

Scheduling in Networks With Time-Varying Channels and Reconfiguration Delay

Güner D. Çelik, *Member, IEEE*, and Eytan Modiano, *Fellow, IEEE*

Abstract—We consider the optimal control problem for networks subjected to *time-varying channels, reconfiguration delays, and interference constraints*. We show that the simultaneous presence of time-varying channels and reconfiguration delays significantly reduces the system stability region and changes the structure of optimal policies. We first consider memoryless channel processes and characterize the stability region in closed form. We prove that a frame-based Max-Weight scheduling algorithm that sets frame durations dynamically, as a function of the current queue lengths and *average* channel gains, is throughput-optimal. Next, we consider arbitrary Markov-modulated channel processes and show that memory in the channel processes can be exploited to improve the stability region. We develop a novel approach to characterizing the stability region of such systems using *state-action frequencies*, which are stationary solutions to a Markov Decision Process (MDP) formulation. Moreover, we develop a dynamic control policy using the state-action frequencies and variable frames whose lengths are functions of queue sizes and show that it is throughput-optimal. The frame-based dynamic control (FBDC) policy is applicable to a broad class of network control systems, with or without reconfiguration delays, and provides a *new framework for developing throughput-optimal network control policies using state-action frequencies*. Finally, we propose Myopic policies that are easy to implement and have better delay properties as compared to the FBDC policy.

Index Terms—Markov decision process, queueing, reconfiguration delay, scheduling, switching delay, time-varying channels.

I. INTRODUCTION

SCHEDULING in wireless networks subject to interference constraints has been studied extensively over the past two decades [13], [14], [17], [30], [31], [34], [39], [40], [45]. However, to the best of our knowledge, the effects of *reconfiguration delays* have not been considered in the context of networks subject to *interference constraints* and *time-varying channel* conditions. Reconfiguration delay is a widespread phenomenon that is observed in many practical telecommunication systems [3], [6], [27], [46]. In satellite networks where multiple mechanically steered antennas are providing service to ground

Manuscript received October 07, 2012; revised July 25, 2013 and October 27, 2013; accepted November 10, 2013; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor U. Ayesta. Date of publication December 19, 2013; date of current version February 12, 2015. This work was supported by the NSF under Grants CNS-0626781 and CNS-0915988, the ARO MURI under Grant No. W911NF-08-1-0238, and the ONR under Grant N00014-12-1-0064. A preliminary version of this paper was presented at the IEEE International Conference on Computer Communications (INFOCOM), Orlando, FL, USA, March 25–20, 2012.

The authors are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: gcelik@mit.edu; modiano@mit.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TNET.2013.2292604

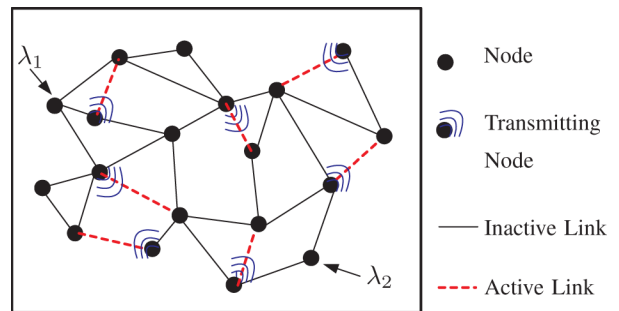


Fig. 1. System model. A single-hop wireless network with interference constraints, time-varying channels, and reconfiguration delays.

stations, the time to switch from one station to another can be around 10 ms [6], [41]. Similarly, in optical communication systems, laser tuning delay for transceivers and optical switching delay can take significant time ranging from microseconds to tens of milliseconds depending on technology [8], [27]. In wireless networks, delays for electronic beamforming or channel switching that occurs in phased-lock loops in oscillators can be more than 200 μ s [3], [6], [41], [46]. Worse yet, such small delay is often impossible to achieve due to delays incurred during different processing tasks such as channel estimation, signal-to-interference ratio, transmit diversity and power control calculations in the physical layer [3], [18], and stopping and restarting the interrupt service routines of various drivers in upper layers [3], [33]. Moreover, in various real-time implementations, channel switching delays from a few hundreds of microseconds to a few milliseconds have been observed [33], [42], [46].

We consider an optimal control problem for single-hop networks given by a graph structure $\mathcal{G}(\mathcal{N}, \mathcal{L})$ of nodes $n \in \mathcal{N}$ and links $\ell \in \mathcal{L}$, subject to reconfiguration delays, time-varying channels, and arbitrary interference constraints. For the time-varying channel states, we consider both independent and identically distributed (i.i.d.) or Markov modulated processes for which the structure of the stability region and the optimal policies differ significantly. Our system model can be used to abstract single-hop wireless networks as shown in Fig. 1, satellite networks with multiple satellite servers and ground stations as shown in Fig. 2, and optical switched networks [8], [34]. The network controller is to dynamically decide to stay with the current schedule of activations or to reconfigure to another schedule based on the channel process and the queue length information, where each decision to reconfigure leaves the network *idle* for an arbitrary but finite amount of time, corresponding to the *reconfiguration delay*. Note that in many networks, some nodes in the network may continue to function, while others are being

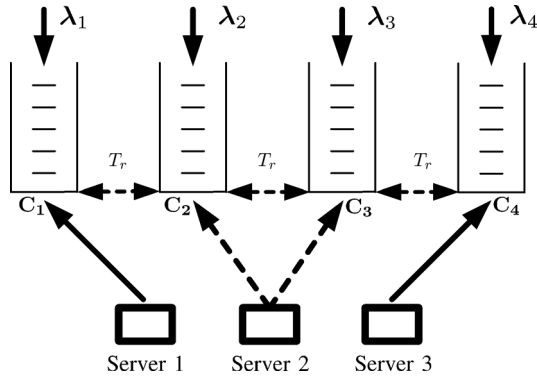


Fig. 2. Example 4×3 satellite network. Ground stations are subject to time-varying channels C_1, C_2, C_3, C_4 , and the servers are subject to T_r slot reconfiguration (switchover) delay. Server 2 is forced to be idle due to interference constraints.

reconfigured. Hence, assuming that all links are idle during network reconfiguration is a pessimistic assumption that may hold in some systems, such as optical switches, satellite transmitters, etc. However, in other cases, this assumption may be restrictive and provides a lower bound on performance. Our goal is to study the impact of reconfiguration delays on system stability and optimal algorithms. We show that, as compared to systems without reconfiguration delays [30], [31], [39], [40], the stability region can be significantly reduced, and that optimal policies take on a different structure.

We first consider the case of memoryless (i.i.d.) channel processes where we characterize the stability region in closed form as the convex hull of feasible activation vectors weighted by the *average* channel gain of each link. This result shows that in the presence of reconfiguration delays, it is not possible to opportunistically take advantage of the diversity in time-varying channels because the i.i.d. channel processes refresh during each reconfiguration interval. Moreover, we show that a class of Variable-size Frame-based Max-Weight (VFMW) algorithms that make scheduling decisions based on time-average channel gains and queue lengths stabilize the system by keeping the current schedule over a frame of duration that is a function of the queue lengths.

Next, we consider Markov modulated channel processes with memory and develop a *novel methodology* to characterize the stability region of the system using *state-action frequencies*, the steady-state solutions to a Markov Decision Process (MDP) formulation for the corresponding saturated system. We show that the stability region enlarges with the memory in the channel processes, which is in contrast to the case of no reconfiguration delays [17], [30], [40]. Furthermore, we develop a novel frame-based dynamic control (FBDC) policy based on the state-action frequencies that achieves the full stability region. To our knowledge, this is the first throughput-optimal scheduling algorithm for wireless networks with time-varying channels and reconfiguration delays. The state-action frequency approach and the FBDC policy are applicable to many network control systems as they provide a general framework that reduces stability region characterization and throughput-optimal algorithm development to solving linear programs (LPs). Finally, we consider Myopic policies that do not require the solution of an LP. Simulation results in Section IV-D suggest that the Myopic policies may in fact achieve the full stability region while providing

better delay performance than the FBDC policy for most arrival rates.

Scheduling in communication networks has been a very active research topic over the past two decades (e.g., [13], [15], [17], [24], [28], [30], [31], [36], [39], [40], [44], and [45]). In the seminal paper [39], Tassiulas and Ephremides characterized the stability region of wireless networks and proposed the throughput-optimal Max-Weight scheduling algorithm. The same authors considered a parallel queuing system with randomly varying connectivity in [40], where they characterized the stability region of the system explicitly and proved the throughput-optimality of the Longest-Connected-Queue scheduling policy. Later, these results were extended to joint power allocation and routing in wireless networks in [30] and [31], and optimal scheduling for switches in [26], [34], and [36]. Suboptimal distributed scheduling algorithms with throughput guarantees have been studied more recently in [10], [22], [24], and [44], while [15] and [28] developed distributed algorithms that achieve throughput optimality (for a detailed review, see [17] and [29]). Networks with delayed channel state information were considered in [21], [35] and [45], which showed that the stability region is reduced and that a policy similar to the Max-Weight algorithm is throughput-optimal. Dynamic server allocation over parallel queues with time-varying channels and limited channel sensing was considered in [1], [23], and [47]. These papers studied saturated system models, and the optimality of myopic policies was established for a single server and two channels in [47], for arbitrary number of channels in [1], and for arbitrary number of channels and servers in [2].

Switching delay has been considered in Polling models in the Queuing Theory literature for single-server systems (e.g., [4] and [43]). However, time-varying channels were not considered since they do not typically arise in classical Polling applications. A detailed survey of the works in this field can be found in [43]. Scheduling in optical or manufacturing networks under reconfiguration delay was considered in [8], [9], [14], and [37], again in the absence of time-varying channels. In [13], we considered a simple queuing system of two queues and a single-server subject to ON/OFF channels and a single-slot switchover delay, where we developed the state-action frequency approach and the throughput optimality of a frame-based policy. Here, we generalize this model to arbitrary single-hop networks.

The main contribution of this paper is in solving the scheduling problem in single-hop networks under arbitrary *reconfiguration delays*, *time-varying channels*, and *interference constraints* for the first time. We introduce the system model in detail in Section II. For systems with memoryless channel processes, we characterize the stability region and propose the class of throughput-optimal VFMW policies in Section III. We develop the state-action frequency approach and characterize the stability region for systems with Markov modulated channels in Section IV-A. We develop the throughput-optimal FBDC policy in Section IV-B and present simulation results in Section IV-D.

II. MODEL

Consider a single-hop wireless network given by a graph structure $\mathcal{G}(\mathcal{N}, \mathcal{L})$ of nodes \mathcal{N} and links $\ell \in \mathcal{L} \doteq \{1, 2, \dots, L\}$, where $L \doteq |\mathcal{L}|$. Data packets arriving at each link ℓ are to be transmitted to their single-hop destinations, where we refer to the packets waiting for service at link ℓ as queue ℓ . We consider

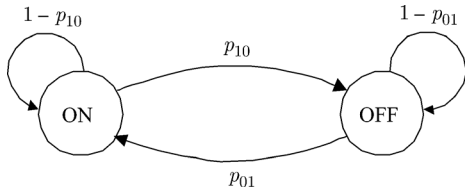


Fig. 3. Markov modulated ON/OFF channel process. The case of $p_{10} + p_{01} < 1$ provides positive correlation.

a discrete-time (slotted) system where an integer number of data packets can arrive at or depart from the corresponding queue at each link during each time-slot. Let the i.i.d. stochastic process $A_\ell(t)$ with arrival rate λ_ℓ denote the number of packets arriving to the source node of link ℓ at time-slot t , where $\mathbb{E}[A_\ell^2(t)] \leq A_{\max}^2$, for all links $\ell \in \mathcal{L}$. Let $\mathbf{Q}(t) = \{Q_1(t), \dots, Q_L(t)\}$ denote the queue sizes at the links at the beginning of time-slot t . Each link $\ell \in \mathcal{L}$ is subject to a time-varying channel process denoted by $C_\ell(t)$ that takes values in a discrete set $\mathcal{C} = \{0, \mu_{\min}, \dots, \mu_{\max}\}$ with $K \doteq |\mathcal{C}|$, where $C_\ell(t)$ corresponds to the number of packets that can be served from queue ℓ at time t . We consider both memoryless channel processes and Markovian channels with memory as defined below.

