Network Utility Maximization over Partially Observable Markovian Channels

Chih-ping Li and Michael J. Neely

***Abstract*—We study throughput utility maximization in a multi-user network with partially observable Markovian channels. Here, instantaneous channel states are unavailable and all controls are based on partial channel information provided by ACK/NACK feedback from past transmissions. Equivalently, we formulate a restless multi-armed bandit problem in which we seek to maximize concave functions of the time average reward vector. Such problems are generally intractable and in our problem the set of all achievable throughput vectors is unknown. We use an achievable region approach by optimizing the utility functions over a non-trivial reduced throughput region, constructed by randomizing well-designed round robin policies. Using a new ratio MaxWeight rule, we design admission control and channel scheduling policies that stabilize the network with throughput utility that is near-optimal within the reduced throughput region. The ratio MaxWeight rule generalizes existing MaxWeight-type policies for the optimization of frame-based control systems with policy-dependent frame sizes. Our results are applicable to limited channel probing in wireless networks, dynamic spectrum access in cognitive radio networks, and target tracking of unmanned aerial vehicles.**

***Index Terms*—stochastic network optimization, Markovian channels, restless multi-armed bandit, cognitive radio, opportunistic spectrum access, achievable region approach, Lyapunov drift analysis, ratio max-weight policy**

I. Introduction

We study a multi-user wireless scheduling problem in a partially observable environment. Consider a base station serving N users with independent Markov ON/OFF channels (see Fig. 1). Time is slotted with normalized slots $t \in \{0, 1, 2, \ldots\}$.

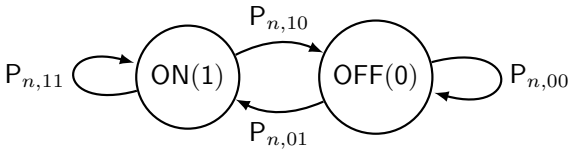

Fig. 1. The Markov ON/OFF channel for user $n \in \{1, 2, \ldots, N\}$.

Channel states are assumed to be fixed in every slot, but may change over time. The base station has unlimited data to send for each user. In every slot, the instantaneous channel states are unknown and the base station may blindly send a packet to a selected user. The transmission succeeds if the channel state is ON, and fails otherwise.[1] At the end of a slot, the served user sends an ACK/NACK feedback to the base station. Since the channels have memory, the feedback provides partial information of future channel states, and can be used to improve future scheduling decisions for better performance. Our goal is to design a network control policy that optimizes the throughput vector over the partially observable channels.

Specifically, let $\overline{y}_n$ be the throughput of user n under a given policy. Let $g_n(\cdot)$ be the utility function of user n, where $g_n(\overline{y}_n)$ represents the satisfaction of user n with throughput $\overline{y}_n$. We assume the functions $g_n(\cdot)$ are concave, continuous, nondecreasing, and differentiable. Examples of the utility functions are weighted throughput $g_n(\overline{y}_n) = c_n\,\overline{y}_n$, proportional fairness [2] $g_n(\overline{y}_n) = \log(\overline{y}_n)$, or the more general α-fairness functions [3] $g_n(\overline{y}_n) = (\overline{y}_n)^{1-\alpha}/(1-\alpha)$, $\alpha \geq 0$. Let Λ be the network capacity region, defined as the closure of the set of all achievable throughput vectors in the network. We consider the utility maximization problem:

$$\text{maximize} \quad \sum_{n=1}^{N} g_n(\overline{y}_n), \quad \text{subject to} \quad (\overline{y}_n) \in \Lambda. \qquad (1)$$

We have two types of control in the network: a flow controller and a channel scheduling policy. We assume the unlimited data at the base station is stored in an upper-layer reservoir. The flow controller admits packets from the reservoir into network-layer queues for transmission in every slot. Flow control is used to differentiate the throughput received by all users so as to maximize the long-term total utility $\sum_{n=1}^{N} g_n(\overline{y}_n)$. The channel scheduling policy serves a user in a slot, basing its decisions on the belief of channel states and on the backlogs of the network-layered queues, so as to stabilize the network.

Before introducing our approach to attack the problem (1), we discuss potential applications of the above model. It is widely known that instantaneous channel state information in wireless networks helps to achieve throughput optimality (e.g., [4]–[6]). Yet, such information may be unavailable in a fast-changing environment in which channel sensing is limited or the channel feedback is delayed. The problem we consider in this paper studies the use of time correlations of wireless channels to improve throughput, and further investigates how to perform utility maximization in this context. In cognitive

Chih-ping Li (email: cpli@mit.edu) is with the Laboratory for Information and Decision Systems, MIT, Cambridge, MA 02139, USA. Michael J. Neely (web: http://www-bcf.usc.edu/∼mjneely) is with the Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089, USA.

This work was performed when the first author was with the Department of Electrical Engineering, University of Southern California. It 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, NSF grant 0964479, the Network Science Collaborative Technology Alliance sponsored by the U.S. Army Research Laboratory W911NF-09-2-0053.

This work was presented in part at the WiOpt 2011 conference, Princeton, NJ, USA, 2011 [1].

[1] Equivalently, we may assume the base station can probe a single channel every slot, and transmits a packet if the state is ON.

radio networks [7], [8], Markov ON/OFF channels are used to model the idleness of radio spectrums exclusively assigned to primary users. The problem stated above studies how a secondary user can opportunistically transmit data over temporarily unused channels to improve spectrum efficiency. In target tracking of unmanned aerial vehicles (UAVs) [9], a UAV monitors one of multiple targets in every slot. The movement of a target can be modeled as a Markov channel; the channel is ON if the target moves to a hotspot, and OFF otherwise. Detecting a target at its hot spot yields a reward; this is the same as delivering a packet over a channel. Tracking all targets with some notion of fairness is an interesting problem captured by the optimization problem (1).

The problem (1) is difficult because it belongs to the class of restless multi-armed bandit (RMAB) problems [10], which are known to be generally intractable [11]. Here, every channel is viewed as a bandit, whose state in a slot is the conditional probability (or the belief) that the channel is at state ON, given the history of past channel observations. These bandits (channels) are *restless* because the belief of channel states changes over time even when the channels are not used in a slot. From the view of optimization, the problem (1) is difficult because the feasible region Λ does not seem to have a closed form expression; every boundary point of Λ can be viewed as the solution to a RMAB problem with some linear cost function. Therefore, it seems impossible to solve (1) exactly.

A. Achievable region approach

Because the problem (1) is seemingly intractable, we resort to an achievable region approach. That is, we construct a convex throughput region Λ_{int} that is a subset of the network capacity region Λ, and solve the convex optimization problem

$$\text{maximize} \quad \sum_{n=1}^{N} g_n(\overline{y}_n), \quad \text{subject to} \quad (\overline{y}_n) \in \Lambda_{\text{int}}. \qquad (2)$$

Solving (2) only provides a suboptimal solution to the original problem (1). Nonetheless, in practice a reduced but easily achievable throughput region Λ_{int} could be of greater interest than exploring the full throughput region Λ; achieving a throughput vector outside the region Λ_{int} may inevitably involve solving a high-dimensional (partially observable) Markov decision process (MDP), which is less desired in practice. We may regard the reduced throughput region Λ_{int} as an *operational* network capacity region, which may be improved by a deeper investigation into the problem structure. Also, as opposed to previous studies on RMAB problems mostly with linear cost functions, convex optimization over restless bandits itself is an interesting problem even with a reduced throughput region.

In [12], we have developed a non-trivial reduced throughput region Λ_{int} using the rich structure of Markovian channels. The region Λ_{int} is constructed as a convex hull of performance vectors of a collection of well-designed round robin policies. These policies take advantage of channel memory to improve throughput. The tightness of Λ_{int} is analyzed in the special case that channels are independent and statistically identical. The tightness of the region Λ_{int} is difficult to check in general cases, but the region is intuitively large due to the nature of its construction (see more details in Section III). We show examples in Section VI that, under linear cost functions, our network control policy resulting from optimizing over the reduced throughput region Λ_{int} has similar performance as Whittle's index [13].

B. Ratio MaxWeight policies

Given the reduced throughput region Λ_{int}, the main contribution of this paper is to develop novel *ratio MaxWeight* policies that, together with simple admission control, solve (2). In our policy, the amount of data admitted in every slot is the solution to a simple convex program. For channel scheduling, we divide time into frames and serve a subset of users (chosen by ratio MaxWeight) in every frame in a round robin fashion with proper dynamic orderings. Our control policy yields a throughput utility that can be made arbitrarily close to the optimal utility *in the reduced throughput region* Λ_{int}, at the expense of increasing average queue sizes (see Section V).

In a broader context, the ratio MaxWeight rule generalizes the existing Lyapunov optimization framework [14] and enables us to solve convex optimization problems over frame-based/renewal systems that have policy-dependent frame sizes. To convey the basic idea, let us consider a network control system in which we can run a set of stationary (possibly random) policies $\Pi = \{\pi_1, \pi_2, \ldots, \pi_M\}$ on a frame-by-frame basis. That is, we divide time into frames and employ a policy in Π in each frame; we only allow one policy per frame. In a renewal system a frame can be the duration of a renewal period. In a Markov decision problem a frame may be the time between two visits to a recurrent system state. Let r_m be the average reward vector when policy π_m is run in all frames, and we have

$$r_m = \frac{\mathbb{E}\left[\text{total reward under } \pi_m \text{ in a frame of size } T_m\right]}{\mathbb{E}\left[T_m\right]}. \qquad (3)$$

In renewal systems, the above equation is a standard renewal reward theory result [15]. The distribution of the random frame size T_m may depend on the choice of π_m. By the time sharing of policies in Π, the resulting performance region, denoted by R, is the convex hull of the vectors $\{r_1, \ldots, r_M\}$. Through Lyapunov analysis, we can solve a convex optimization of the form (2) over R, possibly with additional linear constraints, by dynamically choosing a policy in Π to be executed in each frame. The policy chosen in frame k is the one that maximizes the *ratio MaxWeight sum* (cf. (3))

$$\text{maximize} \quad \sum_{n=1}^{N} Z_{nk}\, r_k, \quad \text{subject to } (r_k) \in R. \qquad (4)$$

The term Z_{nk} is the (virtual) queue backlog observed at the beginning of frame k, constructed to capture the running performance of user n. Note that the linear program (4) is solved by a vertex of the polytope R, and therefore is solved by a policy in Π. The complexity of solving (4) depends on the context of the problem.

In this paper, although our problem does not have an obvious renewal structure, we carefully design our round robin

policies so that, under these policies, the throughput achieved in a round of round robin is independent of the system history prior to this round; thus a renewal property is enforced. Our policy space Π consists of these round robin policies.

The recent applications of the ratio MaxWeight rule are as follows. Work [16] [17, Chapter 5] shows constrained convex optimization over a multi-class single-server queue that has a polymatroid delay region is surprisingly solved by a dynamic $c\mu$ rule [18], which chooses a strict priority policy in every busy period. The optimality of the $c\mu$ rule is mostly established in the past for linear cost functions (e.g., [18], [19]). One example for solving a convex optimization in a single-server queue is to provide fairness for the average delay experienced by different classes of traffic. Work [20] studies opportunistic cooperation between primary and second users in a cognitive network, which is a constrained MDP problem; the ratio MaxWeight rule here reduces to solving an average cost MDP problem that can be solved exactly. The ratio MaxWeight rule is also applied to dynamic index coding problems in wireless networks [21] and energy-aware sleep scheduling in a multi-server system [22]. We refer readers to [23, Chapter 7] for more discussions on using the ratio MaxWeight rule to optimize renewal systems.

C. Related work

Stochastic utility maximization over wireless networks that assumes perfect knowledge of instantaneous channel states is studied in [24]–[27]; see also [28]–[30]. Limited channel probing in wireless networks is studied in [31]–[38], assuming i.i.d. channel states over time. Work [39], [40] develops throughput-optimal policies in networks with delayed/infrequent channel state information over Markovian channels; it is assumed there that channel probing is given as part of the system model and is not to be controlled. In other words, channel probing and network control are decoupled. Under such assumption, the network capacity region can be fully characterized in terms of the expected allowable transmission rates conditioning on the most recent channel observations. The major difference of this paper from [39], [40] is that we assume channel probing is part of the network control actions, which completely changes the nature of the problem.

