Scheduling in Parallel Queues with Randomly Varying Connectivity and Switchover Delay

Güner D. Celik, Long B. Le and Eytan Modiano
Laboratory for Information and Decision Systems
Massachusetts Institute of Technology

Abstract—We consider a dynamic server control problem for two parallel queues with *randomly varying connectivity* and *server switchover delay between the queues*. At each time slot the server decides either to stay with the current queue or switch to the other queue based on the current connectivity and the queue length information. The introduction of switchover time is a new modeling component of this problem, which makes the problem much more challenging. We develop a novel approach to characterize the stability region of the system by using *state-action frequencies*, which are stationary solutions to a Markov Decision Process (MDP) formulation of the corresponding saturated system. We characterize the stability region explicitly in terms of the connectivity parameters and develop a frame-based dynamic control (FBDC) policy that is shown to be throughput-optimal. In fact, the FBDC policy provides *a new framework for developing throughput-optimal network control policies* using state-action frequencies. Furthermore, we develop simple *Myopic policies* that achieve more than 96% of the stability region. Finally, simulation results show that the Myopic policies may achieve the full stability region and are more delay efficient than the FBDC policy in most cases.

I. INTRODUCTION

Scheduling a dynamic server over randomly varying wireless channels is an important and well-studied problem which provides useful mathematical models for many practical applications [8], [10], [11], [13]. However, to the best of our knowledge, the joint effects of *randomly varying connectivity* and *server switchover delay* have not been considered before. In fact, switchover delay is a widespread phenomenon that can be observed in many practical dynamic control systems. In satellite systems where a mechanically steered antenna is providing service to ground stations, the time to switch from one station to another can be around 10ms [3]. Similarly, the delay for electronic beamforming can be on the order of $10\mu s$ in wireless radio systems [3]. Furthermore, in optical communication systems tuning delay for transceivers can take significant time (μs-ms) [4].

We consider the dynamic server control problem for two parallel queues with randomly varying connectivity and server switchover delay as shown in Fig. 1. We consider a slotted system where the slot length is equal to a single packet transmission time and it takes one slot for the server to switch from one queue to the other[1]. packet is successfully received

---

[1]In a slotted system, even a minimal switchover delay will lead to a loss of a slot due to synchronization issues.
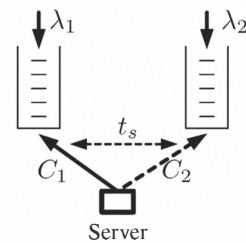
Fig. 1: System model. Two queues with randomly varying connectivities (C_1 and C_2) and $t_s = 1$ slot switchover time. The server is currently connected to queue 1 and it takes 1 time slot to switch to queue 2.

from queue i if the server is currently at queue i, it decides to stay at queue i, and queue i is connected. The server dynamically decides to stay with the current queue or switch to the other queue based on the connectivity and the queue length information. Our goal is to study the impact of the switchover time on system stability and optimal algorithms. We show that as compared to the system without switchover delay in [11], the stability region is significantly reduced and the optimal policies take a different structure.

Optimal control of queuing systems and communication networks has been a very active research topic over the past two decades (e.g., [5], [8], [10], [11]). In the seminal papers [10] and [11], Tassiulas and Ephremides characterized the stability region of multihop wireless networks and parallel queues with randomly varying connectivity. Later, these results were extended to power allocation and routing, and delayed or limited channel state information (e.g., [1], [5], [8], [13]). These works do not consider the server switchover times. While switchover delay has been studied in polling models in the queuing theory community (e.g., [2], [12]) and in optical networks in [4], random connectivity was not considered.

The main contribution of this paper is solving the scheduling problem in parallel queues with *randomly varying connectivity* and *server switchover times* for the first time. The paper provides *a novel framework for establishing throughput-optimality in network control problems* using the state-action frequencies. In particular, we explicitly characterize the stability region of the system using the *state-action frequencies*. We develop a throughput-optimal dynamic control policy that is applicable to systems with more than two queues, arbitrary switchover times and general Markovian channels. Finally, we propose simple and delay-efficient Myopic policies that provably achieve almost the full stability region.

