\lambda}^{\text{out}}$ is the one closest to the origin. We do not know which hyperplane $\boldsymbol{\lambda}^{\text{out}}$ is on, and thus need to consider all $(N+1)$ cases. If $\boldsymbol{\lambda}^{\text{out}}$ is on the plane $\sum_{n=1}^{N} \lambda_n = c_\infty$, i.e., $\sum_{n=1}^{N} \lambda_n^{\text{out}} = c_\infty$, we get

$$\sum_{n=1}^{N}(\lambda_n^{\text{out}} - \lambda_n^{\text{int}}) \overset{(a)}{\leq} c_\infty - c_{d(\boldsymbol{v})} \overset{(b)}{\leq} c_\infty(1-x)^{d(\boldsymbol{v})},$$

where (a) is by Lemma 9 and (b) is by (7). If $\boldsymbol{\lambda}^{\text{out}}$ is on the plane $\lambda_n = \pi_{\mathsf{ON}}$ for some n, then $\theta = \pi_{\mathsf{ON}}/\lambda_n^{\text{int}}$. It follows

$$\sum_{n=1}^{N}(\lambda_n^{\text{out}} - \lambda_n^{\text{int}}) = (\theta - 1)\sum_{n=1}^{N}\lambda_n^{\text{int}} \leq \left(\frac{\pi_{\mathsf{ON}}}{\lambda_n^{\text{int}}} - 1\right) c_\infty.$$

The above discussions lead to the next lemma.

Lemma 10. *The loss of the sum throughput of Λ_{int} from Λ in the direction of $\boldsymbol{v}$ is upper bounded by*

$$\min\left[c_\infty(1-x)^{d(\boldsymbol{v})}, \min_{1\leq n\leq N}\left\{\left(\frac{\pi_{\mathsf{ON}}}{\lambda_n^{int}} - 1\right)c_\infty\right\}\right]$$
$$= c_\infty \min\left[(1-x)^{d(\boldsymbol{v})}, \frac{\pi_{\mathsf{ON}}}{\max_{1\leq n\leq N}\{\lambda_n^{int}\}} - 1\right]. \quad (11)$$

Lemma 10 shows that, if data rates are more balanced, namely, have a larger $d(\boldsymbol{v})$, the sum throughput loss is dominated by the first term in the minimum of (11) and decreases to 0 geometrically fast with $d(\boldsymbol{v})$. If data rates are biased toward a particular user, the second term in the minimum of (11) captures the throughput loss, which goes to 0 as the rate of the favored user goes to the single-user capacity π_{ON}.

VI. Throughput-Achieving Queue-dependent Round Robin Policy

Let $a_n(t)$, for $1 \leq n \leq N$, be the number of exogenous packet arrivals destined for user n in slot t. Suppose $a_n(t)$ are independent across users, i.i.d. over slots with rate $\mathbb{E}[a_n(t)] = \lambda_n$, and $a_n(t)$ is bounded with $0 \leq a_n(t) \leq A_{\text{max}}$, where A_{max} is a finite integer. Let $U_n(t)$ be the backlog of user-n packets queued at the base station at time t. Define $\boldsymbol{U}(t) \triangleq (U_1(t), U_2(t), \ldots, U_N(t))$ and suppose $U_n(0) = 0$ for all n. The queue process $\{U_n(t)\}$ evolves as

$$U_n(t+1) = \max\left[U_n(t) - \mu_n(s_n(t), t), 0\right] + a_n(t), \quad (12)$$

where $\mu_n(s_n(t), t) \in \{0, 1\}$ is the service rate allocated to user n in slot t. We have $\mu_n(s_n(t), t) = 1$ if user n is served and $s_n(t) = \mathsf{ON}$, and 0 otherwise. In the rest of the paper we drop $s_n(t)$ in $\mu_n(s_n(t), t)$ and use $\mu_n(t)$ for notational simplicity. We say the network is (strongly) stable if

$$\limsup_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1}\sum_{n=1}^{N}\mathbb{E}\left[U_n(\tau)\right] < \infty.$$