RMAB problems with Markov ON/OFF states are studied in [13], [41]–[48] for the maximization of sum of time average or discounted rewards; see also [49]. Index policies such Whittle's index [10] are constructed in [13], [44], [49] with good numerical performance, and are shown to have asymptotically optimal properties in [48], [50]. A $(2+\epsilon)$-approximate algorithm is derived in [45] based on duality methods. In particular, work [41]–[43] shows that myopic round robin policies maximize the sum throughput over Markovian channels in some special cases; in [12], we modify these policies to construct the throughput region Λ_{int} used in this paper. In general, most previous studies on RMAB problems focus on linear cost functions. Our analysis in this paper is quite different because we consider concave objective functions and use an achievable region method. Index heuristics developed for linear costs seem difficult to apply to our problem because they use dynamic programming ideas.

Previous work [46] characterizes the full network capacity region Λ over partially observable Markov ON/OFF channels as a limit of a sequence of linear programs, each of which solves a finite-state MDP truncated from an infinite-state MDP that can describe Λ. Due to the curse of dimensionality, this approach does not scale with the number of channels. Our reduced throughput region Λ_{int}, although a strict subset of the full network capacity region Λ in usual cases, is designed to scale well with the number of channels. As a result, our network control policy also scales with the number of channels (see Section IV).

An outline of the paper is as follows. The network model is given in the next section. Section III introduces the performance region Λ_{int} constructed in [12]. Our dynamic control policy is motivated and given in Section IV, followed by performance analysis in Section V. The simulation results, including the performance comparison to Whittle's index, are presented in Section VI.

II. Detailed Network Model

In addition to the network model described in the introduction, we suppose that each Markov ON/OFF channel $n \in \{1, \ldots, N\}$ changes states across slots according to 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 $0 < \mathsf{P}_{n,ij} < 1$ for all states i, j and all channels n, and that the probability matrices $\mathbf{P}_n$ are known at the base station. We assume each channel is positively correlated over time. An equivalent mathematical assumption is to let $\mathsf{P}_{n,01}+\mathsf{P}_{n,10} < 1$ (equivalently, $\mathsf{P}_{n,11} > \mathsf{P}_{n,01}$) for all channels n.[2] For channel n, let $\mathsf{P}_{n,ij}^{(k)}$ be the k-step transition probability from state i to j, and $\pi_{n,\mathsf{ON}}$ be the stationary probability of state ON.

We assume that the base station has a higher-layer unlimited data source for each user. In every slot, the base station admits $r_n(t)$ user-n packets into a network-layer queue $Q_n(t)$ of infinite capacity, where $Q_n(t)$ denotes the user-n backlog in slot t. For simplicity, we assume $r_n(t)$ takes real values in the interval $[0, 1]$ for all n.[3] Let $\mu_n(t) \in \{0, 1\}$ be the service rate for user n in slot t under a given policy; we have $\mu_n(t) = 1$ if user n is served in slot t and its channel is ON, and 0 otherwise. The user-n queue $\{Q_n(t)\}_{t=0}^{\infty}$ evolves over time as

$$Q_n(t+1) = \max[Q_n(t) - \mu_n(t), 0] + r_n(t). \qquad (5)$$

[2]With $\mathsf{P}_{n,11} > \mathsf{P}_{n,01}$, we let $s_n(t)$ be the state of channel n in a slot and observe that the auto-covariance of $s_n(t)$ is positive according to

$$\begin{aligned} &\mathbb{E}\left[s_n(t)s_n(t+1)\right] - \mathbb{E}\left[s_n(t)\right]\mathbb{E}\left[s_n(t+1)\right] \\ &= \theta\,\mathsf{P}_{n,11} - \theta\left[(1-\theta)\mathsf{P}_{n,01} + \theta\,\mathsf{P}_{n,11}\right] \\ &> \theta\,\mathsf{P}_{n,11} - \theta\,\mathsf{P}_{n,11} = 0, \end{aligned}$$

where $\theta = \Pr\left[s_n(t) = 1\right]$.

[3]We can think of $r_n(t)$ as the number of packets admitted in a slot, normalized by the maximal number of packets that can be admitted in every slot; thus $r_n(t)$ can take real values between 0 and 1. We can accommodate the alternative assumption that $r_n(t)$ takes integer values in $\{0, 1\}$ by introducing *auxiliary queues*; see [25] for an example.

Initially, we assume $Q_n(0) = 0$ for all n. We say queue $Q_n(t)$ is (strongly) stable if its limiting average backlog is finite, i.e.,

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

The network is stable if all queues $(Q_1(t), \ldots, Q_N(t))$ are stable. Clearly a sufficient condition for stability is:

$$\limsup_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1}\sum_{n=1}^{N} \mathbb{E}\left[Q_n(\tau)\right] < \infty. \tag{6}$$

Our goal is to design a control policy that admits the right amount of data into the network and serves them properly by channel scheduling, so that the network is stable with a throughput utility that (almost) solves the optimization problem (2).

III. Reduced Throughput Region

This section introduces the reduced throughput region Λ_{int} constructed by randomizing (or time sharing) over a collection of round robin policies that exploit channel memory. For this, we first study the structure of positively correlated Markovian channels and how the belief of the channel states evolves in the system.

A. The structure of Markovian channels

We define the information state (i.e., the belief) $\omega_n(t)$ of channel n as the conditional probability that channel n is ON in slot t given all past channel observations. Namely,

$$\omega_n(t) \triangleq \Pr\left[s_n(t) = \mathsf{ON} \mid \text{channel observation history}\right],$$

where $s_n(t)$ denotes the state of channel n in slot t. We assume initially $\omega_n(0) = \pi_{n,\mathsf{ON}}$ for all n.[4] Conditioning on the most recent observation, the belief $\omega_n(t)$ takes values in the countably infinite set $\mathcal{W}_n \triangleq \{\mathsf{P}_{n,01}^{(k)}, \mathsf{P}_{n,11}^{(k)} : k \in \mathbb{N}\} \cup \{\pi_{n,\mathsf{ON}}\}$. If channel n is last observed k slots in the past and its state was $i \in \{0,1\}$, the current belief in slot t is $\omega_n(t) = \mathsf{P}_{n,i1}^{(k)}$. Let $n(t)$ be the channel observed in slot t. Then the belief $\omega_n(t)$ evolves over time as:

$$\omega_n(t+1) = \begin{cases} \mathsf{P}_{n,01}, & \text{if } n = n(t),\ s_n(t) = \mathsf{OFF} \\ \mathsf{P}_{n,11}, & \text{if } n = n(t),\ s_n(t) = \mathsf{ON} \\ \omega_n(t)\mathsf{P}_{n,11} + (1-\omega_n(t))\mathsf{P}_{n,01}, & \text{if } n \neq n(t). \end{cases} \tag{7}$$

The belief vector $(\omega_n(t))_{n=1}^N$ is a sufficient statistic [51]; i.e., it is optimal to schedule channels based only on the $(\omega_n(t))_{n=1}^N$ information.

B. Policies that exploit channel memory

Since channels are positively correlated over time, we have shown in [12] that the feasible values of $\omega_n(t)$ satisfy

$$\mathsf{P}_{n,11} \geq \mathsf{P}_{n,11}^{(k_1)} \geq \mathsf{P}_{n,11}^{(k_2)} \geq \pi_{n,\mathsf{ON}} \geq \mathsf{P}_{n,01}^{(k_3)} \geq \mathsf{P}_{n,01}^{(k_4)} \geq \mathsf{P}_{n,01} \tag{8}$$

for all integers $k_1 \leq k_2$ and $k_3 \geq k_4$. In particular, $\mathsf{P}_{n,01}^{(k)}$ is increasing in k and $\mathsf{P}_{n,11}^{(k)}$ is decreasing in k. This inequality has important implications. The belief $\omega_n(t)$ is the expected throughput for user n in slot t. Thus, to have better throughput, we should keep serving a channel whenever $\omega_n(t)$ has the maximal value $\mathsf{P}_{n,11}$ (i.e., when the channel n is known to be ON in the previous slot). On the other hand, given channel n was OFF in its last use, we should idle the channel as long as possible so that its belief $\mathsf{P}_{n,01}^{(k)}$ can improve over time (noting that $\mathsf{P}_{n,01}^{(k)}$ is increasing in k). One good policy that exploits the above two throughput-improving properties is the following class of myopic round robin policies:

> Fix a subset A of channels in $\{1, \ldots, N\}$. We serve the channels in a nonstop round robin fashion with a fixed channel ordering, where on each channel we keep transmitting packets until a NACK is received (i.e., when the channel turns OFF).

The use of round robin is indeed to give the last-accessed channel the most time to "recover" from the worst belief state $\mathsf{P}_{n,01}$. This myopic round robin policy when applied to all channels is known to maximize the sum throughput when channels are positively correlated, independent, and statistically identical [42].

Since the myopic round robin policies exploit wireless channel memory, we may construct an achievable throughput region Λ_{int} by randomizing over these policies. The difficulty is that the throughput of a myopic round robin policy is hard to analyze because it is directly related to a high-order Markov chain [41]. Thus, the resulting throughput region is also hard to obtain. Our approach is to develop a modified round robin policy whose performance is easy to analyze. Randomizing these policies yields a well-defined throughput region Λ_{int}, so that the optimization problem (2) can be solved. This modified policy is more conservative than myopic round robin because it allows transmitting dummy packets; this is for the sake of tractability. Nonetheless, the modified round robin policy still exploits channel memory to improve throughput, and is asymptotically optimal in some special cases such as when channels are i.i.d. [12]. The development of the modified round robin policies and their randomizations, together with the associated throughput region, is studied in [12]; we summarize these results in the rest of the section to facilitate later analysis.

C. Randomized round robin

The randomized round robin policy randomly serves a subset of users for one round, after which it selects another subset of users, and so on. The order in which users are served in a round is important. Suppose we choose to serve user 1 and 2 in a round, where user 1 is the last served user in the previous round. Recall from the above discussions that we need to give the last-accessed channel the most time to recover from a bad belief state. Thus, we should serve user 2 before user 1 in this round. In general, we shall serve users in every round with the ordering of *least recently used first*.

We describe the randomized round robin policies. Let Φ be the set of all nonzero N-dimensional binary vectors. Each vector $\boldsymbol{\phi} \triangleq (\phi_n)_{n=1}^N \in \Phi$ denotes a collection of *active*

[4] In general, we need that the initial belief $\omega_n(0)$ belongs to the set $\{\mathsf{P}_{n,11}^{(k)} : k \in \mathbb{N}\} \cup \{\pi_{n,\mathsf{ON}}\}$. This can be achieved by having a training period before time zero in which every user is continuously served until an ACK is received.

channels; we say channel n is active if $\phi_n = 1$. Let $M(\boldsymbol{\phi})$ be the number of active channels (ones) in $\boldsymbol{\phi}$. Later in the paper we also allow $\boldsymbol{\phi} = \mathbf{0}$, meaning that no channels are chosen and the system is forced to idle for one slot.

Randomized Round Robin Policy (RandRR)

1) In every round, pick a subset of active channels $\boldsymbol{\phi} \in \Phi \cup \{\mathbf{0}\}$ with probability $\alpha_{\boldsymbol{\phi}}$, where $\{\alpha_{\boldsymbol{\phi}}\}$ is a stationary distribution with $\alpha_{\mathbf{0}} + \sum_{\boldsymbol{\phi}\in\Phi} \alpha_{\boldsymbol{\phi}} = 1$.
2) If $\boldsymbol{\phi} \in \Phi$ is selected, serve active channels in $\boldsymbol{\phi}$ for one round using the ordering of *least recently used first* (in the first round, the ordering is arbitrary). The service on each channel is described in Step 4. If $\boldsymbol{\phi} = \mathbf{0}$, we idle the system for one slot. At the end of either case, go to Step 1.
3) Update the belief vector $(\omega_n(t))_{n=1}^N$ by (7) in every slot.
4) When starting service on channel n, with probability $\mathsf{P}_{n,01}^{(M(\boldsymbol{\phi}))}/\omega_n(t)$ we keep transmitting packets until a NACK is received. With probability $1 - \mathsf{P}_{n,01}^{(M(\boldsymbol{\phi}))}/\omega_n(t)$ we transmit a dummy packet with no information content for one slot. After either case, switch to the next active channel.