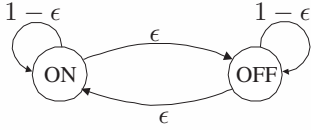This paper is organized as follows. We introduce the system

Fig. 2: Markov modulated ON/OFF channel process. We have $\epsilon \leq 0.5$ for positive correlation.

model in Section II. The stability region characterization for both uncorrelated and correlated channels are in Section III. We prove the throughput-optimality of the FBDC policy in Section IV, and analyze simple Myopic policies in Section V. For brevity, all proofs are omitted and can be found in [6].

II. SYSTEM MODEL

Consider unit-length time slots that are equal to one packet transmission time; $t \in \{0, 1, 2, ...\}$. It takes one slot for the server to switch from one queue to the other. Let $m(t)$ denote the queue at which the server is present at slot t. Let the arrival process $A_i(t)$ with average arrival rate λ_i denote the number of packets arriving to queue i at time slot t, where $A_i(t)$ is independently and identically distributed (i.i.d.) over time slots and $\mathbb{E}[A_i^2(t)] \leq A_{\max}^2$, $i \in \{1, 2\}$. Let $\mathbf{C}(t) = (C_1(t), C_2(t))$ be the channel (connectivity) process at time slot t, where $C_i(t) = 0$ for the OFF state (disconnected) and $C_i(t) = 1$ for the ON state (connected). We assume that the processes $A_1(t), A_2(t), C_1(t)$ and $C_2(t)$ are independent.

The process $C_i(t), i \in \{1, 2\}$, is assumed to form the two-state Markov chain with transition probability $\epsilon \leq 0.5$ as shown in Fig. 2, i.e., the symmetric Gilbert-Elliot (G-E) channel model [1], [14]. The steady state probability of each channel state is equal to 0.5 in this model. Moreover, for $\epsilon = 0.5$, $C_i(t)$ is i.i.d. over time slots and takes the value 1 w.p. 0.5. We refer to this case as *uncorrelated channels*. Our results and algorithms are applicable to non-symmetric channel models, but here we present the symmetric case for ease of exposition. Let $\mathbf{Q}(t) = (Q_1(t), Q_2(t))$ be the queue lengths at time slot t. We assume that $\mathbf{Q}(t)$ and $\mathbf{C}(t)$ are known to the server at the beginning of each time slot. Let $a_t \in \{0, 1\}$ denote the action taken at slot t, where $a_t = 1$ if the server stays with the current queue and $a_t = 0$ otherwise. One packet is successfully received from queue i at time slot t, if $m(t) = i, a_t = 1$ and $C_i(t) = 1$.

A queue is called stable if $\limsup_{t \to \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}[Q(\tau)] < \infty$ [8]. The system is called stable if both queues are stable. The stability region $\mathbf{\Lambda}$ is the set of all arrival rate vectors $\mathbf{\lambda} = (\lambda_1, \lambda_2)$ such that there exists a control algorithm that stabilizes the system. The δ-stripped stability region is defined for some $\delta > 0$ as $\mathbf{\Lambda}^\delta \triangleq \left\{ (\lambda_1, \lambda_2) | (\lambda_1 + \delta, \lambda_2 + \delta) \in \mathbf{\Lambda} \right\}$. A policy is said to achieve γ-fraction of $\mathbf{\Lambda}$, if it stabilizes the system for all input rates inside $\gamma \mathbf{\Lambda}$.

III. STABILITY REGION

We explicitly characterize the stability region for both uncorrelated and correlated channels and show that channel correlation can be exploited to enlarge the stability region significantly.
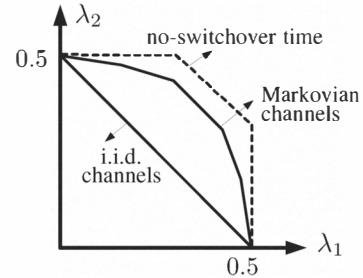

Fig. 3: Stability region under uncorrelated (i.i.d.) and correlated (Markovian) channels with and without switchover time.

A. Motivation-Uncorrelated Channels

We start by considering the case that the channel processes are uncorrelated over time (i.e., $\epsilon = 0.5$). The stability region of the corresponding system with no-switchover time was established in [11]: $\lambda_1, \lambda_2 \in [0, 0.5]$ and $\lambda_1 + \lambda_2 \leq 0.75$. When the switchover time is zero, the stability region is the same for both i.i.d. and Markovian channels, which is a special case of the results in [8]. However, when the switchover time is non-zero, the stability region is considerably reduced.