Definition 1 (Memoryless Channels): For each link $\ell \in \mathcal{L}$, the channel process $\{C_\ell(t); t \geq 0\}$ takes i.i.d. values from the set \mathcal{C} at each time-slot t , according to a probability distribution for link ℓ , \mathbb{P}^ℓ .

A simple example of a memoryless channel process is the Bernoulli process with two-state i.i.d. ON/OFF channels.

Definition 2 (Channels With Memory): The channel process $\{C_\ell(t); t \geq 0\}$ forms a K -state irreducible and aperiodic Markov chain over the set \mathcal{C} , according to a transition probability distribution $\mathbb{P}^\ell(\cdot|j), j \in \mathcal{C}$.

The basic example of a Markovian channel process with memory is the commonly used Gilbert–Elliot channel model shown in Fig. 3. We let \bar{C}_ℓ denote the time-average channel gain of link ℓ defined by

$$\bar{C}_\ell \doteq \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} C_\ell(\tau). \quad (1)$$

The limit exists both for memoryless and Markovian channel processes and is equal to the corresponding ensemble (steady state) average with probability (w.p.) 1 due to the Strong Law of Large Numbers (SLLN) [16]. We assume that all the arrival and channel processes, $A_\ell(t)$, $C_\ell(t)$, are independent of each other and across different links.

Let T_r denote the system reconfiguration delay, namely, it takes T_r time-slots for the system to change a schedule, during which all the servers are necessarily idle.¹ The set of all schedules in the system, \mathcal{I} , is given by the set of *feasible* binary activation vectors $\mathbf{I} = (\mathbf{I}_1, \mathbf{I}_2, \dots, \mathbf{I}_L) \in \{0, 1\}^L$. If the activation vector $\mathbf{I}(t)$ is used at time-slot t , then $\min\{C_\ell(t)\mathbf{I}_\ell(t), Q_\ell(t)\}$ packets depart from queue ℓ . We include the vectors dominated by the feasible activation vectors, as well as the zero vector $\mathbf{I} = \mathbf{0}$ in \mathcal{I} , where the activation vector $\mathbf{I}(t)$ is equal to $\mathbf{0}$ for all time-slots at which the system is undergoing reconfiguration. A policy $\boldsymbol{\pi}$ is a mapping from the set of all possible queue lengths,

¹Note that in a slotted system, even a minimal reconfiguration delay will lead to a loss of a slot due to synchronization issues.

channel process, and action histories to the set of all probability distributions on \mathcal{I} .

The feasibility of a schedule is determined by the interference constraints in the system, which are assumed to be arbitrary. For instance, in a wireless mesh network as shown in Fig. 1, the set \mathcal{I} can be determined according to the well-studied k -hop interference model [17], or signal-to-interference-plus-noise ratio (SINR) interference model [22]. Alternatively, for a satellite network of N queues and M servers where there are a possible $L = NM$ links as shown in Fig. 2, the set \mathcal{I} can be the set of all binary vectors of dimension NM with at most M nonzero elements such that no two active servers interfere with each other [14]. Finally, for an $N \times N$ input-queued optical switch, the set \mathcal{I} can be the set of all matchings [34].

We say that an activation vector \mathbf{I} is *ready to be activated in the current time-slot* if the system does not need to reconfigure in order to activate \mathbf{I} , i.e., in such a case the servers that will be activated under \mathbf{I} are *present* at their corresponding links at the beginning of the time-slot. Finally, we assume that the queues are initially empty and that the arrivals take place after the departures in any given time-slot. Under this model, the queue sizes evolve according to the following expression:

$$Q_\ell(t+1) = \max\{Q_\ell(t) - \mathbf{I}_\ell(t)C_\ell(t), 0\} + A_\ell(t) \quad \forall \ell \in \mathcal{L}. \quad (2)$$

Definition 3 (Stability [29], [30]): The system is stable if

$$\limsup_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \sum_{\ell \in \mathcal{L}} \mathbb{E}[Q_\ell(\tau)] < \infty.$$

For the case of integer-valued arrival and departure processes, as in this paper, this stability criterion implies the existence of a long-run stationary distribution for the queue-size Markov chain with bounded first moments [29].

Definition 4 (Stability Region [29], [30]): The stability region $\boldsymbol{\Lambda}$ is the closure of the set of all arrival rate vectors $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_L)$ such that there exists a control algorithm that stabilizes the system.

A policy is said to be throughput-optimal if it stabilizes the system for all input rates strictly inside $\boldsymbol{\Lambda}$. The δ -stripped stability region is defined for some $\delta > 0$ as $\boldsymbol{\Lambda}^\delta \doteq \{\boldsymbol{\lambda} | (\lambda_1 + \delta, \dots, \lambda_L + \delta) \in \boldsymbol{\Lambda}\}$.

Throughout the paper, we represent vectors, matrices, and sets with bold letters, and we explicitly state when a variable is a matrix. We use the following notation for the inner product of two L -dimensional vectors: $\mathbf{u} \cdot \mathbf{v} \doteq \sum_{\ell=1}^L \mathbf{u}_\ell \mathbf{v}_\ell$.

III. MEMORYLESS CHANNELS

A. Stability Region

We start by characterizing the system stability region for the case of memoryless channels.

Theorem 1 (Stability Region $\boldsymbol{\Lambda}$ —Memoryless Channels): The stability region $\boldsymbol{\Lambda}$ is given by

$$\boldsymbol{\Lambda} = \left\{ \boldsymbol{\lambda} \mid \exists \boldsymbol{\alpha} \geq \mathbf{0}, \sum_{\mathbf{I} \in \mathcal{I}} \alpha_{\mathbf{I}} \leq 1, \text{ such that } \lambda_\ell \leq \bar{C}_\ell \sum_{\mathbf{I} \in \mathcal{I}} \alpha_{\mathbf{I}} \mathbf{I}_\ell, \forall \ell \in \mathcal{L} \right\}. \quad (3)$$

The proof is omitted for brevity and can be found in [11]. The sufficiency of Theorem 1 also follows from Theorem 2 and

Corollary 1 in Section III-B, where we show that a variable frame-based algorithm that keeps the current activation for a duration of time based on the current queue lengths and average channels gains is throughput-optimal. The basic intuition behind Theorem 1 is that no policy can take advantage of the diversity in time-varying memoryless channels and achieve a greater rate than the *average* channel gain for each link. This is because in the presence of reconfiguration delays, the system cannot switch to another schedule instantly in order to opportunistically exploit better channel states, but can switch only after (at least) one time-slot and observe an *average* channel gain upon switching. This is in sharp contrast to the corresponding systems without reconfiguration delay considered in [30], [31], and the references therein, where throughput-optimal policies are able to take advantage of the diversity in i.i.d. channels by instantly and opportunistically switching schedules. As shown in the example below, for a simple two-queue system, this negative impact of the reconfiguration delay reduces the stability region considerably as compared to systems without reconfiguration delays in [30] and [40].

Theorem 1 also establishes that, as long as $T_r \geq 1$, the duration of the reconfiguration interval has no effect on the stability region of the system with memoryless channel processes. This is because for memoryless channels, giving infrequent reconfiguration decisions minimizes the fraction of time-slots lost to reconfiguration. In fact, this is the intuition behind the throughput-optimal policy proposed in Section III-B, where the policy delays the reconfiguration decisions as a function of the queue lengths and the channel gains.

1) *Example: Two Queues and a Single Server:* Consider i.i.d. ON/OFF channels with probability of ON channel state equal to 0.5 for both queues for all time-slots, and 1 slot switchover delay for the server. The stability region of this system can be obtained from (3), and it takes a simple structure

$$\mathbf{A} = \{(\lambda_1, \lambda_2) \mid \lambda_1 + \lambda_2 \leq 0.5, \lambda_1, \lambda_2 \geq 0\}. \quad (4)$$

The stability region of the corresponding system with no switchover delay was established in [40]: $\lambda_1, \lambda_2 \in [0, 0.5]$ and $\lambda_1 + \lambda_2 \leq 0.75$. As depicted in Fig. 4, even 1-slot switchover delay reduces the stability region of the system considerably. Note that for systems in which channels are always connected, the stability conditions are given by $\lambda_1 + \lambda_2 \leq 1, \lambda_1, \lambda_2 \geq 0$ and is *not* affected by the switchover delay [43]. Therefore, *it is the combination of switchover delay and time-varying channels that results in fundamental changes in system stability*. As we show in Section IV, and as displayed in Fig. 4, the memory in the channel processes can be exploited to improve the system stability region when the reconfiguration delay is nonzero.

B. Variable Frame-Based Max-Weight (VFMW) Algorithm

In this section, we propose a throughput-optimal algorithm based on the following intuition: Given that no policy can take advantage of the diversity in channel processes, making infrequent reconfiguration decisions minimizes throughput lost to reconfiguration. For networks with nonzero reconfiguration delays, in the absence of randomly varying connectivity, we proved in [14] that a variable-size frame-based Max-Weight algorithm that keeps the same schedule over a frame of duration based on the queue lengths is throughput-optimal. We show here that an adaptation of the algorithm in [14] that

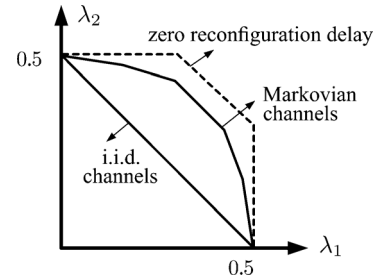


Fig. 4. Stability region under memoryless (i.i.d.) channels and channels with memory (Markovian with $p_{10} + p_{01} < 1$) with and without switchover delay.

Algorithm 1: VFMW Algorithm With Frame Length χ_k
 $= T_r + F(S(\mathbf{Q}(t_k)))$

- 1: Find the Max-Weight schedule at time t_k , $\mathbf{I}^*(t_k)$, w.r.t. $\mathbf{Q}(t_k)$ weighted by the average channel gains $\bar{\mathbf{C}}$

$$\mathbf{I}^*(t_k) = \arg \max_{\mathbf{I} \in \mathcal{I}} \sum_{\ell} \mathbf{I}_{\ell} \bar{\mathbf{C}}_{\ell} Q_{\ell}(t_k).$$
 - 2: For the next T_r slots, if $\mathbf{I}^*(t_k) \neq \mathbf{I}^*(t_{k-1})$, then invoke reconfiguration, otherwise apply $\mathbf{I}^*(t_k)$.
 - 3: Apply $\mathbf{I}^*(t_k)$ for an interval of duration $F(S(\mathbf{Q}(t_k)))$ slots where $F(\cdot) > 0$ is a monotonically increasing sublinear function, i.e., $\lim_{y \rightarrow \infty} F(y)/y = 0$.
 - 4: Repeat above for the next frame starting at $t_{k+1} = t_k + \chi_k$.
-

also takes into account the *average channel gains of the time-varying links* is throughput-optimal for systems with memoryless channel processes. Specifically, let t_k be the first slot of the k th frame, let $\mathbf{Q}(t_k)$ be the queue lengths at t_k , and let $S(\mathbf{Q}(t_k)) \doteq \sum_i Q_i(t_k)$. The VFMW policy calculates the Max-Weight schedule with respect to $\mathbf{Q}(t_k)$ and $\bar{\mathbf{C}} \doteq (\bar{C}_1, \dots, \bar{C}_L)$ and applies this schedule during the frame as defined in detail in Algorithm 1.