Step 4 of the RandRR policy ensures that, when we start serving a channel n in a round in which M channels are chosen, we "fake" the current belief of channel n to be $\mathsf{P}_{n,01}^{(M)}$, which can be strictly worse than its actual belief $\omega_n(t)$ (again, the reason of doing this is to design policies with tractable performance). To see this, in Step 4, the probability that user n successfully transmits a packet in the first slot (and subsequently keeps transmitting until receiving a NACK) is

$$\frac{\mathsf{P}_{n,01}^{(M)}}{\omega_n(t)} \times \omega_n(t) = \mathsf{P}_{n,01}^{(M)}.$$

Also, we switch to the next channel after the first slot with no data packets delivered if either a dummy packet is transmitted or a data packet is served but the channel is OFF. The event happens with probability

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

We see that transmitting a dummy packet is to "fake" an OFF channel state. We note that allowing system idling in a RandRR policy does not degrade the set of achievable throughput vectors. If we let R be the set of feasible throughput vectors under the set of RandRR policies *in which idling the system is prohibited*, permitting system idling simply ensures that any throughput vector dominated entrywise by a point in R can be achieved by a RandRR policy. This is useful in later analysis.

A RandRR policy is feasible only if the inequality $\mathsf{P}_{n,01}^{(M(\boldsymbol{\phi}))} \le \omega_n(t)$ holds whenever channel n starts service. This can be proved by a similar argument as in [12, Lemma 6]; we provide the proof below for completeness.

Lemma 1. Every RandRR policy is feasible.

Proof: When a channel starts service in the first round, by assumption we have $\omega_n(t) = \pi_{n,\mathsf{ON}} \ge \mathsf{P}_{n,01}^{(k)}$ for all $k \in \{1,\dots,N\}$ (see (8)). Thus every RandRR policy is feasible in the first round. Suppose M channels are selected for service in a round after the first. We index the M channels by $\{n_1,\dots,n_M\}$, which is in the decreasing order of the time duration between their last use and the beginning of the current round. In other words, the last use of n_k is earlier than that of $n_{k'}$ only if $k < k'$. Fix a channel n_k. This lemma is proved if we can show when channel n_k starts service, say on slot t, the time elapsed since the end of its last service is at least $(M-1)$ slots. To see this, assuming this condition holds, we have two cases:

- If channel n_k is known to be ON at its last use (when a dummy packet is transmitted), then its belief in slot t is $w_{n_k}(t) = \mathsf{P}_{n_k,11}^{(m)}$ for some $m \ge M$, which is greater than or equal to $\mathsf{P}_{n_k,01}^{(M(\boldsymbol{\phi}))} = \mathsf{P}_{n_k,01}^{(M)}$ by (8).
- If channel n_k is OFF at its last use, then $\omega_{n_k}(t) = \mathsf{P}_{n_k,01}^{(m)}$ for some $m \ge M$. Thus $\omega_{n_k}(t) \ge \mathsf{P}_{n_k,01}^{(M)}$ because $\mathsf{P}_{n_k,01}^{(m)}$ is increasing in m (cf. (8)).

It remains to show that when channel n_k starts service, the time elapsed since the end of its last service is at least $(M-1)$ slots. We partition the M channels except for n_k into $A = \{n_1,\dots,n_{k-1}\}$ and $B = \{n_{k+1},\dots,n_M\}$. The last use of every channel in B occurs after the last use of n_k, and thus channel n_k has been idled for at least $|B|$ slots. However, the policy in this round will serve all channels in A before serving n_k from the ordering of least recently used first. Each channel in A takes at least one slot, and so we wait at least additional $|A|$ slots before serving channel n_k. The total time that channel n_k has been idled since its last use is thus at least $|A| + |B| = (M-1)$ slots. ∎

We give a simple example of using a RandRR policy. Suppose channel subsets $\{1,2,3,4\}$, $\{2,3\}$, $\{1,2,4\}$ are selected in the first three rounds. In the first round, channels are ordered by $4 \to 1 \to 3 \to 2$. The complete ordering by least recently used first in the first three rounds is then

$$(4 \to 1 \to 3 \to 2) \to (3 \to 2) \to (4 \to 1 \to 2).$$

Here are some useful properties of the RandRR policies.

Theorem 1. 1) In a round of a RandRR policy in which the active channels $\boldsymbol{\phi} \in \Phi$ are chosen for service, we let $L_n^{\boldsymbol{\phi}}$ be the time duration an active channel n is accessed. The random variable $L_n^{\boldsymbol{\phi}}$ has the probability distribution:

$$L_n^{\boldsymbol{\phi}} = \begin{cases} 1 & \text{with prob. } 1 - \mathsf{P}_{n,01}^{(M(\boldsymbol{\phi}))} \\ j \ge 2 & \text{with prob. } \mathsf{P}_{n,01}^{(M(\boldsymbol{\phi}))}\,(\mathsf{P}_{n,11})^{(j-2)}\,\mathsf{P}_{n,10} \end{cases} \tag{9}$$

and

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

2) During $L_n^{\boldsymbol{\phi}}$, channel n serves $(L_n^{\boldsymbol{\phi}} - 1)$ packets.

3) The random variables $\{L_n^{\boldsymbol{\phi}}\}_{n:\phi_n=1}$ are mutually independent.

Proof of Theorem 1: The first two results are given in [12,

Corollary 1]. For the last result, it is not difficult to see that

$$\Pr\left[L_n^{\boldsymbol{\phi}} = j | \omega_n(t)\right] = \Pr\left[L_n^{\boldsymbol{\phi}} = j\right], \quad j = 1, 2, \ldots \tag{11}$$

for all possible values of $\omega_n(t)$, where $\omega_n(t)$ denotes the belief of channel n when channel n starts service, and $\Pr\left[L_n^{\boldsymbol{\phi}} = j\right]$ is given in (9). Equation (11) shows that the value of $L_n^{\boldsymbol{\phi}}$ is independent of the belief $\omega_n(t)$. Since $\omega_n(t)$ summarizes the system history, the value of $L_n^{\boldsymbol{\phi}}$ is indeed independent of the system history including the values of $L_m^{\boldsymbol{\phi}}$ for all active channels m served before channel n. Therefore we must have

$$\Pr\left[L_n^{\boldsymbol{\phi}} = j | L_m^{\boldsymbol{\phi}} = i\right] = \Pr\left[L_n^{\boldsymbol{\phi}} = j\right], \quad j = 1, 2, \ldots$$

for all active channels m in $\boldsymbol{\phi}$ receiving service before channel n in the same round. We conclude that the random variables $\{L_n^{\boldsymbol{\phi}}\}_{n:\phi_n=1}$ are mutually independent. ■

Let T_k denote the duration of the kth round in a RandRR policy, and $\boldsymbol{\phi}_k$ represent the active channels served in T_k. An important observation from the construction of the RandRR policy is that the random variables $\{T_k, k = 0, 1, 2, \ldots\}$ are i.i.d.. Specifically, let $\boldsymbol{\omega}_k$ be the belief state vector at the start of T_k. Given that the active channels $\boldsymbol{\phi}$ are served in T_k, we have

$$\begin{aligned}\Pr\left[T_k = j \mid \boldsymbol{\omega}_k, \boldsymbol{\phi}_k = \boldsymbol{\phi}\right] &= \Pr\Big[\sum_{n:\phi_n=1} L_n^{\boldsymbol{\phi}} = j \mid \boldsymbol{\omega}_k\Big] \\ &= \Pr\Big[\sum_{n:\phi_n=1} L_n^{\boldsymbol{\phi}} = j\Big]\end{aligned}$$

where the last equality follows that the value of $L_n^{\boldsymbol{\phi}}$ is independent of the system history. When no channels are served in T_k (i.e., the system is idle), we always have $T_k = 1$. It follows that

$$\begin{aligned}\Pr\left[T_k = j \mid \boldsymbol{\omega}_k\right] &= \sum_{\boldsymbol{\phi}\in\Phi\cup\{\mathbf{0}\}} \alpha_{\boldsymbol{\phi}} \Pr\left[T_k = j \mid \boldsymbol{\omega}_k, \boldsymbol{\phi}_k = \boldsymbol{\phi}\right] \\ &= \alpha_{\mathbf{0}} 1_{[j=1]} + \sum_{\boldsymbol{\phi}\in\Phi} \alpha_{\boldsymbol{\phi}} \Pr\Big[\sum_{n:\phi_n=1} L_n^{\boldsymbol{\phi}} = j\Big],\end{aligned} \tag{12}$$

which holds for all possible values of $\boldsymbol{\omega}_k$, and $1_{[\cdot]}$ is an indicator function. From (12), the value of T_k is independent of the system history, indicating that T_k are independent over k. From (12) we also have

$$\Pr\left[T_k = j\right] = \alpha_{\mathbf{0}} 1_{[j=1]} + \sum_{\boldsymbol{\phi}\in\Phi} \alpha_{\boldsymbol{\phi}} \Pr\Big[\sum_{n:\phi_n=1} L_n^{\boldsymbol{\phi}} = j\Big], \tag{13}$$

which holds for all k; thus T_k are identically distributed. From these discussions we have the next lemma.

Lemma 2. 1) Let T_k denote the duration of the kth transmission round in a RandRR policy. The random variables T_k are i.i.d. over k with

$$\mathbb{E}\left[T_k\right] = \alpha_{\mathbf{0}} + \sum_{\boldsymbol{\phi}\in\Phi} \alpha_{\boldsymbol{\phi}} \Big(\sum_{n:\phi_n=1} \mathbb{E}\left[L_n^{\boldsymbol{\phi}}\right] \Big).$$

2) Let $N_{n,k}$ denote the number of packets served for user n in round T_k. For each user n, the random variables $N_{n,k}$ are i.i.d. over k with $\mathbb{E}\left[N_{n,k}\right] = \sum_{\boldsymbol{\phi}:\phi_n=1} \alpha_{\boldsymbol{\phi}} \mathbb{E}\left[L_n^{\boldsymbol{\phi}} - 1\right]$.

3) Because $N_{n,k}$ and T_k are i.i.d. over k, the throughput for user n under a RandRR policy is equal to $\mathbb{E}\left[N_{n,k}\right] / \mathbb{E}\left[T_k\right]$.

Proof: That T_k are i.i.d. is shown above. The value of $\mathbb{E}\left[T_k\right]$ follows (13). That $N_{n,k}$ are i.i.d. over k follows directly that T_k are i.i.d.. The value of $\mathbb{E}\left[N_{n,k}\right]$ follows the first result and Theorem 1. The last result follows the first two results and the Law of Large Numbers. ■

D. The reduced throughput region Λ_{int}

We define the throughput region Λ_{int} in the optimization problem (2) as the set of all throughput vectors achieved by the collection of RandRR policies. A closed form expression of Λ_{int} is analyzed in [12, Theorem 1] and given next.

Theorem 2. For each nonzero binary vector $\boldsymbol{\phi} \in \Phi$, define an N-dimensional vector $\boldsymbol{\eta}^{\boldsymbol{\phi}} = (\eta_n^{\boldsymbol{\phi}})$ where

$$\eta_n^{\boldsymbol{\phi}} = \begin{cases} \frac{\mathbb{E}\left[L_n^{\boldsymbol{\phi}}\right]-1}{\sum_{n:\phi_n=1} \mathbb{E}\left[L_n^{\boldsymbol{\phi}}\right]} & \text{if } \phi_n = 1 \\ 0 & \text{otherwise,} \end{cases}$$

where $\mathbb{E}\left[L_n^{\boldsymbol{\phi}}\right]$ is defined in (10). The throughput region Λ_{int} rendered by the class of RandRR policies is

$$\Lambda_{\text{int}} = \left\{ (\lambda_n) \;\middle|\; 0 \le \lambda_n \le \mu_n \;\forall n,\; (\mu_n) \in \operatorname{conv}\left(\{\boldsymbol{\eta}^{\boldsymbol{\phi}}\}_{\boldsymbol{\phi}\in\Phi}\right) \right\},$$

where $\operatorname{conv}(A)$ denotes the convex hull of set A.

The tightness of the region Λ_{int} in Theorem 2 is quantified in [12, Section V] when channels are i.i.d.; in particular, it is shown that the gap between the boundary of the region Λ_{int} and that of the full network capacity region Λ in a feasible direction d from the origin decreases to zero exponentially fast as we move d to form a smaller angle with the all-one vector $(1, \ldots, 1)$.