Theorem 1: For the system with i.i.d. channels

$$\mathbf{\Lambda} = \{(\lambda_1, \lambda_2) | \lambda_1 + \lambda_2 \leq 0.5, \lambda_1, \lambda_2 \geq 0\}. \quad (1)$$

In addition, the simple Exhaustive policy is throughput-optimal. The basic idea is that as soon as the server switches to queue i, the time to ON state is a geometric random variable of mean 2 slots, which can be viewed as the service time. The Exhaustive policy is throughput-optimal because under this policy, as the arrival rates increase, the fraction of time the server spends on switching decreases [12].

This reduced stability region is depicted in Fig. 3. When the channels are always connected, the stability region is $\lambda_1, \lambda_2 \geq 0, \lambda_1 + \lambda_2 \leq 1$ and is *not* affected by the switchover delay [12]. It is the *combination of random connectivity and switchover delay* that results in fundamental changes in stability.

Remark 1: These results hold for general systems with N queues, *arbitrary* switchover times and i.i.d. channels with probabilities p_i [6]: $\mathbf{\Lambda} = \{\mathbf{\lambda} \geq 0 | \sum_{i=1}^N \lambda_i / p_i \leq 1\}$.

B. Correlated Channels

When switchover times are non-zero, channel correlation can be exploited to improve the stability region considerably. Moreover, as $\epsilon \to 0$, the stability region tends to that achieved by the system with no-switchover time and for $0 < \epsilon < 0.5$ the stability region lies between the stability regions corresponding to the two extreme cases $\epsilon = 0.5$ and $\epsilon \to 0$ as shown in Fig. 3.

We start by analyzing the corresponding system with saturated queues, i.e., both queues are always non-empty. Let $\mathbf{\Lambda}_s$ denote the set of all time average expected departure rates that can be obtained from the two queues in the saturated system under all possible policies. We will show that $\mathbf{\Lambda} = \mathbf{\Lambda}_s$. We prove the necessary stability conditions in the following Lemma and establish sufficiency in the next section.

Lemma 1:

$$\mathbf{\Lambda} \subseteq \mathbf{\Lambda}_s.$$

317

We omit the proof for brevity but an intuitive explanation is as follows. Given a policy π for the system with random arrivals, apply the same policy in the saturated system with *the same sample path of channel realizations*. It is clear that the total number of departures from each queue in the saturated system is no less than that in the system with random arrivals.

Next, we establish the region Λ_s by formulating the system dynamics as a Markov Decision Process (MDP). Let $\mathbf{s}_t = (m(t), C_1(t), C_2(t)) \in S$ and $a_t \in A = \{0, 1\}$ denote the system state and the action taken at time t where S is the set of all states and A is the set of all actions at each state. For the saturated system, a policy is a mapping from the history of the channel processes until time t to the set of all probability distributions on actions $a_t \in \{0, 1\}$. A *stationary* policy is a policy that depends only on the current state. In each time slot t, the server observes the current state $\mathbf{s}_t$ and chooses an action a_t. Then the next state j is realized according to the transition probabilities $\mathbf{P}(j|\mathbf{s}, a)$ which depend on the random channel processes. Now, we define the reward functions as follows:

$$r_1(\mathbf{s}, a) \triangleq 1 \text{ if } \mathbf{s} = (1, 1, 1) \text{ or } \mathbf{s} = (1, 1, 0), \text{ and } a = 1 \quad (2)$$
$$r_2(\mathbf{s}, a) \triangleq 1 \text{ if } \mathbf{s} = (2, 1, 1) \text{ or } \mathbf{s} = (2, 0, 1), \text{ and } a = 1, \quad (3)$$

and $r_1(\mathbf{s}, a) = r_2(\mathbf{s}, a) \triangleq 0$ otherwise. Given some $\alpha_1, \alpha_2 \geq 0$, define the system reward at time t as $r(\mathbf{s}_t, a_t) \triangleq \alpha_1 r_1(\mathbf{s}_t, a_t) + \alpha_2 r_2(\mathbf{s}_t, a_t)$. The average reward of policy π is defined as

$$r^\pi \triangleq \lim_{K \to \infty} \frac{1}{K} E\left\{ \sum_{t=1}^{K} r(\mathbf{s}_t, a_t^\pi) \right\}.$$