The VFMW algorithm sets the frame length as a suitably increasing sublinear function of the queue lengths, which dynamically adapts the frame duration to the stochastic arrivals. For instance, $\chi_k = T_r + (\sum_i Q_i(t_k))^{\alpha}$ with $\alpha \in (0, 1)$ satisfies the criteria for the frame duration. Under the VFMW policy, the frequency of service reconfiguration is small when the queue lengths are large, limiting the fraction of time spent on switching. Note that this frequency should not be too small, otherwise the system becomes unstable as it is subjected to a bad schedule for an extended period of time. When the queue lengths are small, the VFMW policy makes frequent reconfiguration decisions, becoming more adaptive and providing good delay performance².

Theorem 2: The VFMW policy is throughput-optimal. An immediate corollary to this theorem is as follows.

Corollary 1: The conditions in (3) are sufficient for stability. The proof of Theorem 2 is given in Appendix A and is presented using the frame length function $\chi_k = T_r + (\sum_i Q_i(t_k))^{\alpha}$ for a fixed $\alpha \in (0, 1)$ for ease of exposition. It establishes the fact that the drift over the switching epochs, i.e.,

²Note that a similar policy was considered in [38] to stabilize a very different system, i.e., a system without reconfiguration delays but with asynchronous transmission opportunities.

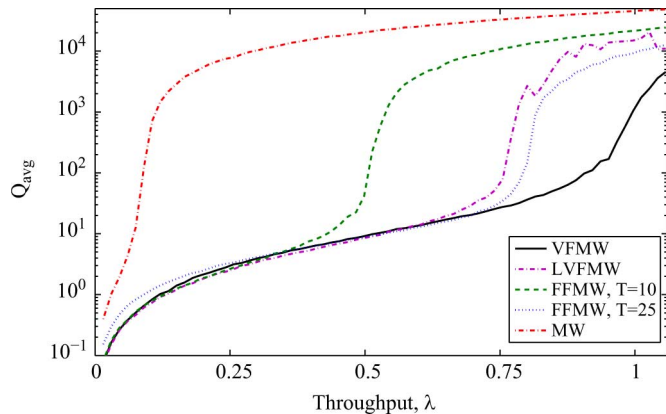


Fig. 5. Average queue length versus throughput under the VFMW, LVFMW, MW, and the FFMW policies.

$\mathbb{E} [L(\mathbf{Q}(t_k + \chi_k)) - L(\mathbf{Q}(t_k)) | \mathbf{Q}(t_k)]$, is negative using the quadratic Lyapunov function, $L(\mathbf{Q}(t)) = \sum_{\ell=1}^L Q_{\ell}^2(t)$. The basic intuition behind the proof is that if the queue sizes are large, the VFMW policy accumulates sufficient negative drift during the frame, which overcomes the cost (i.e., arrivals) accumulated during reconfiguration. Moreover, for large queue lengths, since the policy keeps the same schedule during the resulting long frames, we obtain the time-average channel gains in the system, as seen in the stability condition in (3). Note that choosing the frame length as a *sublinear* function of the queue sizes is critical. This is because the VFMW algorithm uses the Max-Weight schedule corresponding to the beginning of the frame, which “loses weight” as the frame goes on. Therefore, one needs to make sure that the system is not subjected to this “lightweight” schedule for too long. While frame lengths that are sublinear in the queue size are provably stable using standard Lyapunov techniques, frame lengths that are linear in the queue size may not be stable. In fact, simulation results in Fig. 5 show that the VFMW policy implemented using frame lengths that are linear in sum-queue lengths fails to stabilize the system for all arrival rates in the stability region.

Note that the standard Max-Weight scheduling algorithm or its variants are not throughput optimal for systems with nonzero reconfiguration delays. This is proved in [14, Lemma 1] for systems without time-varying channels. Since the case of no time variation in channel gain is a special case of our model ($C_i(t) = 1, \forall i, t$), Max-Weight scheduling algorithm is not throughput-optimal for the system considered here. The reason behind this is that these policies switch the current schedule too often, independent of the reconfiguration delay, wasting a significant fraction of throughput during switching. This can be seen in simulation in Fig. 5 where the Max-Weight policy becomes unstable much before VFMW policy as the arrival rates are increased. It is interesting to note that the VFMW policy requires the knowledge of average channel gains ($\bar{C}_1, \dots, \bar{C}_L$) as opposed to ordinary Max-Weight or its variations, which require instantaneous channel state information ($C_1(t), \dots, C_L(t)$) for *all* time-slots. This is an important advantage for the VFMW policy over Max-Weight as instantaneous channel-state knowledge may not be available in all time-slots, or it may be costly to obtain.

In Section IV, we show that for channel processes with memory, delaying the reconfiguration decisions as in the

VFMW algorithm does not work, and more sophisticated algorithms are necessary in order to exploit the channel memory.

C. Simulation Results—Memoryless Channels

We performed simulation experiments that determine average queue occupancy values for the VFMW policy, the version of the VFMW policy implemented using frame lengths linear in queue lengths (termed LVFMW policy), the ordinary Max-Weight (MW) policy and the Max-Weight policy with fixed frame sizes (FFMW). The MW policy “chooses” the schedule $\arg \max_{\mathbf{I}} \sum_{\ell} Q_{\ell}(t) C_{\ell}(t) \mathbf{I}_{\ell}$, and the FFMW policy applies the same activation vector as the VFMW policy over frames of constant duration. The average total queue occupancy over T_s slots is defined by $Q_{\text{avg}} \doteq \frac{1}{T_s} \sum_{t=1}^{T_s} \sum_{\ell \in \mathcal{L}} Q_{\ell}(t)$, and the frame lengths for the two implementations of the VFMW policy are chosen as $\chi_k = T_r + (\sum_i Q_i(t_k))^{0.9}$ and $\chi_k = T_r + 2 * (\sum_i Q_i(t_k))$. Through Little’s law, the average packet delay in the system is equal to the average queue size divided by the total arrival rate into the system. We considered a network of four links and three servers as shown in Fig. 2, where servers 1 and 3 are dedicated to links (queues) 1 and 4, respectively, and server 2 is shared between queues 2 and 3. This system can also model an appropriate single-hop wireless network as in Fig. 1. Due to interference constraints, no two links that are “adjacent” to each other can be activated simultaneously, namely, the set of feasible activations are given by $\mathbf{I}^1 = [1010]$, $\mathbf{I}^2 = [0101]$, and $\mathbf{I}^3 = [1001]$. For each data point, the simulation length was 100 000 slots, and the arrival and the channel processes were i.i.d. Bernoulli, with arrival rate λ , and probability of ON channel state equal to 0.5, respectively.

We simulated total average queue length as a function of sum-throughput $\sum_{\ell} \lambda_{\ell}$ for λ along the line between the origin and the maximum sum-throughput point λ_{max} given by $\arg \max_{\lambda \in \Lambda} \sum_{\ell} \lambda_{\ell}$, where from (3), λ_{max} can be calculated to be $[0.33 \ 0.17 \ 0.17 \ 0.33]$ with $\sum_{\ell} \lambda_{\ell} = 1$. Note that the maximum sum-throughput for the corresponding system with zero reconfiguration delay is about 1.44 [17], which shows the significant reduction in throughput due to the reconfiguration delay. Fig. 5 presents the delay as a function of sum-throughput for the VFMW, MW, and the FFMW (with frame sizes $T = 10$ and $T = 25$) policies, for $T_r = 5$ slot reconfiguration delay. Fig. 5 confirms that as the arrival rates are increased, the system quickly becomes unstable under the MW policy around $\sum_{\ell} \lambda_{\ell} 0.10$. We also observe that the VFMW policy provides stability for all sum-rates less than 1. The FFMW policy has larger stability region than that of the MW policy, and increasing the frame length of the FFMW policy improves its stability region at the expense of delay performance. The VFMW policy implemented using frame lengths linear in the queue lengths failed to stabilize the system for sum-throughputs larger than about 0.76.

In order to demonstrate the stability of the VFMW policy for a case where all other policies are unstable, for the same system and policies as in Fig. 5, we plot total queue length as a function of time for $\lambda = [0.29 \ 0.15 \ 0.15 \ 0.29]$ in Fig. 6. This figure shows that the MW, FFMW, and LVFMW policies have a linear growth trend in queue lengths over time, therefore they are unstable. On the other hand, under the VFMW policy with frame length $\chi_k = T_r + (\sum_i Q_i(t_k))^{0.9}$, the system is stable, and it reaches steady state, where the total queue length fluctuates

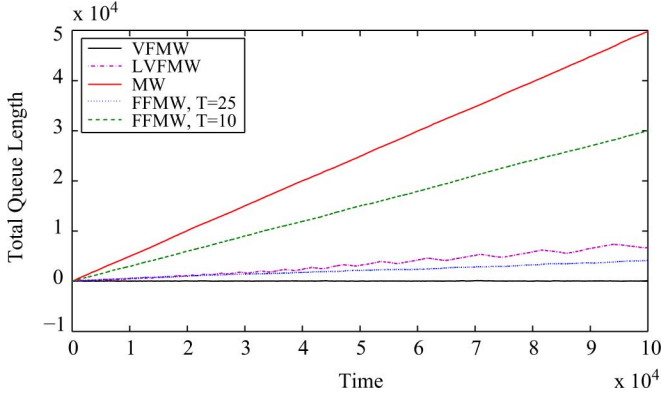


Fig. 6. Total queue length versus time for a fixed arrival rate $\lambda = [0.29 \ 0.15 \ 0.15 \ 0.29]$ under the VFMW, LVFMW, MW, and the FFMW policies.

around a fixed value, and does not appear to grow. The VFMW policy provides a good balance by dynamically adapting the frame length based on the queue sizes and stabilizes the system whenever possible, while providing a delay performance that is similar to that of a FFMW policy with a small frame length for small arrival rates.

IV. CHANNELS WITH MEMORY

In this section, we consider systems that have time-varying channels with memory. We establish the stability region for such systems and propose a throughput-optimal dynamic control policy. We generalize the novel framework we introduced in [13] that characterizes the stability region in terms of state-action frequencies to wireless networks with reconfiguration delays, time-varying channels, and interference constraints. The state-action frequency approach is a general and unifying framework in that, in the absence of reconfiguration delays, this approach simplifies to stability region characterizations introduced in the literature [31], [34], [40].

We show that *the stability region expands with memory in the channel processes*; in particular, it lies between the stability region for the case of i.i.d. channels and the stability region in the absence of reconfiguration delay. For the traditional network control models without reconfiguration delays such as the models considered in [30], [31], and [40], the memory in the channel processes does not affect the stability region [17]. Therefore, scheduling under reconfiguration delays and time-varying channels calls for novel control algorithms that take advantage of the channel memory to improve performance.

A. Stability Region

We start by analyzing the corresponding system with saturated queues, i.e., all queues always have a packet to send. Let Λ_s denote the set of all time average expected departure rate vectors $\mathbf{r} = (r_1, \dots, r_L)$ that can be obtained in the saturated system under all possible policies that are possibly *history-dependent, randomized, or nonstationary*. We will show that the stability region Λ satisfies $\Lambda = \Lambda_s$. We prove the necessary stability conditions in the following lemma and establish sufficiency in Section IV-B.

Lemma 1: We have that $\Lambda \subseteq \Lambda_s$.

Proof: Given a policy π for the dynamic queueing system, consider the saturated system with the *same sample path of channel gains* for $t \in \{0, 1, 2, \dots\}$ and the *same set of actions* as policy π at each time-slot $t \in \{0, 1, 2, \dots\}$. Given a queue ℓ , let $D_\ell(t)$ be the total number of departures by time t from queue ℓ in the dynamic queueing system under policy π , and let $D'_\ell(t)$ be the corresponding quantity for the saturated system. Since some of the nonzero channel gains are wasted in the original system due to empty queues, we have

$$D_\ell(t) \leq D'_\ell(t). \quad (5)$$

Therefore, the time average expectation of $D_\ell(t)$, $\ell \in \mathcal{L}$ is also less than or equal to the time average expectation of $D'_\ell(t)$. This completes the proof since (5) holds under any policy π for the original system. \blacksquare

We establish the region Λ_s by formulating the system dynamics as an MDP.