E. A two-user example

In a two-user network with i.i.d. channels with $\mathsf{P}_{01} = \mathsf{P}_{10} = 0.2$ (the subscript n is dropped due to channel symmetry), Fig. 2 gives an idea of the tightness of the reduced throughput region Λ_{int} compared to the full network capacity region Λ. The points B, A, and C in Fig. 2 maximize the sum throughput

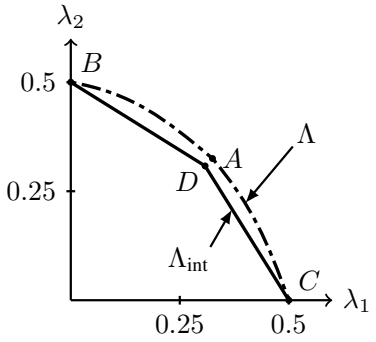

Fig. 2. The closeness of the reduced throughput region Λ_{int} and the network capacity region Λ in a two-user network with i.i.d. channels.

in the direction $(0, 1)$, $(1, 1)$, and $(1, 0)$, respectively, and the boundary of the network capacity region Λ shall be a concave curve connecting these points. The loss of the sum throughput

in the point D as compared to the optimal point A is less than or equal to [12, Section III-C]

$$\frac{\mathsf{P}_{01}}{x\mathsf{P}_{10}+\mathsf{P}_{01}}(1-x)^2, \quad x \triangleq \mathsf{P}_{01}+\mathsf{P}_{10}<1.$$

In general, in an N-user network with i.i.d. channels, the difference between the maximum sum throughput over the collection of RandRR policies and the optimal sum throughput is at most [12, Section III]

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

which decreases to zero geometrically fast with N. We refer readers to see additional examples in [46, Fig. 3], in which the tightness of Λ_{int} is compared to the full network capacity region Λ that is numerically computed in a two-user example.

IV. Network Utility Maximization

A. The QRRNUM policy

By the nature of its construction, the reduced throughput region Λ_{int} defined in the previous section can be viewed as a convex hull of the zero vector and (2^N-1) throughput vectors, where each of the (2^N-1) throughput vectors corresponds to the performance of a RandRR policy that serve a fixed subset of users in every round (cf. Theorem 2). This can be proved by a time sharing argument. As a result, the problem (2) is a well-defined convex program. Yet, solving (2) remains difficult because the feasible region Λ_{int} is represented as a convex hull of 2^N vectors. Next we introduce an admission control and channel scheduling policy that solves (2) in a dynamic manner with low complexity. In this policy, time is divided into frames. In each frame we serve a subset of users by one round of round robin; the frame size is equal to the duration of a round. The amount of data admitted in every slot of a frame is decided at the beginning of the frame.

Queue-Dependent Round Robin for Network Utility Maximization (QRRNUM)

- (Admission control) At the start of every frame, observe the current queue backlog $\big(Q_1(t),\ldots,Q_N(t)\big)$ and solve the convex program for each user n

$$\text{maximize} \quad V\, g_n(r_n) - Q_n(t)\, r_n \tag{14}$$
$$\text{subject to} \quad r_n \in [0,1], \tag{15}$$

  where $V>0$ is a predefined control parameter. Let r_n^{QRR} be the solution to (14)-(15). Admit r_n^{QRR} packets for user n into queue $Q_n(t)$ in every slot of the current frame.
- (Channel scheduling) At the start of every frame, over all nonzero binary vectors $\boldsymbol{\phi}=(\phi_n)_{n=1}^N \in \Phi$, let $\boldsymbol{\phi}^{\mathsf{QRR}}$ be the solution to

$$\max_{\boldsymbol{\phi}\in\Phi} \frac{\sum_{n=1}^N Q_n(t)\mathbb{E}\left[L_n^{\boldsymbol{\phi}}-1\right]\phi_n}{\sum_{n=1}^N \mathbb{E}\left[L_n^{\boldsymbol{\phi}}\right]\phi_n}, \tag{16}$$

  where $\mathbb{E}\left[L_n^{\boldsymbol{\phi}}\right]$ is given in (10). If the maximum of (16) is positive, serve the active channels in $\boldsymbol{\phi}^{\mathsf{QRR}}$ for one round with the ordering of least recently used first; the service on each channel follows Step 4 of the RandRR policy. If the maximum of (16) is less than or equal to zero, idle the system for one slot. At the end of either case, start a new frame of service.

The convex program (14)-(15) has a simple closed-form solution because we assume $g_n(\cdot)$ are differentiable. We illustrate the properties of the admission controller in an example. Let $g_n(r_n)=c_n\log(r_n)$ and the solution to (14)-(15) is

$$r_n^{\mathsf{QRR}} = \min\left[\frac{c_n V}{Q_n(t)}, 1\right]. \tag{17}$$

We have the observations from (17): (a) User n admits less data when the queue $Q_n(t)$ is more congested. (b) User n admits data more aggressively as compared to other users if it has a "better" utility function (in this example, with a larger c_n). (c) The parameter V reflects a performance tradeoff: A large V allows data to be admitted more aggressively so as to maximize the throughput utility, but incurs a large backlog in the queue (see Section IV-D for more discussions on this tradeoff). We can also visualize the above properties by observing the concave function (14).

We note that solving (16) is essentially a max-weight policy. The maximum (16) is equivalent to

$$\max_{\boldsymbol{\phi}\in\Phi}\sum_{n=1}^N Q_n(t)\mu_n^{\boldsymbol{\phi}}(t), \quad \mu_n^{\boldsymbol{\phi}}(t)=\begin{cases}\frac{\mathbb{E}\left[L_n^{\boldsymbol{\phi}}\right]-1}{\sum_{n:\phi_n=1}\mathbb{E}\left[L_n^{\boldsymbol{\phi}}\right]} & \text{if } \phi_n=1\\ 0 & \text{otherwise.}\end{cases}$$

From Theorem 1, the term $\mu_n^{\boldsymbol{\phi}}(t)$ is the average throughput for user n when channels $\boldsymbol{\phi}$ are chosen for service in a round.

B. Computing the maximum (16)

The most complex part of the QRRNUM policy is to solve (16). We introduce a bisection algorithm that searches for the maximum of (16) with exponentially fast speed. This algorithm is motivated by the next lemma.

Lemma 3. ([23, Lemma 7.5]) Let $a(\boldsymbol{\phi})$ and $b(\boldsymbol{\phi})$ denote the numerator and denominator of (16), respectively. Define

$$\theta^* \triangleq \max_{\boldsymbol{\phi}\in\Phi}\left\{\frac{a(\boldsymbol{\phi})}{b(\boldsymbol{\phi})}\right\},\ c(\theta)\triangleq\max_{\boldsymbol{\phi}\in\Phi}\left[a(\boldsymbol{\phi})-\theta b(\boldsymbol{\phi})\right].$$

Then:

$$\begin{aligned}&\text{If } \theta<\theta^*, \text{ then } c(\theta)>0.\\ &\text{If } \theta>\theta^*, \text{ then } c(\theta)<0.\\ &\text{If } \theta=\theta^*, \text{ then } c(\theta)=0.\end{aligned}$$

The value $c(\theta)$ in Lemma 3 is easily computed by noticing

$$c(\theta)=\max_{k\in\{1,\ldots,N\}}\left\{\max_{\boldsymbol{\phi}\in\Phi_k}\left[a(\boldsymbol{\phi})-\theta b(\boldsymbol{\phi})\right]\right\}, \tag{18}$$

where $\Phi_k\subset\Phi$ denotes the set of binary vectors having k ones. For every $k\in\{1,\ldots,N\}$, the inner maximum of (18) is equal to

$$\max_{\boldsymbol{\phi}\in\Phi_k}\left\{\sum_{n=1}^N\left[\frac{\mathsf{P}_{n,01}^{(k)}}{\mathsf{P}_{n,10}}\left(Q_n(t)-\theta\right)-\theta\right]\phi_n\right\}. \tag{19}$$

This is solved by computing $\left[\frac{\mathsf{P}_{n,01}^{(k)}}{\mathsf{P}_{n,10}}\left(Q_n(t)-\theta\right)-\theta\right]$ for each user and activating the k channels (i.e., setting their ϕ_n to be 1) with the first k largest values. The complexity of computing (19) is as follows. It takes $O(N)$ to compute $\left[\frac{\mathsf{P}_{n,01}^{(k)}}{\mathsf{P}_{n,10}}\left(Q_n(t)-\theta\right)-\theta\right]$ for all users, $O(N\log N)$ to sort, and $O(N)$ to add the first k largest values; it takes $O(N\log N)$ to compute (19). Thus, it takes $O(N^2\log N)$ to compute $c(\theta)$.

From Lemma 3, we can search for the maximum ratio θ^* as follows. Suppose initially we know θ^* lies in some interval $[\theta_{\min},\theta_{\max}]$. We compute the midpoint $\theta_{\text{mid}}=\frac{1}{2}(\theta_{\min}+\theta_{\max})$ and evaluate $c(\theta_{\text{mid}})$. If $c(\theta_{\text{mid}})>0$, Lemma 3 indicates that $\theta_{\text{mid}}<\theta^*$, and we know θ^* lies in the reduced region $[\theta_{\text{mid}},\theta_{\max}]$, which is half the size of the initial region $[\theta_{\min},\theta_{\max}]$. Hence, one such bisection operation reduces the feasible region of the unknown θ^* by half. By iterating the bisection process, we can find θ^* with exponential speed.

With the knowledge of θ^*, the maximizer $\boldsymbol{\phi}^{\mathsf{QRR}}$ of (16) is the maximizer of $c(\theta^*)$. To see this, by definition we have

$$\theta^*=\frac{a(\boldsymbol{\phi}^{\mathsf{QRR}})}{b(\boldsymbol{\phi}^{\mathsf{QRR}})}\geq\frac{a(\boldsymbol{\phi})}{b(\boldsymbol{\phi})},\quad\text{for all }\boldsymbol{\phi}\in\Phi.$$

It follows that

$$a(\boldsymbol{\phi}^{\mathsf{QRR}})-\theta^*\,b(\boldsymbol{\phi}^{\mathsf{QRR}})=0\geq a(\boldsymbol{\phi})-\theta^*\,b(\boldsymbol{\phi}),\quad\text{for all }\boldsymbol{\phi}\in\Phi.$$

Thus $\boldsymbol{\phi}^{\mathsf{QRR}}$ is the maximizer of $\max_{\boldsymbol{\phi}\in\Phi}[a(\boldsymbol{\phi})-\theta^*\,b(\boldsymbol{\phi})]$.

Based on the above discussions, the bisection algorithm that maximizes (16) is presented next.

Bisection Algorithm that Solves (16)

- Initially, define $\theta_{\min}\triangleq 0$ and
$$\theta_{\max}\triangleq\left(\sum_{n=1}^{N}Q_n(t)\right)\left(\sum_{n=1}^{N}\frac{\pi_{n,\mathsf{ON}}}{\mathsf{P}_{n,10}}\right).$$
It is not hard to verify that $\theta_{\min}\leq a(\boldsymbol{\phi})/b(\boldsymbol{\phi})\leq\theta_{\max}$ for all $\boldsymbol{\phi}\in\Phi$, where $a(\boldsymbol{\phi})$ and $b(\boldsymbol{\phi})$ denote the numerator and the denominator of (16), respectively. Then, $\theta^*\in[\theta_{\min},\theta_{\max}]$.
- Compute $\theta_{\text{mid}}=\frac{1}{2}(\theta_{\min}+\theta_{\max})$ and $c(\theta_{\text{mid}})$. If $c(\theta_{\text{mid}})=0$, we have $\theta^*=\theta_{\text{mid}}$ and
$$\boldsymbol{\phi}^{\mathsf{QRR}}=\operatorname{argmax}_{\boldsymbol{\phi}\in\Phi}\left[a(\boldsymbol{\phi})-\theta^*b(\boldsymbol{\phi})\right].$$
When $c(\theta_{\text{mid}})<0$, update the feasible region $[\theta_{\min},\theta_{\max}]$ of θ^* as $[\theta_{\min},\theta_{\text{mid}}]$. If $c(\theta_{\text{mid}})>0$, update the region as $[\theta_{\text{mid}},\theta_{\max}]$. In either case, repeat the bisection process.