We are interested in the policy that achieves the maximum time average expected reward $r^* \triangleq \max_\pi r^\pi$. This optimization problem is a discrete time MDP characterized by the state transition probabilities $\mathbf{P}(j|\mathbf{s}, a)$ with 8 states, 2 actions per state and bounded rewards. Furthermore, any given pair of states are accessible from each other (i.e., there exists a positive probability path between the states) under some stationary deterministic policy. Therefore this MDP belongs to the class of *Weakly Communicating* MDPs [9] [2]. Therefore, there exists a stationary deterministic optimal policy independent of the initial state [9]. The feasible region of the Dual Linear Programming approach to solving Dynamic Programs is called the *state-action polytope*, $\mathbf{X}$, that is the set of state-action frequency vectors $\mathbf{x}$ that satisfy the balance equations

$$x(\mathbf{s}; 1) + x(\mathbf{s}; 0) = \sum_{\mathbf{s}'} \sum_{a \in \{0, 1\}} \mathbf{P}(\mathbf{s}|\mathbf{s}', a) x(\mathbf{s}'; a), \ \forall \mathbf{s} \in S, \quad (4)$$

the normalization condition $\sum_{\mathbf{s}} x(\mathbf{s}; 1) + x(\mathbf{s}; 0) = 1$, and the nonnegativity constraints $x(\mathbf{s}; a) \geq 0$, for $\mathbf{s} \in S, a \in A$. Note that $x(\mathbf{s}; 1)$ can be interpreted as the stationary probability that action *stay* is taken at state $\mathbf{s}$. More precisely, a point $\mathbf{x} \in \mathbf{X}$ corresponds to a randomized stationary policy that takes action

[2]In fact, rather than the trivial suboptimal policy π_s that decides to stay with the current queue in all states, all stationry deterministic policies are unichain, namely, they have a single recurrent class regardless of the initial state. Hence, when π_s is excluded, we have a Unichain MDP.

$a \in \{0, 1\}$ at state $\mathbf{s}$ w.p.

$$\mathbf{P}(action \ a \text{ at state } \mathbf{s}) = \frac{x(\mathbf{s}; a)}{x(\mathbf{s}; 1) + x(\mathbf{s}; 0)}, a \in A, \mathbf{s} \in S_x, \quad (5)$$

where S_x is the set of recurrent states given by $S_x \equiv \{\mathbf{s} \in S : x(\mathbf{s}; 1) + x(\mathbf{s}; 0) > 0\}$, and actions are arbitrary for transient states $\mathbf{s} \in S/S_x$ [7], [9]. Furthermore, every policy has a corresponding limiting average state-action frequency vector in $\mathbf{X}$ regardless of the initial state and every $\mathbf{x} \in \mathbf{X}$ can be achieved by a stationary randomized policy as in (5) [7], [9].

The following linear transformation of the state-action polytope $\mathbf{X}$ defines the *reward polytope* [7]: $\{(\bar{r}_1, \bar{r}_2) | \bar{r}_1 = \mathbf{x}.\mathbf{r}_1, \bar{r}_2 = \mathbf{x}.\mathbf{r}_2, \mathbf{x} \in \mathbf{X}\}$, where $(.)$ denotes the vector inner product and $\mathbf{r}_1$ and $\mathbf{r}_2$ are the vector of rewards defined in (2) and (3). This polytope is the set of all time average expected departure rate pairs that can be obtained in the saturated system, i.e., it is the rate region Λ_s. Algorithm 1 explicitly characterizes Λ_s.

Algorithm 1 *Stability Region Characterization*

1: Given $\alpha_1, \alpha_2 \geq 0$, solve the following Linear Program (LP)

$$\max_{\mathbf{x}} \quad \alpha_1 \bar{r}_1 + \alpha_2 \bar{r}_2$$
$$\text{subject to} \quad \mathbf{x} \in \mathbf{X}. \quad (6)$$

2: For a given α_2/α_1 ratio, the optimal solution $(\bar{r}_1^*, \bar{r}_2^*)$ of the LP in (6) gives one of the corner points of $\Lambda_\mathbf{s}$. Find all possible corner points and take their convex combination.