1) *MDP Formulation for Saturated System:* For ease of exposition, we present the analysis for the case of a single-slot reconfiguration delay, i.e., $T_r = 1$. For $T_r = 1$, let $\mathbf{s}_t = (\mathbf{I}(t), \mathbf{C}(t)) \in \mathcal{S}$ denote the system state at time t , where $\mathbf{I}(t)$ is the activation vector in use at time-slot t (i.e., the current set of locations of the servers), $\mathbf{C}(t)$ is the vector of channel gains at each link at time-slot t , and \mathcal{S} is the set of all states. Also, let $\mathbf{a}_t \in \mathcal{I}$ denote the action taken at time-slot t , which determines the activation vector that will be available at the beginning of the next time-slot, namely the next set of locations for the servers. For $T_r > 1$ the state would have one more variable that counts the number of time-slots left until the end of the current reconfiguration. Namely, for $T_r > 1$, we have $\mathbf{s}_t = (\mathbf{I}(t), \mathbf{C}(t), n(t))$ where $n(t) \in [0, 1, \dots, T_r]$ denotes the number of remaining time-slots until the end of the reconfiguration interval. The action \mathbf{a}_t is as before for the case of $n(t) = 0$, however \mathbf{a}_t takes a single value for the case $n(t) > 0$, say $\mathbf{a}_t = 1$, corresponding to the case of no action, and $n(t)$ decreases by 1.

Let $\mathbb{H}(t) = [\mathbf{s}_\tau]_{\tau=0}^t \cup [\mathbf{a}_\tau]_{\tau=0}^{t-1}$ denote the full history of the system until time t , and let $\Upsilon(\mathcal{I})$ denote the set of all probability distributions on the set of all actions \mathcal{I} . For the saturated system, a policy is a mapping from the set of all possible channel state and action histories to $\Upsilon(\mathcal{I})$ [25], [32]. Namely, a policy prescribes the probability of any particular action for every given system history. A policy is said to be *stationary* if, given a particular state, it applies the same decision rule in all stages, and under a stationary policy the process $\{\mathbf{s}_t; t \geq 0\}$ forms a Markov chain.

In each time-slot t , the server observes the current state \mathbf{s}_t and chooses an action \mathbf{a}_t . Then, the next state j is realized according to the transition probabilities $\mathbb{P}(j|\mathbf{s}, \mathbf{a})$, which depend on the random channel processes. Let \mathbf{I}_ℓ be 1 if link ℓ is active under the activation vector \mathbf{I} , and 0 otherwise. We define the reward for link ℓ as a function of the state $\mathbf{s}_t = (\mathbf{I}(t), \mathbf{C}(t))$ as follows:

$$\bar{r}_\ell(\mathbf{s}_t, \mathbf{a}_t) \doteq C_\ell(t), \text{ if } \mathbf{I}_\ell(t) = 1 \text{ and } \mathbf{a}_t \doteq \mathbf{I}(t+1) = \mathbf{I}(t) \quad (6)$$

and $\bar{r}_\ell(\mathbf{s}_t, \mathbf{a}_t) \doteq 0$ otherwise. That is, a reward of $C_\ell(t)$ is obtained if the controller decides to stay with the current schedule and if link ℓ is active under the current schedule. We are interested in the set of all possible time average expected departure rates. Therefore, given some $\alpha_\ell \geq 0$, $\ell \in \mathcal{L}$, we define the system reward at time t by the weighted sum-throughput

$\bar{r}(\mathbf{s}_t, \mathbf{a}_t) \doteq \sum_{\ell \in \mathcal{L}} \alpha_\ell \bar{r}_\ell(\mathbf{s}_t, \mathbf{a}_t)$. The average reward of policy $\boldsymbol{\pi}$ is defined by

$$r^\pi \doteq \limsup_{T \rightarrow \infty} \frac{1}{T} E \left[\sum_{t=1}^T \bar{r}(\mathbf{s}_t, \mathbf{a}_t^\pi) \right]. \quad (7)$$

Given weights $\alpha_\ell \geq 0, \ell \in \mathcal{L}$, we are interested in the policy that achieves the maximum time average expected reward $r^* \doteq \max_{\boldsymbol{\pi}} r^\pi$. This optimization problem is a discrete-time MDP characterized by the state transition probabilities $\mathbb{P}(\cdot | \mathbf{s}, \mathbf{a})$ with $K^L |\mathcal{I}|$ states and $|\mathcal{I}|$ actions per state, where K is the number of channel states. Furthermore, any given pair of states are accessible from each other (i.e., there is a positive probability path between the states) under some stationary deterministic policy. Therefore, this MDP belongs to the class of *Weakly Communicating* MDPs [32], for which there exists a stationary deterministic optimal policy independent of the initial state [32].

2) *State-Action Frequency Approach*: For Weakly Communicating MDPs with finite state and action spaces and bounded rewards, there exists an optimal stationary-deterministic policy, given as a solution to the standard Bellman's equation, with optimal average reward independent of the initial state [32, Theorem 8.4.5]. This is because if a stationary policy has a nonconstant gain over initial states, one can construct another stationary policy with constant gain that dominates the former policy, which is possible since there exists a positive probability path between any two (recurrent) states under some stationary policy [25]. The *state-action frequency* approach, or the *Dual LP* approach, given as follows provides a systematic and intuitive framework to solve such average cost MDPs, and it can be derived using Bellman's equation and the monotonicity property of Dynamic Programs [32, Sec. 8.8]:

$$\max_{\mathbf{x} \in \mathbf{X}} \cdot \sum_{\mathbf{s} \in \mathcal{S}} \sum_{\mathbf{a} \in \mathcal{I}} \bar{r}(\mathbf{s}, \mathbf{a}) \mathbf{x}(\mathbf{s}, \mathbf{a}) \quad (8)$$

subject to the balance equations

$$\sum_{\mathbf{a} \in \mathcal{I}} \mathbf{x}(\mathbf{s}, \mathbf{a}) = \sum_{\mathbf{s}' \in \mathcal{S}} \sum_{\mathbf{a}' \in \mathcal{I}} \mathbb{P}(\mathbf{s} | \mathbf{s}', \mathbf{a}') \mathbf{x}(\mathbf{s}', \mathbf{a}') \quad \forall \mathbf{s} \in \mathcal{S} \quad (9)$$

the normalization condition

$$\sum_{\mathbf{s} \in \mathcal{S}} \sum_{\mathbf{a} \in \mathcal{I}} \mathbf{x}(\mathbf{s}, \mathbf{a}) = 1 \quad (10)$$

and the nonnegativity constraints

$$\mathbf{x}(\mathbf{s}, \mathbf{a}) \geq 0, \quad \text{for } \mathbf{s} \in \mathcal{S}, \mathbf{a} \in \mathcal{I}. \quad (11)$$

Note that the effect of the reconfiguration delay on (8)–(11) is through the system state. Namely, if the reconfiguration delay is zero, the system state does not have to include \mathbf{I} , the location information of the servers. The feasible region of this LP [i.e., (9), (10), and (11)] constitutes a polytope called the *state-action polytope* \mathbf{X} , and the elements of this polytope $\mathbf{x} \in \mathbf{X}$ are called the state-action frequency vectors. A component of a state-action frequency vector, $\mathbf{x}(\mathbf{s}, \mathbf{a})$, corresponds to the probability that the system is at state \mathbf{s} and action \mathbf{a} is taken under the following stationary randomized policy: Take action \mathbf{a} at state \mathbf{s} w.p.

$$\mathbb{P}(\text{action } a \text{ at state } \mathbf{s}) = \frac{\mathbf{x}(\mathbf{s}, \mathbf{a})}{\sum_{\mathbf{a}' \in \mathcal{I}} \mathbf{x}(\mathbf{s}, \mathbf{a}')}, \quad \mathbf{a} \in \mathcal{I}, \mathbf{s} \in S_x \quad (12)$$

where S_x is the set of recurrent states, i.e., S_x is the set of states that have positive probability of occupancy in steady state [25], [32]

$$S_x \equiv \left\{ \mathbf{s} \in \mathcal{S} : \sum_{\mathbf{a} \in \mathcal{I}} \mathbf{x}(\mathbf{s}; \mathbf{a}) > 0 \right\}.$$

If there is a transient state \mathbf{s}' , i.e., $\mathbf{s}' \in \mathcal{S}/S_x$, then an action that leads the system to the set S_x is chosen at \mathbf{s}' . It can be shown that \mathbf{X} is convex, bounded, and closed [25].

Next, we argue that the empirical state-action frequencies corresponding to any given policy (*possibly randomized, history-dependent, nonstationary, or non-Markovian*) lie in the state-action polytope \mathbf{X} . This ensures us that the optimal solution to the dual LP in (8) is over possibly nonstationary and history-dependent policies. In the following, we give the precise definition and the properties of the set of empirical state-action frequencies. We define the *empirical* state-action frequencies $\hat{x}^T(\mathbf{s}, a)$ as

$$\hat{x}^T(\mathbf{s}, \mathbf{a}) \doteq \frac{1}{T} \sum_{t=1}^T I_{\{\mathbf{s}_t=\mathbf{s}, \mathbf{a}_t=\mathbf{a}\}} \quad (13)$$

where I_E is the indicator function of an event E , i.e., $I_E = 1$ if E occurs, and $I_E = 0$ otherwise. Given a policy $\boldsymbol{\pi}$, let P^π be the state-transition probabilities under $\boldsymbol{\pi}$, and let $\phi = (\phi_s)$ be an initial state distribution with $\sum_{\mathbf{s} \in \mathcal{S}} \phi_s = 1$. We let $x_{\boldsymbol{\pi}, \phi}^T(\mathbf{s}, a)$ be the *expected empirical* state-action frequencies under policy $\boldsymbol{\pi}$ and initial state distribution ϕ

$$\begin{aligned} x_{\boldsymbol{\pi}, \phi}^T(\mathbf{s}, \mathbf{a}) &\doteq E^{\boldsymbol{\pi}, \phi} [\hat{x}^T(\mathbf{s}, \mathbf{a})] \\ &= \frac{1}{T} \sum_{t=1}^T \sum_{\mathbf{s}' \in \mathcal{S}} \phi_{\mathbf{s}'} P^\pi(\mathbf{s}_t = \mathbf{s}, \mathbf{a}_t = \mathbf{a} | \mathbf{s}_0 = \mathbf{s}'). \end{aligned}$$

We let $\mathbf{x}_{\boldsymbol{\pi}, \phi} \in \Upsilon(\mathcal{S} \times \mathcal{A})$ (as in [25] and [32]) be the limiting expected state-action frequency vector, if it exists, starting from an initial state distribution ϕ , under a general policy $\boldsymbol{\pi}$ (possibly randomized, history-dependent, nonstationary, or non-Markovian):

$$\mathbf{x}_{\boldsymbol{\pi}, \phi}(\mathbf{s}, \mathbf{a}) = \lim_{T \rightarrow \infty} x_{\boldsymbol{\pi}, \phi}^T(\mathbf{s}, \mathbf{a}). \quad (14)$$

Let the set of all limit points be defined by

$$\mathbf{X}_{\Pi}^\phi \doteq \{ \mathbf{x} \in \Upsilon(\mathcal{S} \times \mathcal{A}) : \text{there exists a policy } \boldsymbol{\pi} \text{ s.t.} \\ \text{the limit in (14) exists and } \mathbf{x} = \mathbf{x}_{\boldsymbol{\pi}, \phi} \}. \quad (15)$$