C. Lyapunov drift inequality

The construction of the QRRNUM policy follows a novel Lyapunov drift argument. We start with constructing a frame-based Lyapunov drift inequality over a frame of size T, where T is possibly random but has a finite second moment bounded by a constant C so that $C\geq\mathbb{E}\left[T^2\mid\boldsymbol{Q}(t)\right]$ for all t and all possible $\boldsymbol{Q}(t)$. Intuition for the inequality is provided later. By iteratively applying (5), it is not hard to show

$$Q_n(t+T)\leq\max\left[Q_n(t)-\sum_{\tau=0}^{T-1}\mu_n(t+\tau),0\right]+\sum_{\tau=0}^{T-1}r_n(t+\tau)\tag{20}$$

for each $n\in\{1,\ldots,N\}$. We define the quadratic Lyapunov function $L(\boldsymbol{Q}(t))\triangleq\frac{1}{2}\sum_{n=1}^{N}Q_n^2(t)$ as a scalar measure of the queue vector $\boldsymbol{Q}(t)$. Define the T-slot Lyapunov drift

$$\Delta_T(\boldsymbol{Q}(t))\triangleq\mathbb{E}\left[L(\boldsymbol{Q}(t+T))-L(\boldsymbol{Q}(t))\mid\boldsymbol{Q}(t)\right]$$

as the conditional expected difference of the queue sizes over T slots, where the expectation is with respect to the randomness of the network (time-varying channels and the possibly random control actions) over T slots and the randomness of the duration T. By taking square of (20) for every n, using inequalities

$$\max[a-b,0]\leq a\quad\forall a\geq 0,$$
$$(\max[a-b,0])^2\leq(a-b)^2,\quad\mu_n(t)\leq 1,\quad r_n(t)\leq 1$$

to simplify terms, summing all resulting inequalities, and taking conditional expectation on $\boldsymbol{Q}(t)$, we can show

$$\begin{aligned}&\Delta_T(\boldsymbol{Q}(t))\leq B\\&-\mathbb{E}\left[\sum_{n=1}^{N}Q_n(t)\left[\sum_{\tau=0}^{T-1}\mu_n(t+\tau)-r_n(t+\tau)\right]\mid\boldsymbol{Q}(t)\right]\end{aligned}\tag{21}$$

where $B\triangleq NC>0$ is a constant. Subtracting from both sides of (21) the term $V\,\mathbb{E}\left[\sum_{\tau=0}^{T-1}\sum_{n=1}^{N}g_n(r_n(t+\tau))\mid\boldsymbol{Q}(t)\right]$ where $V>0$ is a predefined control parameter, we get the Lyapunov drift inequality

$$\begin{aligned}&\Delta_T(\boldsymbol{Q}(t))-V\mathbb{E}\left[\sum_{\tau=0}^{T-1}\sum_{n=1}^{N}g_n(r_n(t+\tau))\mid\boldsymbol{Q}(t)\right]\\&\quad\leq B-f(\boldsymbol{Q}(t))-h(\boldsymbol{Q}(t)),\end{aligned}\tag{22}$$

where

$$f(\boldsymbol{Q}(t))\triangleq\sum_{n=1}^{N}Q_n(t)\mathbb{E}\left[\sum_{\tau=0}^{T-1}\mu_n(t+\tau)\mid\boldsymbol{Q}(t)\right]\tag{23}$$

$$\begin{aligned}h(\boldsymbol{Q}(t))\triangleq\mathbb{E}\Bigg[\sum_{\tau=0}^{T-1}\Bigg[&V\sum_{n=1}^{N}g_n(r_n(t+\tau))\\&-\sum_{n=1}^{N}Q_n(t)r_n(t+\tau)\Bigg]\mid\boldsymbol{Q}(t)\Bigg].\end{aligned}\tag{24}$$

The inequality (22) holds for any admission control and scheduling policy over a duration of any size T.

D. Intuition behind the Lyapunov drift inequality

The desired network control policy shall stabilize all queues $(Q_1(t),\ldots,Q_N(t))$ and maximize the throughput utility $\sum_{n=1}^{N}g_n(\overline{y}_n)$. For queue stability, we want to minimize the Lyapunov drift $\Delta_T(\boldsymbol{Q}(t))$, because it captures the expected growth of queue sizes over a duration of time. To increase throughput utility, we want to admit more data into the system for service and maximize the expected sum utility $\mathbb{E}\left[\sum_{\tau=0}^{T-1}\sum_{n=1}^{N}g_n(r_n(t+\tau))\mid\boldsymbol{Q}(t)\right]$. Minimizing Lya-

punov drift and maximizing throughput utility, however, conflict with each other, because queue sizes increase with more data admitted into the system. To capture this tradeoff, it is natural to minimize a weighted difference of Lyapunov drift and throughput utility, which is the left side of (22). The tradeoff is controlled by the positive parameter V. Intuitively, a large V value puts more weights on throughput utility, thus throughput utility is improved, at the expense of the growth of the queue sizes captured in $\Delta_T(\boldsymbol{Q}(t))$. The construction of the inequality (22) provides a useful upper bound on the weighted difference of Lyapunov drift and throughput utility.

The QRRNUM policy that we construct in the next section uses the above ideas with two modifications. First, it suffices to minimize a bound on the weighted difference of Lyapunov drift and throughput utility, i.e., the right side of (22). Second, since the weighted difference of Lyapunov drift and throughput utility in (22) is made over a frame of T slots, where the value of T is random and depends on the policy used within the frame, it is natural to normalize the weighted difference by the average frame size, and we will minimize the resulting ratio (see (25)). This new ratio rule is a generalization of existing MaxWeight policies for stochastic network optimization over frame-based systems. These two modifications lead to the next analysis.

E. Construction of the QRRNUM *policy*

We consider the policy that, at the start of every round, observes the current queue vector $\boldsymbol{Q}(t)$ and maximizes over all feasible policies the average

$$\frac{f(\boldsymbol{Q}(t)) + h(\boldsymbol{Q}(t))}{\mathbb{E}\left[T \mid \boldsymbol{Q}(t)\right]} \tag{25}$$

over a frame of size T. Every feasible policy here consists of: (1) an admission control mechanism that admits packets into queues $\boldsymbol{Q}(t)$ for all users in every slot; (2) a randomized round robin policy RandRR (given in Section III-C) for data delivery. The frame size T in (25) is considered as the length of one transmission round under the candidate RandRR policy, and its distribution depends on the backlog vector $\boldsymbol{Q}(t)$ via the queue-dependent choice of RandRR. When the feasible policy that maximizes (25) is chosen, it is executed for one round of transmission, after which a new policy is chosen by maximizing the updated ratio of (25), and so on.

We simplify the maximization of (25); the result is the QRRNUM policy. In $h(\boldsymbol{Q}(t))$ (see (24)), the optimal admitted data vector $(r_n(t+\tau))$ in every slot is independent of the frame size T and of the rate allocations $\mu_n(t+\tau)$ in $f(\boldsymbol{Q}(t))$ (see (23)). In addition, it should be the same for all $\tau \in \{0, \ldots, T-1\}$, and is the solution to (14)-(15). These observations lead to the admission control subroutine in the QRRNUM policy.

Let $\Psi^*(\boldsymbol{Q}(t))$ denote the optimal objective of (14)-(15). Since the optimal admitted data vector is independent of the frame size T, we have $h(\boldsymbol{Q}(t)) = \mathbb{E}\left[T \mid \boldsymbol{Q}(t)\right]\Psi^*(\boldsymbol{Q}(t))$, and (25) is equal to

$$\frac{f(\boldsymbol{Q}(t))}{\mathbb{E}\left[T \mid \boldsymbol{Q}(t)\right]} + \Psi^*(\boldsymbol{Q}(t)). \tag{26}$$

It indicates that finding the optimal admission policy is independent of finding the optimal RandRR policy that maximizes the first term of (26).

Next we evaluate the first term of (26) under a fixed RandRR policy with parameters $\{\alpha_{\boldsymbol{\phi}}\}_{\boldsymbol{\phi}\in\Phi\cup\{\mathbf{0}\}}$. Conditioning on the subset $\boldsymbol{\phi}$ of channels served in a round, we have

$$f(\boldsymbol{Q}(t)) = \sum_{\boldsymbol{\phi}\in\Phi\cup\{\mathbf{0}\}} \alpha_{\boldsymbol{\phi}}\, f(\boldsymbol{Q}(t), \boldsymbol{\phi}),$$

where $f(\boldsymbol{Q}(t), \boldsymbol{\phi})$ denotes the term $f(\boldsymbol{Q}(t))$ evaluated when channels $\boldsymbol{\phi}$ are served for one round. If $\boldsymbol{\phi} = \mathbf{0}$, $f(\boldsymbol{Q}(t), \mathbf{0})$ corresponds to the decision of idling the system for one slot. Similarly, by conditioning we can show [5]

$$\mathbb{E}\left[T\right] = \mathbb{E}\left[T \mid \boldsymbol{Q}(t)\right] = \sum_{\boldsymbol{\phi}\in\Phi\cup\{\mathbf{0}\}} \alpha_{\boldsymbol{\phi}}\, \mathbb{E}\left[T_{\boldsymbol{\phi}}\right],$$

where $T_{\boldsymbol{\phi}}$ denotes the duration of serving channels $\boldsymbol{\phi}$ for one round; note that $T_{\mathbf{0}} = 1$ when $\boldsymbol{\phi} = \mathbf{0}$. It follows that

$$\frac{f(\boldsymbol{Q}(t))}{\mathbb{E}\left[T \mid \boldsymbol{Q}(t)\right]} = \frac{\sum_{\boldsymbol{\phi}\in\Phi\cup\{\mathbf{0}\}} \alpha_{\boldsymbol{\phi}}\, f(\boldsymbol{Q}(t), \boldsymbol{\phi})}{\sum_{\boldsymbol{\phi}\in\Phi\cup\{\mathbf{0}\}} \alpha_{\boldsymbol{\phi}}\, \mathbb{E}\left[T_{\boldsymbol{\phi}}\right]}. \tag{27}$$

The next lemma shows that the maximum of (27) over the collection of RandRR policies is achieved by either serving a subset of channels for one round or idling the system for one slot.

Lemma 4. We index the 2^N vectors $\boldsymbol{\phi} \in \Phi \cup \{\mathbf{0}\}$. For the vector $\boldsymbol{\phi}$ with index k we define

$$f_k \triangleq f(\boldsymbol{Q}(t), \boldsymbol{\phi}), \quad D_k \triangleq \mathbb{E}\left[T_{\boldsymbol{\phi}}\right].$$

Without loss of generality, assume

$$\frac{f_1}{D_1} \geq \frac{f_k}{D_k}, \quad \forall k \in \{2, 3, \ldots, 2^N\}.$$

Then for any probability distribution $\{\alpha_k\}_{k\in\{1,\ldots,2^N\}}$ with $\alpha_k \geq 0$ and $\sum_{k=1}^{2^N} \alpha_k = 1$, we have

$$\frac{f_1}{D_1} \geq \frac{\sum_{k=1}^{2^N} \alpha_k\, f_k}{\sum_{k=1}^{2^N} \alpha_k\, D_k}.$$

Proof of Lemma 4: In Appendix A. ∎

If maximizing (27) is to idle the system for one slot, we have $f(\boldsymbol{Q}(t))/\mathbb{E}\left[T \mid \boldsymbol{Q}(t)\right] = 0$. Otherwise, let us now evaluate (27) when it is optimal to serve the channels $\boldsymbol{\phi} \in \Phi$ in this round. From Theorem 1, we have

$$\mathbb{E}\left[T\right] = \mathbb{E}\left[T \mid \boldsymbol{Q}(t)\right] = \sum_{n:\phi_n=1} \mathbb{E}\left[L_n^{\boldsymbol{\phi}}\right],$$

$$\mathbb{E}\left[\sum_{\tau=0}^{T-1} \mu_n(t+\tau) \mid \boldsymbol{Q}(t)\right] = \begin{cases} \mathbb{E}\left[L_n^{\boldsymbol{\phi}}\right] - 1 & \text{if } \phi_n = 1 \\ 0 & \text{if } \phi_n = 0 \end{cases}$$

As a result,

$$\frac{f(\boldsymbol{Q}(t))}{\mathbb{E}\left[T \mid \boldsymbol{Q}(t)\right]} = \frac{\sum_{n=1}^{N} Q_n(t)\mathbb{E}\left[L_n^{\boldsymbol{\phi}} - 1\right]\phi_n}{\sum_{n=1}^{N} \mathbb{E}\left[L_n^{\boldsymbol{\phi}}\right]\phi_n},$$

which is the ratio in (16). The above discussions lead to the

[5] Given a fixed RandRR policy, the frame size T no longer depends on the backlog vector $\boldsymbol{Q}(t)$. Thus $\mathbb{E}\left[T\right] = \mathbb{E}\left[T \mid \boldsymbol{Q}(t)\right]$.

channel scheduling subroutine of the QRRNUM policy.

V. Performance Analysis

Theorem 3. For any control parameter $V > 0$, the QRRNUM policy stabilizes all queues $\{Q_1(t), \ldots, Q_N(t)\}$ and yields a throughput utility satisfying

$$\liminf_{t\to\infty} \sum_{n=1}^{N} g_n \left(\frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\left[r_n(\tau)\right] \right) \geq g^* - \frac{B}{V},$$

where g^* is *the optimal utility in the optimization problem* (2) and $B = N\mathbb{E}\left[T_{\max}^2\right]$ is a finite constant, where $T_{\max}$ denotes the duration of serving all channels for one round by the QRRNUM policy.