Lemma 2 is useful for finding the solution of (6), [7], [9].

Lemma 2: The corner points of $\mathbf{X}$ have a one-to-one corresspondence with the stationary deterministic policies.
The intuition behind this lemma is as follows. If $\mathbf{x}$ is a corner point of $\mathbf{X}$, it cannot be expressed as a convex combination of any two other elements in $\mathbf{X}$, therefore, for each state $\mathbf{s}$ only one action has a nonzero probability. Therefore, *the corners of the rate polytope $\Lambda_\mathbf{s}$ are given by stationary deterministic policies*. There are a total of 2^8 stationary deterministic policies since we have 8 states and 2 actions per state. Hence, finding the rate pairs corresponding to the 256 deterministic policies and taking their convex combination gives Λ_s. Fortunately, we do not have to go through this tedious procedure. The fact that at a vertex of (6) either $x(\mathbf{s}; 1)$ or $x(\mathbf{s}; 0)$ has to be zero for each $\mathbf{s} \in \mathbf{S}$ provides a useful guideline for analytically solving this LP. The solution is lengthy and is omitted here.

We present the solution of this LP for all (α_1, α_2) pairs for $\epsilon \geq \epsilon_c \triangleq 1 - \sqrt{2}/2$ in Fig. 4 (a) and for $\epsilon < \epsilon_c$ in Fig. 4 (b). As $\epsilon \to 0.5$, the stability region converges to that of the system with i.i.d. channels with ON probability equal to 0.5. In this regime, knowledge of the current channel state is of no value. As $\epsilon \to 0$ the stability region converges to that for the system with no-switchover time in [11]. In this regime, the channels are likely to stay the same in many consecutive time slots, therefore, the effect of switching delay is negligible.

Remark 2: The technique used for characterizing the stability region in terms of the state-action frequencies can be used
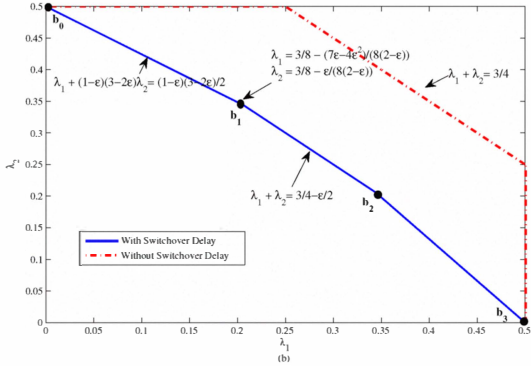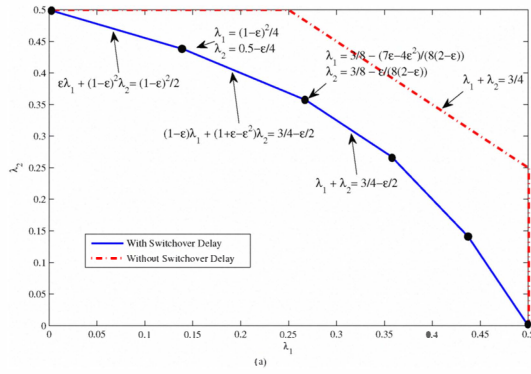
Fig. 4: Stability region under correlated channels with and without switchover time for (a) $\epsilon = 0.25 < \epsilon_c = 1 - \sqrt{2}/2$ and (b) $\epsilon = 0.40 \geq \epsilon_c$.

corner b_3	corner b_2	corner b_1	corner b_0
(1,1,1): stay	(1,1,1): stay	(1,1,1): stay	(1,1,1): switch
(1,1,0): stay	(1,1,0): stay	(1,1,0): stay	(1,1,0): switch
(1,0,1): stay	(1,0,1): switch	(1,0,1): switch	(1,0,1): switch
(1,0,0): stay	(1,0,0): switch	(1,0,0): switch	(1,0,0): switch
(2,1,1): switch	(2,1,1): stay	(2,1,1): stay	(2,1,1): stay
(2,1,0): switch	(2,1,0): switch	(2,1,0): switch	(2,1,0): stay
(2,0,1): switch	(2,0,1): stay	(2,0,1): stay	(2,0,1): stay
(2,0,0): switch	(2,0,0): switch	(2,0,0): stay	(2,0,0): stay