Similarly let $X_{\Pi'}^\phi$ denote the set of all limit points of a particular class of policies Π' , starting from an initial state distribution ϕ . We let Π_{SD} denote the set of all stationary-deterministic policies, and we let $\text{co}(-10000 \text{fil})$ denote the closed convex hull of set -10000fil . The following theorem establishes the equivalency between the set of all achievable limiting state-action frequencies and the state-action polytope:

Theorem 3 [32, Theorem 8.9.3], [25, Theorem 3.1]: For any initial state distribution ϕ

$$\text{co}(\mathbf{X}_{\Pi_{\text{SD}}}^\phi) = \mathbf{X}_{\Pi}^\phi = \mathbf{X}.$$

We have $\text{co}(\mathbf{X}_{\Pi_{\text{SD}}}^\phi) \subseteq \mathbf{X}_{\Pi}^\phi$ since the convex combination of the vectors in $\mathbf{X}_{\Pi_{\text{SD}}}^\phi$ corresponds to limiting expected state-action frequencies for stationary-randomized

policies, which can also be obtained by time-sharing between stationary-deterministic policies. The inverse relation $co(\mathbf{X}_{\Pi_{SD}}^\phi) \supseteq \mathbf{X}_\Pi^\phi$ holds since for weakly communicating state structures, there exists a stationary-deterministic optimal policy independent of the initial state distribution. Next, for any stationary-deterministic policy, the underlying Markov chain is stationary, and therefore the limits $\mathbf{x}_{\pi,\phi}$ exist and satisfy the constraints (9)–(11) of the polytope \mathbf{X} . Using $co(\mathbf{X}_{\Pi_{SD}}^\phi) = \mathbf{X}_\Pi^\phi$ and the convexity of \mathbf{X} establishes $\mathbf{X}_\Pi^\phi \subseteq \mathbf{X}$. Furthermore, via (12), every $\mathbf{x} \in \mathbf{X}$ corresponds to a stationary-randomized policy for which the limits $\mathbf{x}_{\pi,\phi}$ exist, establishing $\mathbf{X}_\Pi^\phi \supseteq \mathbf{X}$.

Letting $\text{ext}(\mathbf{X})$ denote the set of extreme (corner) points of \mathbf{X} , an immediate corollary to Theorem 3 is as follows.

Corollary 2 [25], [32]: For any initial state distribution ϕ

$$\text{ext}(\mathbf{X}) = \mathbf{X}_{\Pi_{SD}}^\phi.$$

The intuition behind this corollary is that 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. Thus, corner points of \mathbf{X} have a one-to-one correspondence with the stationary-deterministic policies.

Finally, we have that, under any policy, the probability of a large distance between the empirical expected state-action frequency vectors and the state-action polytope \mathbf{X} decays exponentially fast in time. This result is similar to the mixing time of an underlying Markov chain to its steady state, and we utilize such convergence results within the Lyapunov drift analysis for the dynamic queuing system in Section IV-B.

3) *Rate Polytope $\mathbf{\Lambda}_s$* : Using the theory of state-action polytopes in Section IV-A.2, we characterize $\mathbf{\Lambda}_s$, the set of all achievable time-average expected rates in the saturated system. The following linear transformation of the state-action polytope \mathbf{X} defines the L -dimensional *rate polytope* [25]:

$$\mathbf{\Lambda}_s = \left\{ \mathbf{r} \mid r_\ell = \sum_{\mathbf{s} \in \mathcal{S}} \sum_{\mathbf{a} \in \mathcal{I}} \bar{r}_\ell(\mathbf{s}, \mathbf{a}) \mathbf{x}(\mathbf{s}, \mathbf{a}), \ell \in \mathcal{L} \right\} \quad (16)$$

where \bar{r}_ℓ is the reward function for link ℓ defined in (6). This polytope is the set of all time average expected departure rate vectors that can be obtained in the saturated system, i.e., it is the rate region $\mathbf{\Lambda}_s$. Furthermore, $\mathbf{\Lambda}_s$ is a linear transformation of \mathbf{X} . This is because the reward functions, $r_\ell, \ell \in \mathcal{L}$ in (16), are linear combinations of $\mathbf{x}(\mathbf{s}, \mathbf{a})$. To see this, given a link ℓ , the coefficient of the linear combination $\bar{r}_\ell(\mathbf{s}, \mathbf{a})$ is equal to C_ℓ if the action \mathbf{a} is the same as the current schedule in the state \mathbf{I} and if the l th component of \mathbf{I} is 1 [see (6)]. Therefore, corner points of $\mathbf{\Lambda}_s$ are also achieved by stationary deterministic policies. An explicit way of characterizing $\mathbf{\Lambda}_s$ is given in Algorithm 2.

We will establish in the next section that the rate region $\mathbf{\Lambda}_s$ is in fact achievable in the dynamic queueing system, which will imply that $\mathbf{\Lambda} = \mathbf{\Lambda}_s$.

Note that rate region $\mathbf{\Lambda}_s$ can be obtained by solving the following LP for some $\alpha_1, \dots, \alpha_L \geq 0$:

$$\begin{aligned} \max_{\mathbf{x}} \quad & \sum_{\ell=1}^L \alpha_\ell r_\ell(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{x} \in \mathbf{X}. \end{aligned} \quad (17)$$

Algorithm 2: Stability Region Characterization

- 1: Given a stationary deterministic policy π , find the corresponding basic feasible solution \mathbf{x} of the LP in (8) (namely, the corresponding corner point of \mathbf{X}) by solving (9)–(11).
 - 2: Find the corresponding rate vector (r_1^*, \dots, r_L^*) using the linear transformation in (16).
 - 3: Repeat the above steps for all possible corners of the polytope \mathbf{X} . Note that there are $K^L |\mathcal{I}|^2$ stationary deterministic policies.
 - 4: Take the convex combination of the resulting rate vectors.
-

The fundamental theorem of Linear Programming guarantees the existence of an optimal solution to (17) at a corner point of the polytope \mathbf{X} and hence of $\mathbf{\Lambda}_s$ [7]. By enumerating the weights $\alpha_1, \dots, \alpha_L \geq 0$, it is possible to obtain the corner points of $\mathbf{\Lambda}_s$, and thus $\mathbf{\Lambda}_s$. While exact enumeration of all possible weight vectors, corresponding to corner points of $\mathbf{\Lambda}_s$, may be prohibitive, this approach may be useful to numerically approximate the stability region.

We note that our characterization of the stability region does not scale with the number of nodes. This is also true for networks without reconfiguration delays. In fact, for a fixed channel state, the stability region of the corresponding system without any reconfiguration delay is the convex-hull of the possible activations. Typically, the number of activations is exponential in the number of nodes. In fact, the stability region characterization is NP-hard. For example, it is generally NP hard to compute the maximum achievable rate between a pair of nodes [19]. Nonetheless, the stability region characterization is useful for some special cases (such as bipartite graphs) and for establishing the existence of a stationary randomized policy. Such properties are also true for our characterization of the stability region for systems with reconfiguration delay. While our approach allows us to find points in the stability region numerically, in some very special cases it leads to the full characterization of the stability region. For instance, for the two-queue and single-server system introduced in Section III-A.1, the LP in (17) can be solved explicitly to derive the rate region $\mathbf{\Lambda}_s$, which is displayed in Fig. 7. As expected, the stability region is improved for channels with memory as compared to the stability region for the case of i.i.d. channels shown in Fig. 4. Moreover, the region $\mathbf{\Lambda}_s$ is significantly reduced as compared to the stability region for the corresponding system with zero reconfiguration delays shown with dashed lines in Fig. 7.

B. Frame-Based Dynamic Control Policy

We propose an FBDC policy inspired by the state-action frequency approach and prove that it is throughput-optimal. The motivation behind the FBDC policy is that a policy π^* that achieves the maximum in (17) for given weights $\alpha_\ell, \ell \in \mathcal{L}$ for the saturated system should achieve a *good* performance in the original system when the queue lengths are used as weights. This is because, first, the policy π^* will lead to similar average departure rates in both systems for sufficiently large queue lengths and, second, the usage of queue lengths as weights creates self-adjusting policies that dynamically capture the changes

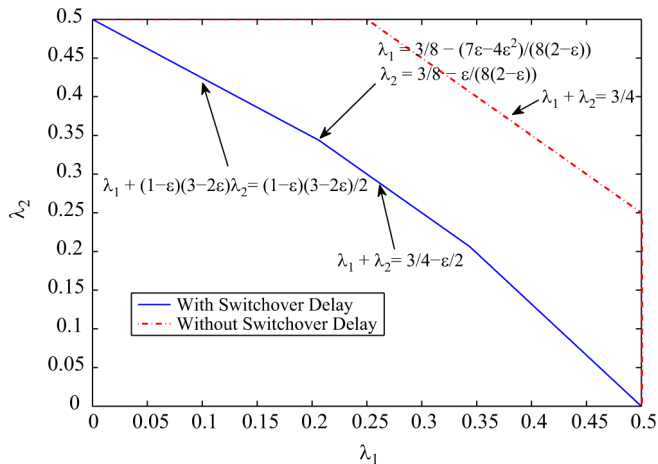


Fig. 7. Stability region for the simple system with and without switchover time for Gilbert–Elliot channel model with $p_{10} = p_{01} = 0.40$.

Algorithm 3: Frame-Based Dynamic Control Policy With Frame Length $\chi_k = T_r + F(S(\mathbf{Q}(t_k)))$

- 1: Find the policy π^* that optimally solves the following LP:

$$\begin{aligned} \max_{\mathbf{x}} \quad & \sum_{\ell=1}^L Q_{\ell}(t_k) r_{\ell}(\mathbf{x}) \\ \text{subject to } & \mathbf{x} \in \mathbf{X}. \end{aligned} \quad (18)$$

- 2: Apply π^* for an interval of duration $F(S(\mathbf{Q}(t_k)))$ slots where $F(\cdot) > 0$ is a monotonically increasing sublinear function, i.e., $\lim_{y \rightarrow \infty} F(y)/y = 0$.
-

due to stochastic arrivals. The FBDC algorithm is a variable frame-based algorithm similar to the VFMW algorithm proposed for the case of i.i.d. channel processes, where the frame lengths are sublinear functions of the queue lengths. Let $\mathbf{Q}(t_k)$ denote the queue lengths at the beginning of the k th frame. We find the stationary deterministic policy that optimally solves (17) when $Q_1(t_k), \dots, Q_L(t_k)$ are used as weights and then apply this policy throughout the frame in the dynamic queueing system. The FBDC policy is described in detail in Algorithm 3, where $S(\mathbf{Q}) = \sum_{\ell} Q_{\ell}$ denotes the sum of queue lengths.

The LP in (18) can be restated as $\max_{\{\mathbf{r}\}} \sum_{\ell=1}^L Q_{\ell}(t_k) r_{\ell}$ subject to $\mathbf{r} \in \mathbf{\Lambda}_s$. There exists an optimal solution \mathbf{r}^* of this LP, corresponding to the optimal solution \mathbf{x}^* of (18), that is a corner point of \mathbf{X} and hence of $\mathbf{\Lambda}_s$ [7]. Moreover, the policy π^* that corresponds to this corner point is a stationary deterministic policy by Corollary 2.

Theorem 4: The FBDC policy is throughput-optimal.

The proof of this theorem is given in Appendix B. This theorem implies that $\mathbf{\Lambda} = \mathbf{\Lambda}_s$. The proof performs a drift analysis using the standard quadratic Lyapunov function over variable frames whose lengths are functions of the queue lengths. However, it is novel in utilizing the state-action frequency approach of MDP theory within the Lyapunov drift framework. The basic idea is that for sufficiently large queue lengths, when the optimal policy solving (18), π^* , is applied over a sufficiently long frame of χ_k 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 and the steady-state rates decreases exponentially fast in time [25], similar to the convergence of a positive recurrent Markov chain to its steady state. Therefore, for sufficiently large queue lengths, the difference between the empirical rates in the actual system and \mathbf{r}^* also decreases with χ_k . This ultimately results in a negative drift for sufficiently large queue lengths since \mathbf{r}^* is the solution to $\max_{\mathbf{r}} \sum_{\ell=1}^L Q_{\ell} r_{\ell}$, $\mathbf{r} \in \mathbf{\Lambda}_s$, which leads to

$$\sum_{\ell} Q_{\ell} r_{\ell}^* \geq \sum_{\ell} Q_{\ell} \lambda_{\ell}$$

for all λ in $\mathbf{\Lambda}_s$.