Proof of Theorem 3: In Appendix B. ∎

Theorem 3 shows that the throughput utility rendered by the QRRNUM policy is at most B/V away from g^*. By choosing V sufficiently large, the utility can be made arbitrarily close to g^*. The tradeoff is that, as shown in (43), the average queue size grows linearly with V. Such a tradeoff agrees with the design principle of the QRRNUM policy discussed in Section IV-D.

VI. Simulations

A. Rate proportional fairness

We use the QRRNUM policy to solve a variant of the rate proportional fairness problem. We consider a two-user network with i.i.d. channels that have transition probabilities $\mathsf{P}_{01} = \mathsf{P}_{10} = 0.2$. From [12, Theorem 1], the throughput region Λ_{int} is a convex hull of the set $\{(0,0), (0.5,0), (0,0.5), (4/13, 4/13)\}$; it is illustrated in Fig. 2. We consider the problem:

$$\text{maximize} \quad f(\overline{y}_1, \overline{y}_2) = 2\log(1+\overline{y}_1) + \log(1+\overline{y}_2) \quad (28)$$

$$\text{subject to} \quad (\overline{y}_1, \overline{y}_2) \in \Lambda_{\text{int}}. \quad (29)$$

The solution to (28)-(29) is $(5/12, 2/15) \approx (0.4167, 0.1333)$. We simulate the QRRNUM policy for the problem (28)-(29) for 10^6 rounds. The simulation result in Table I shows that

V	$\overline{y}_1$	$\overline{y}_2$	$f(\overline{y}_1, \overline{y}_2)$
10	0.391	0.1477	0.7977
100	0.4133	0.1392	0.8221
1000	0.4165	0.1345	0.8226
solution to (28)-(29)	0.4167	0.1333	0.8218

TABLE I
Simulation for the QRRNUM policy in a rate proportional fairness problem.

the throughput utility approaches the optimal value for the problem (28)-(29) as V increases, as proved in Theorem 3.

B. Comparison to Whittle's index

Whittle's index [10] is a well-known heuristic for RMAB problems with linear cost functions. The index for each project (channel) at a given state represents the minimum subsidy for which we would choose not to play the project. In other words, it captures the attractiveness of a given state on a project. The index policy is simply to play the arm with the largest index in every slot.

For Markov ON/OFF channels, Whittle's index exists and is computed in closed form in [13]. Here, we compare the QRRNUM policy with Whittle's index by simulations in the case of linear reward functions. We consider a network of 10 independent, positively correlated Markov ON/OFF channels. The objective is to maximize the weighted sum throughput $\sum_{n=1}^{10} c_n \overline{y}_n$, where $c_1, \ldots, c_{10}$ are positive weights.

We conduct 20 simulation runs. In each run, the weights c_n are normalized and randomly generated according to a uniform distribution over $(0, 1]$. The transition probabilities for each channel are randomly generated as well. Every simulation run lasts for 10^5 rounds for the QRRNUM policy, and 10^6 slots for the Whittle's index policy. We use $V = 1000$ throughout in the QRRNUM policy.

In our simulations, we take advantage of a minor improvement on the QRRNUM policy: When a channel is selected for transmission, we simply keep transmitting until the channel turns OFF. This modification yields a better throughput because we do not transmit dummy packets.

Fig. 3 compares the performance of the QRRNUM policy and Whittle's index. Every data point (a, b) in Fig. 3 corresponds to one of the 20 simulation runs, where coordinates a and b are the weighted sum throughput under Whittle's index and the QRRNUM policy, respectively. We note that Whittle's index outperforms QRRNUM if a data point lies below the dotted line $x = y$, and vice versa. These simulations suggest that the two policies have comparable performance over channels of fairly arbitrary statistics.

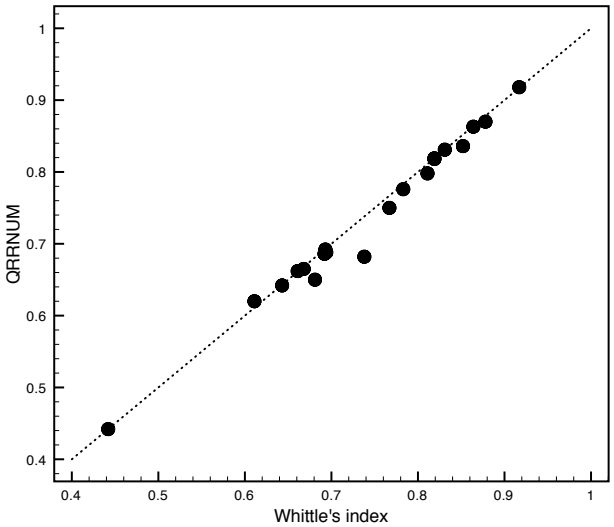

Fig. 3. Comparison of the weighted sum throughput under the QRRNUM policy and Whittle's index in a 10-user network with randomly generated channel statistics and user rewards.

VII. Conclusion

We provide an analytical framework for network utility maximization over partially observable Markov ON/OFF

channels. The performance and control in this network are constrained by limiting channel probing and delayed/uncertain channel state information, but can be improved by exploiting channel memory. Equivalently, we consider a restless multi-armed bandit (RMAB) problem with concave reward functions, which is difficult to solve using existing tools such as Whittle's index or Markov decision theory. We adopt an achievable region method that uses a novel ratio MaxWeight policy to solve the RMAB problem over a reduced throughput region constructed by randomizing well-designed round robin policies. Extensions of this new achievable region method to other open stochastic convex optimization problems are interesting future research.

REFERENCES

[1] C.-P. Li and M. J. Neely, "Network utility maximization over partially observable Markovian channels," in *IEEE Proc. Int. Symp. Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, Princeton, NJ, USA, May 2011.

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

[3] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Trans. Netw.*, vol. 8, no. 5, pp. 556–567, Oct. 2000.

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

[5] M. J. Neely, E. Modiano, and C. E. Rohrs, "Power allocation and routing in multibeam satellites with time-varying channels," *IEEE/ACM Trans. Netw.*, vol. 11, no. 1, pp. 138–152, Feb. 2003.

[6] A. Eryilmaz, R. Srikant, and J. R. Perkins, "Stable scheduling policies for fading wireless channels," *IEEE/ACM Trans. Netw.*, vol. 13, no. 2, pp. 411–424, Apr. 2005.

[7] Q. Zhao and B. M. Sadler, "A survey of dynamic spectrum access," *IEEE Signal Process. Mag.*, vol. 24, no. 3, pp. 79–89, May 2007.

[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] J. L. Ny, M. Dahleh, and E. Feron, "Multi-uav dynamic routing with partial observations using restless bandit allocation indices," in *American Control Conference*, Seattle, WA, USA, Jun. 2008.

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

[11] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of optimal queueing network control," *Math. of Oper. Res.*, vol. 24, pp. 293–305, May 1999.

[12] C.-P. Li and M. J. Neely, "Exploiting channel memory for multiuser wireless scheduling without channel measurement: Capacity regions and algorithms," *Performance Evaluation*, vol. 68, no. 8, pp. 631–657, Aug. 2011.

[13] K. Liu and Q. Zhao, "Indexability of restless bandit problems and optimality of whittle's index for dynamic multichannel access," *IEEE Trans. Inf. Theory*, vol. 56, no. 11, pp. 5547–5567, Nov. 2010.

[14] 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.

[15] S. M. Ross, *Stochastic Processes*, 2nd ed. Wiley, 1996.

[16] C.-P. Li and M. J. Neely, "Delay and rate-optimal control in a multi-class priority queue with adjustable service rates," in *IEEE Proc. INFOCOM (mini-conference)*, 2012.

[17] C.-P. Li, "Stochastic optimization over parallel queues: Channel-blind scheduling, restless bandit, and optimal delay," Ph.D. dissertation, University of Southern California, 2011.

[18] D. D. Yao, "Dynamic scheduling via polymatroid optimization," in *Performance Evaluation of Complex Systems: Techniques and Tools, Performance 2002, Tutorial Lectures*. London, UK: Springer-Verlag, 2002, pp. 89–113.

[19] J. Walrand, *An Introduction to Queueing Networks*. Prentice Hall, 1988.

[20] R. Urgaonkar and M. J. Neely, "Opportunistic cooperation in cognitive femtocell networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 3, pp. 607–616, Apr. 2012.

[21] M. J. Neely, A. S. Tehrani, and Z. Zhang, "Dynamic index coding for wireless broadcast networks," in *IEEE Proc. INFOCOM*, 2012.

[22] M. J. Neely, "Asynchronous scheduling for energy optimality in systems with multiple servers," in *Conf. Information Sciences and Systems*, Mar. 2012.

[23] ——, *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan & Claypool, 2010.

[24] ——, "Dynamic power allocation and routing for satellite and wireless networks with time varying channels," Ph.D. dissertation, Massachusetts Institute of Technology, November 2003.

[25] M. J. Neely, E. Modiano, and C.-P. Li, "Fairness and optimal stochastic control for heterogeneous networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 2, pp. 396–409, Apr. 2008.

[26] A. Eryilmaz and R. Srikant, "Fair resource allocation in wireless networks using queue-length-based scheduling and congestion control," *IEEE/ACM Trans. Netw.*, vol. 15, no. 6, pp. 1333–1344, Dec. 2007.

[27] ——, "Joint congestion control, routing, and mac for stability and fairness in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1514–1524, Aug. 2006.

[28] X. Lin and N. B. Shroff, "Joint rate control and scheduling in multihop wireless networks," in *IEEE Conf. Decision and Control (CDC)*, Dec. 2004, pp. 1484–1489.

[29] A. L. Stolyar, "Maximizing queueing network utility subject to stability: Greedy primal-dual algorithm," *Queueing Syst.*, vol. 50, no. 4, pp. 401–457, 2005.

[30] X. Lin, N. B. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1452–1463, Aug. 2006.

[31] 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.

[32] 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.

[33] N. B. Chang and M. Liu, "Optimal channel probing and transmission scheduling for opportunistic spectrum access," in *ACM Int. Conf. Mobile Computing and Networking (MobiCom)*, New York, NY, 2007, pp. 27–38.

[34] P. Chaporkar, A. Proutiere, and H. Asnani, "Learning to optimally exploit multi-channel diversity in wireless systems," in *IEEE Proc. INFOCOM*, 2010.

[35] P. Chaporkar and A. Proutiere, "Optimal joint probing and transmission strategy for maximizing throughput in wireless systems," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 8, pp. 1546–1555, Oct. 2008.

[36] S. Guha, K. Munagala, and S. Sarkar, "Jointly optimal transmission and probing strategies for multichannel wireless systems," in *Conf. Information Sciences and Systems*, Mar. 2006.

[37] W. Ouyang, S. Murugesan, A. Eryilmaz, and N. B. Shroff, "Scheduling with rate adaptation under incomplete knowledge of channel/estimator statistics," in *Allerton Conf. Communication, Control, and Computing*, 2010.

[38] A. Gopalan, C. Caramanis, and S. Shakkottai, "On wireless scheduling with partial channel-state information," *IEEE Trans. Inf. Theory*, vol. 58, no. 1, pp. 403–420, Jan. 2012.

[39] L. Ying and S. Shakkottai, "On throughput optimality with delayed network-state information," *IEEE Trans. Inf. Theory*, vol. 57, no. 8, pp. 5116–5132, Aug. 2011.

[40] K. Kar, X. Luo, and S. Sarkar, "Throughput-optimal scheduling in multichannel access point networks under infrequent channel measurements," *IEEE Trans. Wireless Commun.*, vol. 7, no. 7, pp. 2619–2629, Jul. 2008.

[41] 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.

[42] 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.

[43] S. H. A. Ahmad and M. Liu, "Multi-channel opportunistic access: A case of restless bandits with multiple plays," in *Allerton Conf. Communication, Control, and Computing*, 2009, pp. 1361–1368.

[44] J. Niño-Mora, "An index policy for dynamic fading-channel allocation to heterogeneous mobile users with partial observations," in *Next Generation Internet Networks*, 2008, pp. 231–238.

[45] S. Guha, K. Munagala, and P. Shi, "Approximation algorithms for restless bandit problems," *Journal of the ACM*, vol. 58, no. 1, Dec. 2010.