TABLE I: Mapping from the queue sizes to the corners of $\boldsymbol{\Lambda_s}$, b_0, b_1, b_2, b_3 shown in Fig. 4 (b), for $\epsilon \geq \epsilon_c$. For each state $\mathbf{s} = (m(t), C_1(t), C_2(t))$ the optimal action is specified. The thresholds on Q_2/Q_1 for the FBDC policy are $0, T_1 = 1/((1-\epsilon)(3-2\epsilon)), 1, T_2 = (1-\epsilon)(3-2\epsilon)$ and for the OLM policy are $0, T_1' = \epsilon/(1-\epsilon), 1, T_2' = (1-\epsilon)/\epsilon$. For example corner b_1 is chosen in the FBDC policy if $1 \leq Q_2/Q_1 < T_2$, whereas in the OLM policy if $1 \leq Q_2/Q_1 < T_2'$.

Theorem *2: The FBDC policy stabilizes the system as long as the arrival rates are within the $\delta(T)$-stripped stability region $\boldsymbol{\Lambda}_s^{\delta(T)}$, where $\delta(T)$ is a decreasing function of T.*

The proof is omitted for brevity. It involves a drift analysis using the standard quadratic Lyapunov function. However, *it is novel in utilizing an MDP framework in Lyapunov drift arguments*. The basic idea is that when the optimal policy solving (7), π^*, is applied over a sufficiently long frame of T slots, the average output rates of both the actual system and the corresponding saturated system converge to $\mathbf{r}^*$. For the saturated system, the probability of a large difference between the empirical rates and $\mathbf{r}^*$ is essentially due to the mixing time of the Markov chain induced by policy π^*, which decays exponentially fast in T [7]. Therefore, for sufficiently large queue lengths, the difference between the empirical rates in the actual system and r^* also decreases with T. This results in a negative Lyapunov drift when $\boldsymbol{\lambda}$ is inside the $\delta(T)$-stripped stability region since from (7) we have $\mathbf{Q}(jT).\mathbf{r}^* \geq \mathbf{Q}(jT).\boldsymbol{\lambda}$.

The parameter $\delta(T)$, capturing the difference between the stability region of the FBDC policy and $\boldsymbol{\Lambda}_s$, is related to the mixing time of the system Markov chain and is a decreasing function of T. This establishes that the FBDC policy is asymptotically throughput-optimal and that $\boldsymbol{\Lambda} = \boldsymbol{\Lambda_s}$. Moreover, $\delta(T)$ is negligible even for relatively small values of T.

The FBDC policy is easy to implement since it does not require the solution of the LP for each frame. Instead, we can solve the LP for all relevant $Q_1(t), Q_2(t)$ values only *once* in advance and create a mapping from the $Q_2(t)/Q_1(t)$ values to the corner points of the stability region. Then, we can use this mapping to find the corresponding optimal policy for each frame. Such a mapping depends only on the slopes of the lines in the stability region in Fig. 4 and is shown in Table I for the case of $\epsilon \geq \epsilon_c$. A similar mapping can be obtained for $\epsilon < \epsilon_c$.

Remark *3: The FBDC policy provides a new framework for developing throughput-optimal policies for network control problems* by solving an LP of state-action frequencies and applying this solution over a frame. In particular, *the FBDC policy can stabilize systems with more than two queues, arbitrary switchover times and complicated Markovian channels as well as most of the classical network control systems such as the ones in [1], [11] or [13].*

to find the stability regions of systems with arbitrary number of queues, switching times and channel states.

IV. FRAME-BASED DYNAMIC CONTROL (FBDC) POLICY

We propose a frame-based dynamic control policy inspired by the state-action frequencies and prove that it is throughput-optimal asymptotically in the frame length. The motivation behind the FBDC policy is that an optimal policy π^* that achieves the maximization in (6) for given weights α_1 and α_2 for the saturated system should achieve a *good* performance also in the original system when the queue sizes Q_1 and Q_2 are used as weights. Specifically, divide the time into equal-size intervals of T slots and let $Q_1(jT)$ and $Q_2(jT)$ be the queue lengths at the beginning of the jth interval. We find the deterministic policy that optimally solves (6) when $Q_1(jT)$ and $Q_2(jT)$ are used as weights and then apply this policy in each time slot of the frame. The FBDC policy is described in Algorithm 2 in details.