Remark 1: The FBDC policy provides a new framework for developing throughput-optimal policies for network control. Namely, given any queuing system whose corresponding saturated system is Markovian with finite state and action spaces, throughput optimality is achieved by solving an LP in order to find the stationary MDP solution for the corresponding saturated system and applying this solution over frames in the actual system. The FBDC policy can also be used to achieve throughput optimality for classical network control systems [31], [40], optical switches [34], or systems with delayed channel state information [20], [45].

In the absence of time-varying channels, or for systems with i.i.d. channel processes as in Section III, variable-size frame-based generalizations of the Max-Weight policy are throughput-optimal [14]. However, under the simultaneous presence of reconfiguration delays and time-varying channels with memory, the FBDC policy is the only policy to achieve throughput optimality, and it has a significantly different structure from the Max-Weight policy.

We note that the FBDC policy can also be implemented without frames by setting the frame length to 1, namely, by solving the LP in Algorithm 3 in each time-slot. The simulation results in Section IV-D suggest that such an implementation has a similar throughput performance to the original FBDC policy. This is because the optimal solution of the LP in (18) depends on the ratio of the queue lengths that are used as weights. Therefore, the policy π^* solving the LP optimally stays as the solution for long periods of time when the queue lengths are large. Note that when the policy is implemented without frames, it becomes more adaptive to dynamic changes in the queue lengths, which results in better delay performance as compared to the frame-based implementations.

In Section IV-C, we consider simple Myopic policies that do not require the solution of an LP. Simulation results presented in Section IV-D suggest that the stability region achieved by the Myopic policies is close to the full stability region, while their delay performance is similar to that of the FBDC policy.

C. Myopic Control Policies

We investigate the performance of simple Myopic policies that make scheduling/switching decisions according to weight functions that are products of the queue lengths and the channel gain 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. Specifically, in the One-Lookahead Myopic policy, assuming that the system is employing

Algorithm 4: One-Lookahead Myopic Policy

- 1: Assuming that schedule \mathbf{I}^j is currently being used at time-slot t , calculate the following weights:

$$W_{\mathbf{I}^j}(t) = \sum_{\ell} \mathbf{I}_{\ell}^j \left(C_{\ell}(t) + \mathbb{E}[C_{\ell}(t+1)|C_{\ell}(t)] \right) Q_{\ell}(t)$$

$$W_{\mathbf{I}^i}(t) = \sum_{\ell} \mathbf{I}_{\ell}^i \mathbb{E}[C_{\ell}(t+1)|C_{\ell}(t)] Q_{\ell}(t) \quad \forall i \neq j.$$
(19)

- 2: If $W_{\mathbf{I}^j}(t) \geq W_{\mathbf{I}^i}(t)$, $i \neq j$, $i \in \mathcal{I}$, then stay with schedule \mathbf{I}^j . Otherwise, switch to a schedule with the maximum weight $W_{\mathbf{I}^i}(t)$.
-

schedule \mathbf{I}^j at some time-slot t , $W_{\mathbf{I}^j}$, the weight of \mathbf{I}^j , is obtained as follows: If link ℓ is active under \mathbf{I}^j , the contribution of link ℓ to $W_{\mathbf{I}^j}(t)$ is the product of $Q_{\ell}(t)$ and the sum of the expectations of $C_{\ell}(t)$ in the current and the next time-slot. The weight of other schedules are calculated similarly, except that we consider the expectation of channel processes only at time-slot $t+1$ because the system will idle at time-slot t in order to switch to other schedules. We describe the One-Lookahead Myopic (OLM) policy in Algorithm 4.

We investigated the performance of the OLM policy through simulations. The simulation results in Section IV-D suggest that the OLM policy may achieve the full stability region while providing a better delay performance as compared to the FBDC policy.

The k -Lookahead Myopic policy is a generalization of the OLM policy in that the following weight functions are used for scheduling decisions: Assuming that the system is currently employing schedule \mathbf{I}^j at time-slot t , $W_{\mathbf{I}^j}(t) = \sum_{\ell} \mathbf{I}_{\ell}^j (C_{\ell}(t) + \sum_{\tau=1}^k \mathbb{E}[C_{\ell}(t+\tau)|C_{\ell}(t)]) Q_{\ell}(t)$ and $W_{\mathbf{I}^i}(t) = \sum_{\ell} \mathbf{I}_{\ell}^i (\sum_{\tau=1}^k \mathbb{E}[C_{\ell}(t+\tau)|C_{\ell}(t)]) Q_{\ell}(t)$, $\forall i \neq j$. These policies have low complexity, and they are simpler to implement as compared to the FBDC policy.

D. Simulation Results—Channels With Memory

We performed simulation experiments that determine average queue occupancy values for the FBDC, OLM, and MW policies. The frame length for the k th frame of the FBDC policy is chosen to be $\chi_k = T_r + (\sum_i Q_i(t_k))^{0.9}$ as for the case of memoryless channels. We consider the same system and the simulation model as in Section III-C, except that we use the Gilbert–Elliot channel model in Fig. 3 and that the switching delay T_r is taken to be 1 slot.

We utilized three sets of transition probabilities, for $p \doteq p_{10} = p_{01}$; $p = 0.10, 0.25$, and 0.30 . As for the case of i.i.d. channels considered in Section III-C, the steady-state probability of ON channel state for each queue is 0.5 in each of these cases. By numerically solving the LP in (17), the maximum sum-throughput can be calculated to be 1.11 for $p = 0.30$, 1.14 for $p = 0.25$, and 1.29 for $p = 0.10$ as shown in Table I. While these values are significantly larger than the maximum sum-throughput of 1 for the case of i.i.d. channels, they are less than the sum-throughput of 1.44 for the corresponding system with zero reconfiguration delay, as expected. The enlargement in the stability region with channel memory is in sharp contrast

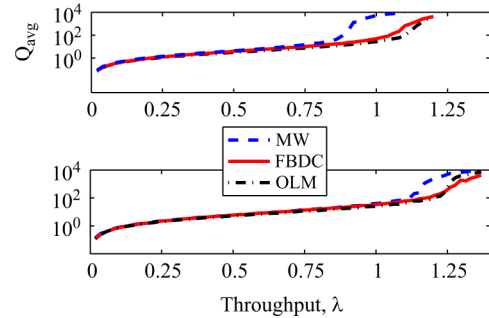


Fig. 8. Total average queue size versus the sum-throughput under the FBDC policy, the OLM, and the MW policies, for (top) $p_{10} = p_{01} = 0.25$ and for (bottom) $p_{10} = p_{01} = 0.10$.

TABLE I
MAXIMUM SUM-THROUGHPUT UNDER DIFFERENT VALUES OF CHANNEL MEMORY AND RECONFIGURATION DELAY

Model	p	T_r	max. sum-throughput
No switching delay	0.50	0	1.44
Memoryless	0.50	1	1.00
Memory level-1	0.30	1	1.11
Memory level-2	0.25	1	1.14
Memory level-3	0.10	1	1.29

to systems with zero reconfiguration delays for which the stability region only depends on the steady-state behavior of the channel processes [17].

Fig. 8 (top) presents delay as a function of sum-throughput along the line between the origin and the maximum sum-throughput point for the FBDC, the OLM, and the MW policies for $p_{10} = p_{01} = 0.25$. This figure shows that the system becomes unstable around the sum-throughput value of 0.9 under the MW policy. Moreover, the FBDC policy and the OLM policy have large queue lengths only for sum-throughput values outside the stability region. Furthermore, the OLM policy provides a similar delay performance to the FBDC policy in Fig. 8. Fig. 8 (bottom) shows delay as a function of sum-throughput for these policies for $p_{10} = p_{01} = 0.10$. While confirming similar results to Fig. 8 (top), Fig. 8 (bottom) also shows that the stability region becomes larger with increasing channel memory.

Fig. 9 presents total average queue length as a function of sum-throughput along the line between the origin and the maximum sum-throughput point for the FBDC policy implemented without frames, the OLM, and the MW policies for $p_{10} = p_{01} = 0.30$. This figure shows that the system becomes unstable around the sum-throughput value of 0.84 under the MW policy. Moreover, the FBDC and the OLM policies have large queue lengths only for sum-throughputs greater than the value of 1.11. This figures suggests that the no-frame implementations of the FBDC policy and the OLM policy provide a good throughput-delay performance. Furthermore, we observe that the difference in delay between the FBDC and the Max-Weight policies is wider for the no-frame implementation of the FBDC policy in Fig. 9, as compared to Fig. 8. This suggests that as the frame length is decreased, the delay performance of the FBDC policy improves.

Fig. 10 presents the total average queue length under a version of the FBDC policy implemented using a fixed frame of length

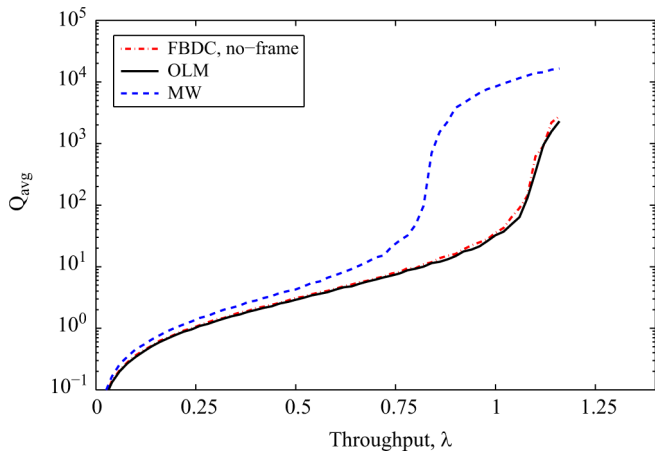


Fig. 9. Total average queue size versus the sum-throughput under the FBDC policy implemented without frames, the OLM, and the MW policies for $p_{10} = p_{01} = 0.30$.

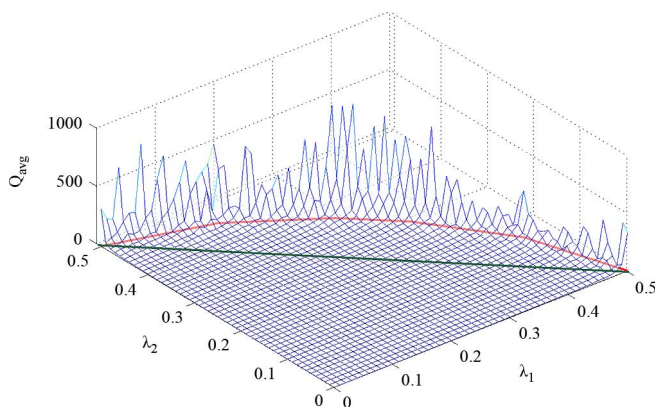


Fig. 10. Total average queue size for the FBDC policy for $p_{10} = p_{01} = 0.20$.

25 slots for $p_{10} = p_{01} = 0.20$ for the simple two-queue system introduced in Section III-A.1. The boundary of the stability region is shown by (red) lines on the two-dimensional $\lambda_1 - \lambda_2$ plane. We observe that the average queue lengths are small for all $(\lambda_1, \lambda_2) \in \mathbf{A}$, and the big jumps in queue lengths occur for points outside \mathbf{A} . Finally, the stability region is much larger than the stability region for the corresponding system with i.i.d. channel processes with the same steady state, which is represented by the diagonal line segment between the points $(0, 0.5)$ and $(0.5, 0)$.