[46] K. Jagannathan, I. Menache, E. Modiano, and S. Mannor, "A state action frequency approach to throughput maximization over uncertain wireless channels," in *IEEE Proc. INFOCOM*, Shanghai, China, Apr. 2011.
[47] S. Murugesan, P. Schniter, and N. Shroff, "Multiuser scheduling in a markov-modeled downlink using randomly delayed ARQ feedback," *IEEE Trans. Inf. Theory*, vol. 58, no. 2, pp. 1025–1042, Feb. 2012.
[48] W. Ouyang, A. Eryilmaz, and N. Shroff, "Asymptotically optimal downlink scheduling over markovian fading channels," in *IEEE Proc. INFOCOM*, Mar. 2012, pp. 1224–1232.
[49] W. Ouyang, S. Murugesan, A. Eryilmaz, and N. Shroff, "Exploiting channel memory for joint estimation and scheduling in downlink networks," in *IEEE Proc. INFOCOM*, Apr. 2011, pp. 3056–3064.
[50] R. Weber and G. Weiss, "On an index policy for restless bandits," *J. Appl. Probab.*, vol. 27, pp. 637–648, 1990.
[51] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed. Athena Scientific, 2005, vol. I.

APPENDIX A

Proof of Lemma 4: Fact 1: Let $\{a_1, a_2, b_1, b_2\}$ be four positive numbers, and suppose there is a bound z such that $a_1/b_1 \le z$ and $a_2/b_2 \le z$. Then for any probability θ (where $0 \le \theta \le 1$), we have:

$$\frac{\theta a_1 + (1-\theta)a_2}{\theta b_1 + (1-\theta)b_2} \le z. \tag{30}$$

We prove Lemma 4 by induction and (30). Initially, for any $\alpha_1,\, \alpha_2 \ge 0$, $\alpha_1 + \alpha_2 = 1$, from $f_1/D_1 \ge f_2/D_2$ we get

$$\frac{f_1}{D_1} \ge \frac{\alpha_1 f_1 + \alpha_2 f_2}{\alpha_1 D_1 + \alpha_2 D_2}.$$

For some $K > 2$, assume

$$\frac{f_1}{D_1} \ge \frac{\sum_{k=1}^{K-1} \alpha_k f_k}{\sum_{k=1}^{K-1} \alpha_k D_k}$$

holds for any probability distribution $\{\alpha_k\}_{k=1}^{K-1}$. It follows that, for any probability distribution $\{\alpha_k\}_{k=1}^{K}$, we get

$$\frac{\sum_{k=1}^{K} \alpha_k f_k}{\sum_{k=1}^{K} \alpha_k D_k} = \frac{(1-\alpha_K)\left[\sum_{k=1}^{K-1} \frac{\alpha_k}{1-\alpha_K} f_k\right] + \alpha_K f_K}{(1-\alpha_K)\left[\sum_{k=1}^{K-1} \frac{\alpha_k}{1-\alpha_K} D_k\right] + \alpha_K D_K} \overset{(a)}{\le} \frac{f_1}{D_1}$$

where (a) follows Fact 1, because $f_1/D_1 \ge f_K/D_K$ by definition and

$$\frac{f_1}{D_1} \ge \frac{\sum_{k=1}^{K-1} \frac{\alpha_k}{1-\alpha_K} f_k}{\sum_{k=1}^{K-1} \frac{\alpha_k}{1-\alpha_K} D_k}$$

by the induction assumption. ∎

APPENDIX B

Proof of Theorem 3: In the QRRNUM policy, let t_{k-1} and T_k be the beginning and the duration of the kth transmission round, respectively. We have $T_k = t_k - t_{k-1}$ and $t_k = \sum_{i=1}^{k} T_i$ for all $k \in \mathbb{N}$. Assume $t_0 = 0$. The term T_k is the duration of serving a subset of channels in the kth round of the QRRNUM policy. Before we proceed, we present a useful lemma.

Lemma 5. Let $T_{\max}$ be the duration of serving all channels for one round by a RandRR policy. Then

1) The random variable $T_{\max}$ is stochastically larger than T_k for all $k \in \{1, 2, 3, \ldots\}$; i.e., $T_{\max} \ge_{\text{st}} T_k$. In other words,
$$\Pr[T_{\max} > a] \ge \Pr[T_k > a], \quad \forall a \in \{0, 1, 2, \ldots\}.$$
2) The random variable $T_{\max}^2$ is stochastically larger than T_k^2 for all $k \in \{1, 2, 3, \ldots\}$.
3) We have, for all $k \in \{1, 2, 3, \ldots\}$,
$$\mathbb{E}[T_k] \le \mathbb{E}[T_{\max}] < \infty, \quad \mathbb{E}\left[T_k^2\right] \le \mathbb{E}\left[T_{\max}^2\right] < \infty.$$

Proof of Lemma 5: In Appendix C. ∎

To analyze the performance of the QRRNUM policy, we compare it to a near-optimal feasible solution of the optimization problem (2). We will adopt the approach in [25] but generalize it to a frame-based analysis. For some $\epsilon > 0$, consider the *ϵ-constrained* version of the optimization problem (2):

$$\text{maximize} \sum_{n=1}^{N} g_n(\overline{y}_n), \quad \text{subject to } (\overline{y}_n)_{n=1}^{N} \in \Lambda_{\text{int}}(\epsilon), \tag{31}$$

where $\Lambda_{\text{int}}(\epsilon)$ is the achievable region Λ_{int} stripping an "ϵ-layer" off the boundary:

$$\Lambda_{\text{int}}(\epsilon) \triangleq \{(\overline{y}_n)_{n=1}^{N} \mid (\overline{y}_n + \epsilon)_{n=1}^{N} \in \Lambda_{\text{int}}\}.$$

Notice that $\Lambda_{\text{int}}(\epsilon) \to \Lambda_{\text{int}}$ as $\epsilon \to 0$. Let $(\overline{y}_n^*(\epsilon))$ be the solution to (31) and $(\overline{y}_n^*)$ be the solution to problem (2). For simplicity, we assume $(\overline{y}_n^*(\epsilon))$ converges to $(\overline{y}_n^*)$ as $\epsilon \to 0$.[6]

By definition of the reduced throughput region Λ_{int}, there exists a randomized round robin policy $\mathsf{RandRR}_\epsilon^*$ that yields the throughput vector $(\overline{y}_n^*(\epsilon) + \epsilon) \in \Lambda_{\text{int}}$. Let T_ϵ^* denote the length of one transmission round under $\mathsf{RandRR}_\epsilon^*$. From Lemma 2, we have for every user $n \in \{1, \ldots, N\}$:

$$\mathbb{E}\left[\sum_{\tau=0}^{T_\epsilon^*-1} \mu_n(t+\tau) \mid \boldsymbol{Q}(t)\right] = \mathbb{E}\left[\sum_{\tau=0}^{T_\epsilon^*-1} \mu_n(t+\tau)\right] = (\overline{y}_n^*(\epsilon) + \epsilon)\mathbb{E}[T_\epsilon^*].$$

Combining the policy $\mathsf{RandRR}_\epsilon^*$ with the admission control policy σ^* that sets $r_n(t+\tau) = \overline{y}_n^*(\epsilon)$ for all users n and all $\tau \in \{0, \ldots, T_\epsilon^* - 1\}$,[7] we get

$$f_\epsilon^*(\boldsymbol{Q}(t)) = \mathbb{E}[T_\epsilon^*] \sum_{n=1}^{N} Q_n(t)(\overline{y}_n^*(\epsilon) + \epsilon) \tag{32}$$

$$h_\epsilon^*(\boldsymbol{Q}(t)) = \mathbb{E}[T_\epsilon^*] \left[V g_\epsilon^* - \sum_{n=1}^{N} Q_n(t)\, \overline{y}_n^*(\epsilon)\right] \tag{33}$$

where (32) and (33) are $f(\boldsymbol{Q}(t))$ and $h(\boldsymbol{Q}(t))$ (see (23), (24)) evaluated under policy $\mathsf{RandRR}_\epsilon^*$ and σ^*, respectively. The term g_ϵ^* is defined as $g_\epsilon^* = \sum_{n=1}^{N} g_n(\overline{y}_n^*(\epsilon))$.

Since the QRRNUM policy maximizes (25), computing (25) under both the QRRNUM policy and the joint policy $(\mathsf{RandRR}_\epsilon^*, \sigma^*)$ yields

$$\begin{aligned} &f_{\mathsf{QRRNUM}}(\boldsymbol{Q}(t_k)) + h_{\mathsf{QRRNUM}}(\boldsymbol{Q}(t_k)) \\ &\ge \mathbb{E}[T_{k+1} \mid \boldsymbol{Q}(t_k)] \frac{f_\epsilon^*(\boldsymbol{Q}(t_k)) + h_\epsilon^*(\boldsymbol{Q}(t_k))}{\mathbb{E}[T_\epsilon^*]} \\ &\overset{(a)}{=} \mathbb{E}[T_{k+1} \mid \boldsymbol{Q}(t_k)] \left[V g_\epsilon^* + \epsilon \sum_{n=1}^{N} Q_n(t_k)\right] \end{aligned}$$

[6] This property is proved in a similar case in [24, Chapter 5.5.2].
[7] The throughput $\overline{y}_n^*(\epsilon)$ is less than or equal to one. Thus it is a feasible choice of $r_n(t+\tau)$.

$$= \mathbb{E}\left[T_{k+1}\left(Vg_\epsilon^* + \epsilon\sum_{n=1}^{N} Q_n(t_k)\right) \mid \boldsymbol{Q}(t_k)\right], \tag{34}$$

where (a) is from (32) and (33). The inequality (22) under the QRRNUM policy in the $(k+1)$th round of transmission then satisfies

$$\begin{aligned}
&\Delta_{T_{k+1}}(\boldsymbol{Q}(t_k)) - V\mathbb{E}\left[\sum_{\tau=0}^{T_{k+1}-1}\sum_{n=1}^{N} g_n(r_n(t_k+\tau)) \mid \boldsymbol{Q}(t_k)\right] \\
&\overset{(a)}{\leq} B - f_{\mathsf{QRRNUM}}(\boldsymbol{Q}(t_k)) - h_{\mathsf{QRRNUM}}(\boldsymbol{Q}(t_k)) \\
&\overset{(b)}{\leq} B - \mathbb{E}\left[T_{k+1}\left(Vg_\epsilon^* + \epsilon\sum_{n=1}^{N} Q_n(t_k)\right) \mid \boldsymbol{Q}(t_k)\right],
\end{aligned} \tag{35}$$

where (a) is the inequality (22) under policy QRRNUM, and (b) uses (34). Taking expectation over $\boldsymbol{Q}(t_k)$ in (35) and summing it over $k \in \{0, \ldots, K-1\}$, we get

$$\begin{aligned}
&\mathbb{E}\left[L(\boldsymbol{Q}(t_K))\right] - \mathbb{E}\left[L(\boldsymbol{Q}(t_0))\right] - V\mathbb{E}\left[\sum_{\tau=0}^{t_K-1}\sum_{n=1}^{N} g_n(r_n(\tau))\right] \\
&\leq BK - Vg_\epsilon^*\,\mathbb{E}\left[t_K\right] - \epsilon\,\mathbb{E}\left[\sum_{k=0}^{K-1}\left(T_{k+1}\sum_{n=1}^{N} Q_n(t_k)\right)\right].
\end{aligned} \tag{36}$$

Since queue backlogs $(Q_1(\cdot), \ldots, Q_N(\cdot))$ and $L(\boldsymbol{Q}(\cdot))$ are all nonnegative, and by assumption $\boldsymbol{Q}(t_0) = \boldsymbol{Q}(0) = \mathbf{0}$, ignoring all backlog-related terms in (36) yields

$$\begin{aligned}
-V\mathbb{E}\left[\sum_{\tau=0}^{t_K-1}\sum_{n=1}^{N} g_n(r_n(\tau))\right] &\leq BK - Vg_\epsilon^*\,\mathbb{E}\left[t_K\right] \\
&\overset{(a)}{\leq} B\mathbb{E}\left[t_K\right] - Vg_\epsilon^*\,\mathbb{E}\left[t_K\right]
\end{aligned} \tag{37}$$

where (a) uses $t_K = \sum_{k=1}^{K} T_k \geq K$. Dividing (37) by V and rearranging terms, we get

$$\mathbb{E}\left[\sum_{\tau=0}^{t_K-1}\sum_{n=1}^{N} g_n(r_n(\tau))\right] \geq \left(g_\epsilon^* - \frac{B}{V}\right)\mathbb{E}\left[t_K\right]. \tag{38}$$

Recall from Section IV-C that B in (38) is an unspecified constant satisfying $B \geq N\mathbb{E}\left[T_k^2 \mid \boldsymbol{Q}(t)\right]$ for all $k \in \mathbb{N}$ and all possible $\boldsymbol{Q}(t)$. From Lemma 5, it suffices to define $B \triangleq N\mathbb{E}\left[T_{\max}^2\right]$.