Consider a rate vector $\boldsymbol{\lambda}$ interior to the inner capacity region bound Λ_{int} given in Theorem 1. Namely, there exists an $\epsilon > 0$ and a probability distribution $\{\beta_{\boldsymbol{\phi}}\}_{\boldsymbol{\phi}\in\Phi}$ such that

$$\lambda_n + \epsilon < \sum_{\boldsymbol{\phi}\in\Phi}\beta_{\boldsymbol{\phi}}\eta_n^{\boldsymbol{\phi}}, \quad \text{for all } 1 \leq n \leq N, \quad (13)$$

where $\eta_n^{\boldsymbol{\phi}}$ is defined in (9). By Theorem 1, there exists a RandRR policy that yields service rates equal to the right-side of (13) and thus stabilizes the network with arrival rate vector $\boldsymbol{\lambda}$ [18, Lemma 3.6]. The existence of this policy is useful and we shall denote it by RandRR^*. Recall that on each new scheduling round, the policy RandRR^* randomly picks a binary vector $\boldsymbol{\phi}$ using probabilities $\alpha_{\boldsymbol{\phi}}$ (defined over all of the $(2^N - 1)$ subsets of users). The $M(\boldsymbol{\phi})$ active users in $\boldsymbol{\phi}$ are served for one round by the round robin policy $\mathsf{RR}(\boldsymbol{\phi})$, serving the least recently used users first. However, solving for the probabilities needed to implement the RandRR^* policy that yields (13) is intractable when N is large, because we need to find a right combination of $(2^N - 1)$ unknown probabilities $\{\alpha_{\boldsymbol{\phi}}\}_{\boldsymbol{\phi}\in\Phi}$ for a RandRR policy whose achievable throughput vector satisfies (13). Instead, we use the following simple *queue-dependent* policy.

Queue-dependent Round Robin Policy (QRR):

1) Start with $t = 0$.
2) At time t, observe the current queue backlog vector $\boldsymbol{U}(t)$ and find the binary vector $\boldsymbol{\phi}(t) \in \Phi$ defined as[6]

$$\boldsymbol{\phi}(t) \triangleq \arg\max_{\boldsymbol{\phi}\in\Phi} f(\boldsymbol{U}(t), \mathsf{RR}(\boldsymbol{\phi})), \quad \text{where} \quad (14)$$

$$f(\boldsymbol{U}(t), \mathsf{RR}(\boldsymbol{\phi}))$$
$$\triangleq \sum_{n:\phi_n=1}\left[U_n(t)\mathbb{E}\left[L_{1n}^{\boldsymbol{\phi}} - 1\right] - \mathbb{E}\left[L_{1n}^{\boldsymbol{\phi}}\right]\sum_{n=1}^{N}U_n(t)\lambda_n\right]$$

   and $\mathbb{E}\left[L_{1n}^{\boldsymbol{\phi}}\right] = 1 + \mathsf{P}_{n,01}^{(M(\boldsymbol{\phi}))}/\mathsf{P}_{n,10}$ from (8). Ties are broken arbitrarily.[7]
3) Run $\mathsf{RR}(\boldsymbol{\phi}(t))$ for one round of transmission. We emphasize that active channels in $\boldsymbol{\phi}$ are served in the least-recently-used order. After the round ends, go to Step 2.

The QRR policy is a frame-based algorithm similar to RandRR, except that at the beginning of every transmission round the policy selection is no longer random but based on a queue-dependent rule. We note that QRR is a polynomial time algorithm because we can compute $\boldsymbol{\phi}(t)$ in (14) in polynomial

[6]The vector $\boldsymbol{\phi}(t)$ is a queue-dependent decision and thus we should write $\boldsymbol{\phi}(\boldsymbol{U}(t), t)$ as a function of $\boldsymbol{U}(t)$. For simplicity we use $\boldsymbol{\phi}(t)$ instead.