Algorithm 2 FBDC POLICY

1: Find the optimal solution to the following Linear Program

$$\max_{\{r_1, r_2\}} \quad Q_1(jT)r_1 + Q_2(jT)r_2$$
$$\text{subject to} \quad (r_1, r_2) \in \boldsymbol{\Lambda}_s \quad (7)$$

where $\boldsymbol{\Lambda}_s$ is the rate polytope derived in Section III-B.

2: The optimal solution (r_1^*, r_2^*) in step 1 is a corner point of $\boldsymbol{\Lambda}_s$ that corresponds to a stationary deterministic policy denoted by π^*. Apply π^* in each time slot of the frame.

V. Myopic Control Policies

We investigate the performance of simple *Myopic* policies that do not require the solution of an LP and achieve more than 96% of the stability region, while providing better delay performance than the FBDC policy in most cases. These policies make scheduling/switching decisions according to weight functions that are products of the queue lengths and the channel predictions for a small number of slots into the future. We refer to a Myopic policy considering k future time slots as the k-Lookahead Myopic policy. We implement these policies over frames of length T time slots where during the jth frame, the queue lengths at the beginning of the frame, $Q_1(jT)$ and $Q_2(jT)$, are used for weight calculations. The detailed description of the 1-Lookahead Myopic policy is given below.

Algorithm 3 1-LOOKAHEAD MYOPIC POLICY

1: Assuming that the server is currently with queue 1 and the system is at the jth frame, calculate the following weights in each time slot of the current frame;

$$W_1(t) = Q_1(jT)\Big(C_1(t) + \mathbb{E}\big[C_1(t+1)|C_1(t)\big]\Big)$$
$$W_2(t) = Q_2(jT)\mathbb{E}\big[C_2(t+1)|C_2(t)\big]. \qquad (8)$$

2: If $W_1(t) \geq W_2(t)$ stay with queue one, otherwise, switch to the other queue. A similar rule applies for queue 2.

Next we establish a lower bound on the stability region of the 1-Lookahead Myopic Policy by comparing its drift over a frame to the drift of the FBDC policy.

Theorem 3: The 1-Lookahead Myopic policy achieves at least γ-fraction of Λ asymptotically in T where $\gamma \geq 90\%$.
The proof is constructive and can be found in [6]. Here we highlight the key steps. The basic idea is that the 1-Lookahead Myopic (OLM) policy produces a mapping from the set of queue sizes to the corners of the stability region. This mapping is similar to that of the FBDC policy, however, the thresholds on the queue size ratios Q_2/Q_1 are determined according to (8): For $\epsilon \geq \epsilon_c$, there are 4 corners in the stability region denoted by b_0, b_1, b_2 and b_3 as shown in Fig. 4 (b). We derive conditions on Q_2/Q_1 such that the OLM policy chooses the deterministic actions that correspond to a given corner point. For instance, from Table I, the deterministic actions corresponding to corner b_1 are as follows: At queue 1, stay only if the channel states (C_1, C_2) are $(1,0)$ or $(1,1)$, and at queue 2, switch only if the channel states are $(1,0)$. The most limiting actions are switching at $(C_1, C_2) = (1,0)$ at queue 2, staying at $(C_1, C_2) = (1,1)$ at queue 1 and switching at $(C_1, C_2) = (0,0)$ at queue 1. The conditions on Q_2/Q_1 for the OLM policy to take these actions are $Q_1(1-\epsilon) > Q_2\epsilon$, $Q_1(2-\epsilon) > Q_2(1-\epsilon)$ and $Q_1 \leq Q_2$ respectively. Combining these and noting that since $\epsilon \geq \epsilon_c$ we have $\frac{1-\epsilon}{\epsilon} < \frac{2-\epsilon}{1-\epsilon}$, we obtain the intersection of all the conditions given by