Under the QRRNUM policy, let $K(t)$ be the number of transmission rounds ending by time t. We have $t_{K(t)+1} > t$. The expected sum utility over the first t slots satisfies

$$\begin{aligned}
&\sum_{\tau=0}^{t-1}\sum_{n=1}^{N}\mathbb{E}\left[g_n(r_n(\tau))\right] = \mathbb{E}\left[\sum_{\tau=0}^{t_{K(t)+1}-1}\sum_{n=1}^{N} g_n(r_n(\tau))\right] \\
&- \mathbb{E}\left[\sum_{\tau=t}^{t_{K(t)+1}-1}\sum_{n=1}^{N} g_n(r_n(\tau))\right] \\
&\overset{(b)}{\geq} \left[g_\epsilon^* - \frac{B}{V}\right]\mathbb{E}\left[t_{K(t)+1}\right] - \mathbb{E}\left[t_{K(t)+1} - t\right] g_{\max} \\
&= \left[g_\epsilon^* - \frac{B}{V}\right]t + \left[g_\epsilon^* - \frac{B}{V} - g_{\max}\right]\mathbb{E}\left[t_{K(t)+1} - t\right] \\
&\overset{(c)}{\geq} \left[g_\epsilon^* - \frac{B}{V}\right]t,
\end{aligned} \tag{39}$$

where we define $g_{\max} \triangleq \sum_{n=1}^{N} g_n(1) < \infty$ as an upper bound on the sum utility (since $g_n(\cdot)$ are nondecreasing), (b) follows (38), and (c) follows $g_\epsilon^* \leq g_{\max}$. Taking a limiting time average of (39) yields

$$\liminf_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1}\sum_{n=1}^{N}\mathbb{E}\left[g_n(r_n(\tau))\right] \geq g_\epsilon^* - \frac{B}{V}. \tag{40}$$

Using Jensen's inequality and concavity of $g_n(\cdot)$, we get

$$\liminf_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1}\sum_{n=1}^{N}\mathbb{E}\left[g_n(r_n(\tau))\right] \leq \liminf_{t\to\infty}\sum_{n=1}^{N} g_n\left(\overline{r}_n^{(t)}\right), \tag{41}$$

where we define

$$\overline{r}_n^{(t)} \triangleq \frac{1}{t}\sum_{\tau=0}^{t-1}\mathbb{E}\left[r_n(\tau)\right].$$

Combining (40) and (41) yields

$$\liminf_{t\to\infty}\sum_{n=1}^{N} g_n\left(\overline{r}_n^{(t)}\right) \geq g_\epsilon^* - \frac{B}{V},$$

which holds for any sufficiently small ϵ. Passing $\epsilon \to 0$ yields

$$\liminf_{t\to\infty}\sum_{n=1}^{N} g_n\left(\overline{r}_n^{(t)}\right) \geq g^* - \frac{B}{V},$$

where g^* is the optimal utility in the optimization problem (2) (we have $g_\epsilon^* \to g^*$ as $\epsilon \to 0$ because $g_n(\cdot)$ are continuous). It remains to show that the network is stable.

To prove network stability, ignoring the first, second, and fifth term in (36) yields

$$\begin{aligned}
&\epsilon\,\mathbb{E}\left[\sum_{k=0}^{K-1}\left(T_{k+1}\sum_{n=1}^{N} Q_n(t_k)\right)\right] \\
&\leq BK + V\mathbb{E}\left[\sum_{\tau=0}^{t_K-1}\sum_{n=1}^{N} g_n(r_n(\tau))\right] \\
&\overset{(a)}{\leq} K\big(B + Vg_{\max}\mathbb{E}\left[T_{\max}\right]\big)
\end{aligned} \tag{42}$$

where (a) uses

$$\sum_{n=1}^{N} g_n(r_n(\tau)) \leq g_{\max}, \quad \mathbb{E}\left[t_K\right] = \sum_{k=1}^{K}\mathbb{E}\left[T_k\right] \leq K\mathbb{E}\left[T_{\max}\right].$$

Dividing (42) by $K\epsilon$, taking a $\limsup$ as $K \to \infty$, and using $T_{k+1} \geq 1$, we get

$$\limsup_{K\to\infty}\frac{1}{K}\mathbb{E}\left[\sum_{k=0}^{K-1}\sum_{n=1}^{N} Q_n(t_k)\right] \leq \frac{B + Vg_{\max}\mathbb{E}\left[T_{\max}\right]}{\epsilon} < \infty. \tag{43}$$

We note that (43) holds for all $\epsilon > 0$ such that the $\epsilon\mathbf{1} \in \Lambda_{\text{int}}$, where $\mathbf{1}$ is the all-one vector. Different values of ϵ do not alter the system dynamics under the QRRNUM policy, but only affect the finite upper bound on the average backlog.

Equation (43) shows that the average backlog is bounded when sampled at time instants $\{t_k\}_{k=0}^{\infty}$. This property is

enough to conclude that the average backlog over the whole time horizon is bounded, i.e., inequality (6) holds and the network is stable. It is because the length of each transmission round T_k has a finite second moment and the maximal amount of data admitted to each user in every slot is at most 1; see [12, Lemma 13] for a detailed proof. ∎

APPENDIX C

Proof of Lemma 5: The random variable T_k is the duration of serving a subset of channels in the kth round of the QRRNUM policy. Define $T_{\boldsymbol{\phi}}$ as the duration of serving the channels in $\boldsymbol{\phi} \in \Phi$ for one round in the QRRNUM policy. Then, it suffices to show, for every $\boldsymbol{\phi} \in \Phi$,

(1) $T_{\max} \geq_{\text{st}} T_{\boldsymbol{\phi}}$.
(2) $T_{\max}^2 \geq_{\text{st}} T_{\boldsymbol{\phi}}^2$.
(3) $\mathbb{E}[T_{\boldsymbol{\phi}}] \leq \mathbb{E}[T_{\max}] < \infty$, $\mathbb{E}\left[T_{\boldsymbol{\phi}}^2\right] \leq \mathbb{E}\left[T_{\max}^2\right] < \infty$.

When the system is idle in the kth round, i.e., $T_k = 1$, all the above inequalities naturally hold. Given a vector $\boldsymbol{\phi} \in \Phi$, from Theorem 1 we have $T_{\boldsymbol{\phi}} = \sum_{n:\phi_n=1} L_n^{\boldsymbol{\phi}}$ where the random variable $L_n^{\boldsymbol{\phi}}$ is defined in (9). We also have

$$T_{\max} = \sum_{n=1}^{N} L_n^{\mathbf{1}}, \tag{44}$$

where $L_n^{\mathbf{1}}$ is a special case of $L_n^{\boldsymbol{\phi}}$ with $\boldsymbol{\phi} = \mathbf{1}$.

Next we show that

$$L_n^{\mathbf{1}} \geq_{\text{st}} L_n^{\boldsymbol{\phi}}, \quad \text{for all } n \text{ such that } \phi_n = 1. \tag{45}$$

Since $L_n^{\mathbf{1}}$ and $L_n^{\boldsymbol{\phi}}$ are both at least one, we have

$$\Pr\left[L_n^{\boldsymbol{\phi}} > 0\right] = 1 = \Pr\left[L_n^{\mathbf{1}} > 0\right]. \tag{46}$$

From Theorem 1, for every integer $m \geq 1$, we have

$$\begin{aligned} \Pr\left[L_n^{\boldsymbol{\phi}} > m\right] &= \mathsf{P}_{n,01}^{(M(\boldsymbol{\phi}))} \mathsf{P}_{n,11}^{(m-1)} \\ &\leq \mathsf{P}_{n,01}^{(N)} \mathsf{P}_{n,11}^{(m-1)} = \Pr\left[L_n^{\mathbf{1}} > m\right]. \end{aligned} \tag{47}$$

The inequality in (47) follows from the fact that, for positively correlated channels, the k-step transition probability $\mathsf{P}_{n,01}^{(k)}$ is increasing in k. Combining (46) and (47) proves (45).

Next we show, for a given $\boldsymbol{\phi} \in \Phi$,

$$\sum_{n:\phi_n=1} L_n^{\mathbf{1}} \geq_{\text{st}} \sum_{n:\phi_n=1} L_n^{\boldsymbol{\phi}}. \tag{48}$$

If (48) holds, then for any $m \in \{0, 1, 2, \ldots\}$,

$$\begin{aligned} \Pr[T_{\max} > m] &\overset{(a)}{\geq} \Pr\left[\sum_{n:\phi_n=1} L_n^{\mathbf{1}} > m\right] \\ &\overset{(b)}{\geq} \Pr\left[\sum_{n:\phi_n=1} L_n^{\boldsymbol{\phi}} > m\right] = \Pr[T_{\boldsymbol{\phi}} > m], \end{aligned}$$

where (a) follows $T_{\max} \geq \sum_{n:\phi_n=1} L_n^{\mathbf{1}}$ and (b) uses (48). Thus we have $T_{\max} \geq_{\text{st}} T_{\boldsymbol{\phi}}$, proving the first part of the lemma. Inequality (48) can be shown by iteratively applying Lemma 6 presented later to the random variables $\{L_n^{\mathbf{1}}\}_{n:\phi_n=1}$ and $\{L_n^{\boldsymbol{\phi}}\}_{n:\phi_n=1}$; these random variables are mutually independent by Theorem 1.

Next we show $T_{\max}^2 \geq_{\text{st}} T_{\boldsymbol{\phi}}^2$ for every $\boldsymbol{\phi} \in \Phi$. For every $m \in \{0, 1, 2, \ldots\}$,

$$\begin{aligned} \Pr\left[T_{\max}^2 > m\right] &= \Pr\left[T_{\max} > \sqrt{m}\right] \\ &\overset{(a)}{=} \Pr\left[T_{\max} > \lfloor\sqrt{m}\rfloor\right] \\ &\overset{(b)}{\geq} \Pr\left[T_{\boldsymbol{\phi}} > \lfloor\sqrt{m}\rfloor\right] \overset{(c)}{=} \Pr\left[T_{\boldsymbol{\phi}}^2 > m\right], \end{aligned}$$

where (a)(c) are because $T_{\max}$ and $T_{\boldsymbol{\phi}}$ are integer-valued, and (b) follows $T_{\max} \geq_{\text{st}} T_{\boldsymbol{\phi}}$.

Finally, that $\mathbb{E}[T_{\boldsymbol{\phi}}] \leq \mathbb{E}[T_{\max}]$ and $\mathbb{E}\left[T_{\boldsymbol{\phi}}^2\right] \leq \mathbb{E}\left[T_{\max}^2\right]$ follows directly from the first two results [15, Lemma 9.1.1]. The finiteness of $\mathbb{E}[T_{\max}]$ and $\mathbb{E}\left[T_{\max}^2\right]$ can be easily verified using (44) and Theorem 1. ∎

Lemma 6. Consider four positive integer-valued random variables X_1, X_2, Y_1, and Y_2. Suppose they are mutually independent, and $X_n \geq_{\text{st}} Y_n$ for $n \in \{1, 2\}$. Then $X_1 + X_2 \geq_{\text{st}} Y_1 + Y_2$.

Proof of Lemma 6: Since all four random variables are positive, for $m \in \{0, 1\}$,

$$\Pr[X_1 + X_2 > m] = 1 = \Pr[Y_1 + Y_2 > m]. \tag{49}$$

For every integer $m \geq 2$,

$$\begin{aligned} &\Pr[X_1 + X_2 > m] \\ &= \sum_{a=1}^{m} \Pr[X_1 + X_2 > m \mid X_2 = a] \Pr[X_2 = a] \\ &= \sum_{a=1}^{m} \Pr[X_1 > m - a] \Pr[X_2 = a] \\ &\overset{(a)}{\geq} \sum_{a=1}^{m} \Pr[Y_1 > m - a] \Pr[X_2 = a] \\ &= \Pr[Y_1 + X_2 > m], \end{aligned}$$

where (a) follows $X_1 \geq_{\text{st}} Y_1$. Likewise, from $X_2 \geq_{\text{st}} Y_2$ we have

$$\Pr[Y_1 + X_2 > m] \geq \Pr[Y_1 + Y_2 > m], \quad m \geq 2.$$

Hence,

$$\begin{aligned} \Pr[X_1 + X_2 > m] &\geq \Pr[Y_1 + X_2 > m] \\ &\geq \Pr[Y_1 + Y_2 > m], \quad m \geq 2. \end{aligned} \tag{50}$$

Combining (49) and (50) completes the proof. ∎