[7]It can be shown that as long as the queue backlog vector $\boldsymbol{U}(t)$ is not identically zero and the arrival rate vector $\boldsymbol{\lambda}$ is interior to the inner capacity bound Λ_{int}, we always have $\max_{\boldsymbol{\phi}\in\Phi} f(\boldsymbol{U}(t), \mathsf{RR}(\boldsymbol{\phi})) > 0$.

time with the following divide and conquer approach:

1) Partition the set Φ into subsets $\{\Phi_1, \dots, \Phi_N\}$, where Φ_M, $M \in \{1, \dots, N\}$, is the set of N-dimensional binary vectors having exactly M entries be 1.
2) For each $M \in \{1, \dots, N\}$, find the maximizer of $f(\boldsymbol{U}(t), \mathsf{RR}(\boldsymbol{\phi}))$ among vectors in Φ_M. For each $\boldsymbol{\phi} \in \Phi_M$, we have
$$f(\boldsymbol{U}(t), \mathsf{RR}(\boldsymbol{\phi})) = \sum_{n:\phi_n=1} \left[U_n(t) \frac{\mathsf{P}_{n,01}^{(M)}}{\mathsf{P}_{n,10}} - \left(1 + \frac{\mathsf{P}_{n,01}^{(M)}}{\mathsf{P}_{n,10}}\right) \sum_{n=1}^{N} U_n(t)\lambda_n \right],$$
and the maximizer of $f(\boldsymbol{U}(t), \mathsf{RR}(\boldsymbol{\phi}))$ is to *activate* the M channels that yield the M largest summands of the above equation.
3) Obtain $\boldsymbol{\phi}(t)$ by comparing the maximizers from the above step for different values of M.

The detailed implementation is as follows.

Polynomial time implementation of Step 2 of QRR:
For each fixed $M \in \{1, \dots, N\}$, compute
$$U_n(t) \frac{\mathsf{P}_{n,01}^{(M)}}{\mathsf{P}_{n,10}} - \left(1 + \frac{\mathsf{P}_{n,01}^{(M)}}{\mathsf{P}_{n,10}}\right) \sum_{n=1}^{N} U_n(t)\lambda_n \qquad (15)$$
for all $n \in \{1, \dots, N\}$. Sort these N numbers and define the binary vector $\boldsymbol{\phi}^M = (\phi_1^M, \dots, \phi_N^M)$ such that $\phi_n^M = 1$ if the value (15) of channel n is among the M largest, otherwise $\phi_n^M = 0$. Ties are broken arbitrarily. Let $\hat{f}(\boldsymbol{U}(t), M)$ denote the sum of the M largest values of (15). Then define $M(t) \triangleq \arg\max_{1 \le M \le N} \hat{f}(\boldsymbol{U}(t), M)$ and assign $\boldsymbol{\phi}(t) = \boldsymbol{\phi}^{M(t)}$.

Using a novel variable-length frame-based Lyapunov analysis, we show in the next theorem that QRR stabilizes the network with any arrival rate vector $\boldsymbol{\lambda}$ strictly within the inner capacity bound Λ_{int}. The idea is that we compare QRR with the (unknown) policy RandRR* that stabilizes $\boldsymbol{\lambda}$. We show that, in every transmission round, QRR finds and executes a round robin policy RR$(\boldsymbol{\phi}(t))$ that yields a larger negative drift on the queue backlogs than RandRR* does in the current round. Therefore, QRR is stable.

Theorem 3. *For any data rate vector $\boldsymbol{\lambda}$ interior to Λ_{int}, policy* QRR *strongly stabilizes the network.*

Proof of Theorem 3: See [14]. ■

VII. Conclusion

The network capacity of a wireless network is practically degraded by communication overhead. In this paper, we take a step forward by studying the fundamental achievable rate region when communication overhead is kept minimum, that is, when channel probing is not permitted. While solving the original problem is difficult, we construct an inner and an outer bound on the network capacity region, with the aid of channel memory. When channels are symmetric and the network serves a large number of users, we show the inner and outer bound are progressively tight when the data rates of different users are more balanced. We also derive a simple queue-dependent frame-based policy and show that it stabilizes the network for any data rates strictly within the inner capacity bound.