V. CONCLUSION

We investigated the optimal scheduling problem for systems with *reconfiguration delays*, *time-varying channels*, and *interference constraints*. We characterized the stability region of the system in closed form for the case of i.i.d. channel processes and proved that a variable-size frame-based Max-Weight algorithm that makes scheduling decisions based on the queue lengths and the *average channel gains* is throughput-optimal. For the case of Markovian channels with memory, we characterized the system stability region using state-action frequencies that are stationary solutions to an MDP formulation. We developed the FBDC policy using state-action frequencies and variable frames that are functions of queue lengths and proved that it is throughput-optimal. Finally, we investigated the performance

of low-complexity Myopic algorithms that appear to have a similar throughput-delay performance to that of the FBDC policy in simulations.

The FBDC policy and state-action frequency approach provide a new framework for stability region characterization and throughput-optimal policy development for general network control systems, with or without reconfiguration delays. This framework is a first attempt at developing throughput-optimal algorithms for systems with time-varying channels and switching delays, and hopefully it will provide insight into designing scalable algorithms that can stabilize such systems. The Myopic control policies we considered constitute a first step in this direction.

In the future, we intend to study the joint scheduling and routing problem in multihop networks with time-varying channels and reconfiguration delays.

APPENDIX A

PROOF OF THEOREM 2

Consider the following χ_k -step queue evolution expression, similar to the 1-step queue evolution in (2). where χ_k is the length of the k th frame length, fixed given $\mathbf{Q}(t_k)$:

$$\begin{aligned} Q_\ell(t_k + \chi_k) &\leq \max \left\{ Q_\ell(t_k) - \sum_{\tau=0}^{\chi_k-1} C_\ell(t_k + \tau) \mathbf{I}_\ell(t_k + \tau), 0 \right\} \\ &\quad + \sum_{\tau=0}^{\chi_k-1} A_\ell(t_k + \tau). \end{aligned} \quad (20)$$

To see this, note that if $\sum_{\tau=0}^{\chi_k-1} \mathbf{I}_\ell(t_k + \tau) C_\ell(t_k + \tau)$, the total *service opportunity* given to link ℓ during the k th frame, is smaller than $Q_\ell(t_k)$, then we have an equality. Otherwise, the first term is 0 and we have an inequality. This is because some of the arrivals during the frame might depart before the end of the frame. Note that $C_\ell(t) \mathbf{I}_\ell(t)$ is not the actual number of departures from queue ℓ , but it is the service opportunity given to queue ℓ at time-slot t under activation $\mathbf{I}(t)$.

We first prove stability at the frame boundaries. Squaring both sides, using $\max(0, x)^2 \leq x^2, \forall x \in \mathbb{N} \cup \{0\}$, and $\mathbf{I}_\ell(t) C_\ell(t) \leq \mu_{\max}, \forall t$ we have

$$\begin{aligned} Q_\ell(t_k + \chi_k)^2 - Q_\ell(t_k)^2 &\leq \chi_k^2 \mu_{\max}^2 + \left(\sum_{\tau=0}^{\chi_k-1} A_\ell(t_k + \tau) \right)^2 \\ &\quad - 2Q_\ell(t_k) \left(\sum_{\tau=0}^{\chi_k-1} \mathbf{I}_\ell(t_k + \tau) C_\ell(t_k + \tau) - \sum_{\tau=0}^{\chi_k-1} A_\ell(t_k + \tau) \right). \end{aligned} \quad (21)$$

Define the quadratic Lyapunov function

$$L(\mathbf{Q}(t)) = \sum_{\ell=1}^L Q_\ell^2(t)$$

which represents a quadratic measure of the total load in the system at time-slot t . Define the χ_k -step conditional drift

$$\Delta_{\chi_k}(t_k) \triangleq \mathbb{E} [L(\mathbf{Q}(t_k + \chi_k)) - L(\mathbf{Q}(t_k)) | \mathbf{Q}(t_k)]$$

where the conditional expectation is over the randomness in arrivals, channel processes, and possibly the scheduling decisions.

For the VFMW algorithm, scheduling decisions are deterministic given the queue lengths at the beginning of the frames.

Summing (21) over the links, taking conditional expectation, using the assumption $\mathbb{E}[A_\ell(t)^2] \leq A_{\max}^2 \forall t$ (which also implies $\mathbb{E}[A_\ell(t_1)A_\ell(t_2)] \leq \sqrt{\mathbb{E}[A_\ell(t_1)]^2 \mathbb{E}[A_\ell(t_2)]^2} \leq A_{\max}^2$ for all t_1 and t_2), we have

$$\Delta_{\chi_k}(t_k) \leq LB\chi_k^2 + 2\chi_k \sum_{\ell} Q_\ell(t_k)\lambda_\ell - 2 \sum_{\ell} Q_\ell(t_k) \mathbb{E} \left[\sum_{\tau=0}^{\chi_k-1} \mathbf{I}_\ell(t_k + \tau) C_\ell(t_k + \tau) | \mathbf{Q}(t_k) \right] \quad (22)$$

where $B \doteq A_{\max}^2 + \mu_{\max}^2$, and we used the fact that the arrival processes are i.i.d. over time, independent of the queue lengths. The VFMW policy makes scheduling decisions once per frame based on the queue sizes at the beginning of the frame. Therefore, given $\mathbf{Q}(t_k)$, the scheduling variables $\mathbf{I}_\ell(t) = \mathbf{I}_\ell$ are deterministic and same for all $t \in \{t_k + T_r, \dots, t_{k+1} - 1\}$, and independent of $C_\ell(t)$, the value of the i.i.d. channel process at time t , for all time-slots t . Using this and the fact that the system is idle for the first T_r slots of the frame due to reconfiguration, namely, $\mathbf{I}_\ell(t) = 0, t \in \{t_k, \dots, t_k + T_r - 1\}, \forall \ell \in \mathcal{L}$, we have

$$\Delta_{\chi_k}(t_k) \leq LB\chi_k^2 + 2\chi_k \sum_{\ell} Q_\ell(t_k)\lambda_\ell - 2 \sum_{\ell} Q_\ell(t_k) \mathbf{I}_\ell \mathbb{E} \left[\sum_{\tau=T_r}^{\chi_k-1} C_\ell(t_k + \tau) | \mathbf{Q}(t_k) \right].$$

We have $\mathbb{E} \left[\sum_{\tau=T_r}^{\chi_k-1} C_\ell(t_k + \tau) | \mathbf{Q}(t_k) \right] = (\chi_k - T_r) \bar{C}_\ell$ because the channel processes are i.i.d. independent of the queue lengths for all time-slots. Therefore, we have

$$\Delta_{\chi_k}(t_k) \leq LB\chi_k^2 + 2\chi_k \sum_{\ell} Q_\ell(t_k)\lambda_\ell - 2(\chi_k - T_r) \sum_{\ell} Q_\ell(t_k) \bar{C}_\ell \mathbf{I}_\ell.$$

Now, we have $\sum_{\ell} Q_\ell(t_k) \bar{C}_\ell \mathbf{I}_\ell = \sum_{\ell} Q_\ell(t_k) \bar{C}_\ell \mathbf{I}_\ell^* (t_k)$ by definition of the VFMW policy in Algorithm 1. From Theorem 1, we have that for any arrival rate vector $\boldsymbol{\lambda}$ strictly inside \mathbf{A} , $\mathbf{C}^{-1}\boldsymbol{\lambda} \doteq (\lambda_1/\bar{C}_1, \dots, \lambda_L/\bar{C}_L)$ is dominated by some rate vector in the convex hull of the activation vectors \mathcal{I} , where \mathbf{C} is a diagonal matrix with elements $\bar{C}_1, \dots, \bar{C}_L$. Therefore, for any arrival rate vector $\boldsymbol{\lambda}$ that is strictly inside \mathbf{A} , there exist real numbers $\beta^1, \dots, \beta^{|\mathcal{I}|}$ such that $\beta^j \geq 0, \forall j \in 1, \dots, |\mathcal{I}|, \sum_{j=1}^{|\mathcal{I}|} \beta^j = 1 - \epsilon$ for some $\epsilon > 0$ and $\mathbf{C}^{-1}\boldsymbol{\lambda} = \sum_{j=1}^{|\mathcal{I}|} \beta^j \mathbf{I}^j, \forall \ell \in \mathcal{L}$ [8]. Therefore, we have

$$\begin{aligned} \Delta_{\chi_k}(t_k) &\leq LB\chi_k^2 + 2\chi_k \mathbf{Q}(t_k) \cdot \left(\sum_{j=1}^{|\mathcal{I}|} \beta^j \sum_{\ell} \mathbf{I}_\ell^j \bar{C}_\ell \right) \\ &= LB\chi_k^2 + 2\chi_k \left(\sum_{j=1}^{|\mathcal{I}|} \beta^j \sum_{\ell} Q_\ell(t_k) \bar{C}_\ell \mathbf{I}_\ell^j \right) \\ &\quad - 2(\chi_k - T_r) \sum_{\ell} Q_\ell(t_k) \bar{C}_\ell \mathbf{I}_\ell^* (t_k) \\ &\leq LB\chi_k^2 + 2\chi_k \left(\sum_{j=1}^{|\mathcal{I}|} \beta^j \sum_{\ell} Q_\ell(t_k) \bar{C}_\ell \mathbf{I}_\ell^* (t_k) \right) \\ &\quad - 2(\chi_k - T_r) \sum_{\ell} Q_\ell(t_k) \bar{C}_\ell \mathbf{I}_\ell^* (t_k) \end{aligned}$$

where we used the fact that $\sum_{\ell} Q_\ell(t_k) \bar{C}_\ell \mathbf{I}_\ell^* (t_k) \geq \sum_{\ell} Q_\ell(t_k) \bar{C}_\ell \mathbf{I}_\ell, \forall \mathbf{I} \in \mathcal{I}$ by definition of the VFMW policy in Algorithm 1. Changing the order of the summation in the second term on the right-hand side and using $\sum_{j=1}^{|\mathcal{I}|} \beta^j = 1 - \epsilon$, we have

$$\Delta_{\chi_k}(t_k) = \chi_k \left(LB\chi_k - 2 \left(\epsilon - \frac{T_r}{\chi_k} \right) \sum_{\ell} Q_\ell(t_k) \bar{C}_\ell \mathbf{I}_\ell^* (t_k) \right).$$

If $\chi_k = T_r + (\sum_{\ell} Q_\ell(t_k))^\alpha \leq \frac{T_r}{\epsilon}$, then $\Delta_{\chi_k}(t_k) \leq C_0$, where C_0 is a constant. Otherwise, there exists a small $\delta_1 > 0$ such that $\epsilon - \frac{T_r}{\chi_k} > \delta_1$. Hence, for $\sum_{\ell} Q_\ell(t_k) > \frac{T_r^{1/\alpha}(1-\epsilon)^{1/\alpha}}{\epsilon^{1/\alpha}}$, we have

$$\Delta_{\chi_k}(t_k) \leq \chi_k^2 LB - 2\chi_k \delta_1 \frac{\bar{C}_{\min}}{L} \sum_{\ell} Q_\ell(t_k)$$

where we also used the fact that $\sum_{\ell} Q_\ell(t_k) \bar{C}_\ell \mathbf{I}_\ell^* (t_k) \geq \frac{\bar{C}_{\min}}{L} \sum_{\ell} Q_\ell(t_k)$ with $\bar{C}_{\min} \doteq \min\{\bar{C}_1, \dots, \bar{C}_L\}$. Therefore, there exists a constant B_1 such that

$$\Delta_{\chi_k}(t_k) \leq B_1 - \delta \left(\sum_{\ell} Q_\ell(t_k) \right)^{1+\alpha} \quad (23)$$

where $\delta \triangleq \delta_1 \bar{C}_{\min}/L$. Taking expectations with respect to $\mathbf{Q}(t_k)$, writing a similar expression over the frame boundaries $t_k, k \in \{0, 1, 2, \dots, M\}$, summing them, and telescoping these expressions leads to

$$L(\mathbf{Q}(t_M)) - L(\mathbf{Q}(0)) \leq MB_1 - \delta \sum_{k=0}^{M-1} \mathbb{E} \left[\left(\sum_{\ell} Q_\ell(t_k) \right)^{1+\alpha} \right].$$

Using $L(\mathbf{Q}(t_M)) \geq 0$ and $L(\mathbf{Q}(0)) = 0$, we have

$$\frac{1}{M} \sum_{k=0}^{M-1} \mathbb{E} \left[\left(\sum_{\ell} Q_\ell(t_k) \right)^{1+\alpha} \right] \leq \frac{B_1}{\delta} < \infty.$$

This implies that

$$\limsup_{M \rightarrow \infty} \frac{1}{M} \sum_{k=0}^{M-1} \mathbb{E} \left[\left(\sum_{\ell} Q_\ell(t_k) \right)^{1+\alpha} \right] \leq \frac{B_1}{\delta} < \infty. \quad (24)$$

This establishes stability (as defined in Definition 4) at the frame boundaries $t_k, k \in \{0, 1, 2, \dots\}$.