$$1 \leq \frac{Q_2}{Q_1} < \frac{1-\epsilon}{\epsilon}.$$

We proceed similarly for the other corners and also for the case $\epsilon < \epsilon_c$. In Table I the shaded regions are the regions

of Q_2/Q_1 in which the OLM and the FBDC policies apply decisions for different corner points, denoted by $\mathbf{r}^{OLM}$ and $\mathbf{r}^*$. The following lemma proved in [6] completes the proof by establishing the 90% bound on the weighted average departure rate of the OLM policy w.r.t. to that of the FBDC policy.
Lemma 3:

$$\Psi \triangleq \frac{\sum_i Q_i(t)r_i^{OLM}}{\sum_i Q_i(t)r_i^*} \geq 90\%. \qquad (9)$$

The k-Lookahead Myopic Policy uses the following weight functions: Assuming the server is with queue 1 at time slot t,
$W_1(t) = Q_1(jT)\big(C_1(t) + \sum_{\tau=1}^{k} \mathbb{E}[C_1(t+\tau)|C_1(t)]\big)$
$W_2(t) = Q_2(jT) \sum_{\tau=1}^{k} \mathbb{E}[C_2(t+\tau)|C_2(t)]$.
A similar analysis shows that the *2-Lookahead Myopic Policy* achieves at least 94% of Λ, while the *3-Lookahead Myopic Policy* achieves at least 96% of Λ.

Simulation experiments in [6] suggest that the OLM policy may achieve the full stability region. For a frame size of $T=10$ and $\epsilon=0.4$ the average delay under the OLM policy is no more than that under the FBDC policy for 86% of all arrival rates considered, while this delay improvement is 81% of all arrival rates considered for $T = 25$ and $\epsilon = 0.25$. These results show that the OLM policy is not only simpler to implement than the FBDC policy, but it can also be more delay efficient.

References

[1] S. Ahmad, L. Mingyan, T. Javidi, Q. Zhao, and B. Krishnamachari, "Optimality of myopic sensing in multichannel opportunistic access," *IEEE Trans. Infor. Theory*, vol. 55, no. 9, pp. 4040-4050, Sept. 2009.

[2] E. Altman, P. Konstantopoulos, and Z. Liu, "Stability, monotonicity and invariant quantities in general polling systems," *Queuing Sys.*, vol. 11, pp. 35-57, Mar. 1992.

[3] L. Blake and M. Long, "Antennas: Fundamentals, design, measurement," *SciTech*, 2009.

[4] A. Brzezinski, "Scheduling algorithms for throughput maximization in data networks," Ph.D. thesis, MIT, 2007.

[5] L. B. Le, E. Modiano, C. Joo, and N. B. Shroff, "Longest-Queue-First scheduling under SINR interference model," In *Proc. ACM MobiHoc'10*, Sept. 2010, to appear.

[6] G. D. Celik, L. B. Le, and E. Modiano, "Scheduling in parallel queues with randomly varying connectivity and switchover delay," *Technical Report*, MIT, Jul. 2010 [Online]. Available: http://web.mit.edu/gcelik/Public

[7] S. Mannor and J. N. Tsitsiklis, "On the emperical state-action frequencies in Markov Decision Processes under general policies," *Mathematics of Operation Research*, vol. 30, no. 3, Aug. 2005.

[8] M. J. Neely, E. Modiano, and C. E. Rohrs, "Dynamic power allocation and routing for time varying wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 1, pp. 89–103, Jan. 2005.

[9] M. Puterman, "Markov Decision Processes: Discrete Stochastic Dynamic Programming," *Wiley*, 2005.

[10] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. Auto. Control*, vol. 37, no. 12, pp. 1936-1948, Dec. 1992.

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

[12] V. M. Vishnevskii and O. V. Semenova, "Mathematical methods to study the polling systems," *Auto. and Rem. Cont.*, vol. 67, no. 2, pp. 173-220, Feb. 2006.

[13] L. Ying, and S. Shakkottai, "On throughput optimality with delayed network-state information," In *Proc. Inform. Theory and Applic. Workshop*, Jan. 2008.

[14] M. Zorzi, R. Rao, and L. Milstein, "On the accuracy of a first-order Markov model for data transmission on fading channels," In *Proc. ICUPC'95*, 1995.