Transmitting data without channel probing is one of the many options for communication over a wireless network. Practically each option may have pros and cons on criteria like the achievable throughput, power efficiency, implementation complexity, etc. Part of our future work is to explore how to combine all possible options to push the practically achievable network capacity to the limit. We also like to generalize the methodology and framework developed in this paper to more general cases, such as when limited probing is allowed and other QoS metrics such as energy consumption are considered. It will also be interesting to see how this framework can be applied to solve new problems in opportunistic spectrum access in cognitive radio networks, in opportunistic scheduling with delayed/uncertain CSI, and in restless bandit problems.

References

[1] C.-P. Li and M. J. Neely, "Energy-optimal scheduling with dynamic channel acquisition in wireless downlinks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 4, pp. 527 –539, Apr. 2010.

[2] H. S. Wang and P.-C. Chang, "On verifying the first-order markovian assumption for a rayleigh fading channel model," *IEEE Trans. Veh. Technol.*, vol. 45, no. 2, pp. 353–357, May 1996.

[3] M. Zorzi, R. R. Rao, and L. B. Milstein, "A Markov model for block errors on fading channels," in *Personal, Indoor and Mobile Radio Communications Symp. PIMRC*, Oct. 1996.

[4] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed. Athena Scientific, 2005, vol. I.

[5] Q. Zhao, B. Krishnamachari, and K. Liu, "On myopic sensing for multi-channel opportunistic access: Structure, optimality, and preformance," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 5431–5440, Dec. 2008.

[6] S. H. A. Ahmad, M. Liu, T. Javidi, Q. Zhao, and B. Krishnamachari, "Optimality of myopic sensing in multichannel opportunistic access," *IEEE Trans. Inf. Theory*, vol. 55, no. 9, pp. 4040–4050, Sep. 2009.

[7] M. J. Neely, "Stochastic optimization for markov modulated networks with application to delay constrained wireless scheduling," in *IEEE Conf. Decision and Control (CDC)*, 2009.

[8] Q. Zhao and A. Swami, "A decision-theoretic framework for opportunistic spectrum access," *IEEE Wireless Commun. Mag.*, vol. 14, no. 4, pp. 14–20, Aug. 2007.

[9] A. Pantelidou, A. Ephremides, and A. L. Tits, "Joint scheduling and routing for ad-hoc networks under channel state uncertainty," in *IEEE Int. Symp. Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, Apr. 2007.

[10] L. Ying and S. Shakkottai, "On throughput optimality with delayed network-state information," in *Information Theory and Application Workshop (ITA)*, 2008, pp. 339–344.

[11] ——, "Scheduling in mobile ad hoc networks with topology and channel-state uncertainty," in *IEEE INFOCOM*, Rio de Janeiro, Brazil, Apr. 2009.

[12] P. Chaporkar, A. Proutiere, H. Asnani, and A. Karandikar, "Scheduling with limited information in wireless systems," in *ACM Int. Symp. Mobile Ad Hoc Networking and Computing (MobiHoc)*, New Orleans, LA, May 2009.

[13] R. G. Gallager, *Discrete Stochastic Processes*. Kluwer Academic Publishers, 1996.

[14] C.-P. Li and M. J. Neely, "Exploiting channel memory for multi-user wireless scheduling without channel measurement: Capacity regions and algorithms," Tech. Rep., Apr. 2010. [Online]. Available: http://arxiv.org/abs/1003.2675

[15] P. Whittle, "Restless bandits: Activity allocation in a changing world," *J. Appl. Probab.*, vol. 25, pp. 287–298, 1988.

[16] J. C. Gittins, *Multi-Armed Bandit Allocation Indices*. New York, NY: Wiley, 1989.

[17] L. Tassiulas and A. Ephremides, "Dynamic server allocation to parallel queues with random varying connectivity," *IEEE Trans. Inf. Theory*, vol. 39, no. 2, pp. 466–478, Mar. 1993.

[18] L. Georgiadis, M. J. Neely, and L. Tassiulas, "Resource allocation and cross-layer control in wireless networks," *Foundations and Trends in Networking*, vol. 1, no. 1, 2006.