Now, we have for all frames $k \in \{0, 1, 2, \dots\}$

$$\begin{aligned} &\sum_{\tau=0}^{\chi_k-1} \sum_i Q_i(t_k + \tau) \\ &\leq \sum_{\tau=0}^{\chi_k-1} \sum_i \left(Q_i(t_k) + \sum_{\tau_1=0}^{\chi_k-1} A_i(t_k + \tau_1) \right). \end{aligned}$$

Taking conditional expectation, we have

$$\sum_{\tau=0}^{\chi_k-1} \sum_i \mathbb{E} [Q_i(t_k + \tau) | \mathbf{Q}(t_k)] \leq \chi_k \sum_i Q_i(t_k) + \chi_k^2 \sum_i \lambda_i$$

where we used the fact that arrival processes are i.i.d. and independent of the queue lengths. Recalling $\chi_k = T_r + (\sum_i Q_i(t_k))^\alpha$ with $0 < \alpha < 1$, we have

$$\begin{aligned} & \sum_{\tau=0}^{\chi_k-1} \sum_i \mathbb{E}[Q_i(t_k + \tau) | \mathbf{Q}(t_k)] \leq \left(\sum_i Q_i(t_k) \right)^{1+\alpha} \\ & + T_r \sum_i Q_i(t_k) + \left(T_r + \left(\sum_i Q_i(t_k) \right)^\alpha \right)^2 \sum_i \lambda_i. \end{aligned}$$

Now, for any given large T , let K_T be the number of frames up to and including T . We have

$$\begin{aligned} & \sum_{t=0}^{T-1} \sum_i \mathbb{E}[Q_i(t)] \\ & \leq \sum_{k=0}^{K_T-1} \mathbb{E} \left[\left(\sum_i Q_i(t_k) \right)^{1+\alpha} + T_r \sum_i Q_i(t_k) \right] \\ & + \sum_{k=0}^{K_T-1} \mathbb{E} \left[\left(T_r + \left(\sum_i Q_i(t_k) \right)^\alpha \right)^2 \sum_i \lambda_i \right]. \end{aligned}$$

Dividing both sides by T , using $T > K_T$ for any T , taking the $\limsup_{T \rightarrow \infty}$ of both sides, using (24) and $0 < \alpha < 1$, we have

$$\limsup_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{i=1}^N \mathbb{E}[Q_i(t)] < \infty. \quad (25)$$

Therefore, the system is stable.

APPENDIX B PROOF OF THEOREM 4

We utilize the same Lyapunov function and the χ_k -step drift expression as in Appendix A, and the following expression follows similar to (22):

$$\begin{aligned} \Delta_{\chi_k}(t_k) & \leq LB\chi_k^2 + 2\chi_k \sum_\ell Q_\ell(t_k)\lambda_\ell \\ & - 2 \sum_\ell Q_\ell(t_k) \mathbb{E} \left[\sum_{\tau=0}^{\chi_k-1} D_\ell(t_k + \tau) | \mathbf{Q}(t_k) \right], \quad (26) \end{aligned}$$

where $B \doteq A_{\max}^2 + \mu_{\max}^2$ and $D_\ell(t) \doteq C_\ell(t)\mathbf{I}_\ell(t)$. Note that $\sum_{\tau=0}^{\chi_k-1} D_\ell(t_k + \tau)$ is the total *service opportunity* given to link ℓ during the k th frame, i.e., it denotes the link ℓ departures that would happen in the corresponding *saturated system* if we were to apply the *same* reconfiguration decisions over χ_k time-slots in the saturated system. We first prove stability at the frame boundaries. Recall the definition of the reward functions $\bar{r}_\ell(\mathbf{s}_t, \mathbf{a}_t)$ in (6), and let $\bar{r}_\ell(\mathbf{s}_t, \mathbf{a}_t)$ be the reward function associated with applying policy $\boldsymbol{\pi}^*$ given in the definition of the FBDC policy in Algorithm 3 to the saturated system. Let $\bar{r}_\ell(t)$ denote $\bar{r}_\ell(\mathbf{s}_t, \mathbf{a}_t)$ for notational simplicity, $\ell \in \mathcal{L}$. Note that $r_\ell(t)$ is equal to $D_\ell(t)$ since $D_\ell(t)$ is the *service opportunity* given to link ℓ at time-slot t . Now let $\mathbf{r}^* = (r_\ell^*)_\ell$ be the infinite horizon average rate associated with policy $\boldsymbol{\pi}^*$. Let \mathbf{x}^* be the optimal vector of state-action frequencies corresponding to $\boldsymbol{\pi}^*$. Define

³Note that all the norms $\|\cdot\|$ used in this proof and the rest of the paper denote the Euclidean norm, and $|\cdot|$ denotes the absolute value.

the time-average empirical reward from queue ℓ in the saturated system, $\hat{r}_{\chi_k, \ell}(t_k)$, $\ell \in \mathcal{L}$ by

$$\hat{r}_{\chi_k, \ell}(t_k) \doteq \frac{1}{\chi_k} \sum_{\tau=0}^{\chi_k-1} \bar{r}_\ell(t_k + \tau).$$

Similarly, define the time average empirical state-action frequency vector $\hat{\mathbf{x}}_{\chi_k}(t_k; \mathbf{s}, \mathbf{a})$

$$\hat{\mathbf{x}}_{\chi_k}(t_k; \mathbf{s}, \mathbf{a}) \doteq \frac{1}{\chi_k} \sum_{\tau=t_k}^{t_k + \chi_k - 1} I_{\{\mathbf{s}_\tau = \mathbf{s}, \mathbf{a}_\tau = \mathbf{a}\}} \quad (27)$$

where I_E is the indicator function of an event E , i.e., $I_E = 1$ if E occurs, and $I_E = 0$ otherwise. Using the definition of the reward functions in (6), we have that

$$\hat{r}_{\chi_k, \ell}(t_k) = \sum_{\mathbf{s} \in \mathcal{S}} \sum_{\mathbf{a} \in \mathcal{I}} \bar{r}_\ell(\mathbf{s}, \mathbf{a}) \hat{\mathbf{x}}_{\chi_k}(t_k; \mathbf{s}, \mathbf{a}), \quad \ell \in \mathcal{L}$$

and $\hat{\mathbf{r}}_{\chi_k}(t_k) = (\hat{r}_{\chi_k, \ell}(t_k))_\ell$. Similarly, we have

$$\mathbf{r}_\ell^* = \sum_{\mathbf{s} \in \mathcal{S}} \sum_{\mathbf{a} \in \mathcal{I}} \bar{r}_\ell(\mathbf{s}, \mathbf{a}) \mathbf{x}^*(\mathbf{s}, \mathbf{a}), \quad \ell \in \mathcal{L}.$$

Now we utilize the following key MDP theory result in [25, Lemma 4.1], which states that as χ_k increases, $\hat{\mathbf{x}}_{\chi_k}(t_k) = (\hat{\mathbf{x}}_{\chi_k}(t_k; \mathbf{s}, \mathbf{a}))_{\mathbf{s}, \mathbf{a}}$ converges to \mathbf{x}^* .

Lemma 2: For every choice of initial state distribution, there exist constants c_1 and c_2 such that³

$$\mathbb{P}(\|\hat{\mathbf{x}}_{\chi_k}(t_k) - \mathbf{x}^*\| \geq \delta_0) \leq c_1 e^{-c_2 \delta_0^2 \chi_k}, \quad \forall \chi_k \geq 1, \forall \delta_0 > 0.$$

Furthermore, convergence of $\hat{\mathbf{x}}_{\chi_k}(t_k)$ to \mathbf{x}^* is w.p. 1. This result applies in our system because every extreme point \mathbf{x}^* of \mathbf{X} can be attained by a stationary and deterministic policy that has a single irreducible recurrent class in its underlying Markov chain [25], [32].⁴ Due to the linear mapping from the state-action frequencies to the rewards, by Schwartz inequality, each component of $\hat{\mathbf{r}}_{\chi_k}(t_k)$ also converges to the corresponding component of \mathbf{r}^* . Therefore, we have that for every choice of initial state distribution, there exists constants c_1 and c_2 such that

$$\mathbb{P}(\|\hat{\mathbf{r}}_{\chi_k}(t_k) - \mathbf{r}^*\| \geq \delta_1) \leq c_1 e^{-c_2 \delta_1^2 \chi_k}, \quad \forall \chi_k \geq 1, \forall \delta_1 > 0. \quad (28)$$

Furthermore, convergence of $\hat{\mathbf{r}}_{\chi_k, \ell}(t_k)$ to \mathbf{r}^* is w.p. 1. Now let $R_{\chi_k}(t_k) \doteq \sum_\ell Q_\ell(t_k) \hat{r}_{\chi_k, \ell}(t_k)$ and $R^*(t_k) \doteq \sum_\ell Q_\ell(t_k) r_\ell^*$. We rewrite the drift expression

$$\begin{aligned} \frac{\Delta_{\chi_k}(t_k)}{2\chi_k} & \leq \frac{LB\chi_k}{2} + \sum_\ell Q_\ell(t_k)\lambda_\ell - \mathbb{E}[R_{\chi_k}(t_k) | \mathbf{Q}(t_k)] \\ & = \frac{LB\chi_k}{2} + \sum_\ell Q_\ell(t_k)\lambda_\ell - \sum_\ell Q_\ell(t_k)r_\ell^* \\ & + \mathbb{E}[R^*(t_k) - R_{\chi_k}(t_k) | \mathbf{Q}(t_k)]. \quad (29) \end{aligned}$$

⁴Note that in general multiple stationary-deterministic policies can yield the same optimal reward vector \mathbf{r}^* . Among these, we choose the one that forms a Markov chain with a single recurrent class.

Now we bound the last term. For all $\delta_2 > 0$, we have

$$\begin{aligned}
& \mathbb{E}[|R^*(t_k) - R_{\chi_k}(t_k)| | \mathbf{Q}(t_k)] \\
&= \mathbb{E}[|R^*(t_k) - R_{\chi_k}(t_k)| | \mathbf{Q}(t_k), |R^*(t_k) - R_{\chi_k}(t_k)| \geq \delta_2 || \mathbf{Q}(t_k) ||] \\
&\quad \cdot \mathbb{P}(|R^*(t_k) - R_{\chi_k}(t_k)| \geq \delta_2 || \mathbf{Q}(t_k) || | \mathbf{Q}(t_k)) \\
&\quad + \mathbb{E}[|R^*(t_k) - R_{\chi_k}(t_k)| | \mathbf{Q}(t_k), |R^*(t_k) - R_{\chi_k}(t_k)| < \delta_2 || \mathbf{Q}(t_k) ||] \\
&\quad \cdot \mathbb{P}(|R^*(t_k) - R_{\chi_k}(t_k)| < \delta_2 || \mathbf{Q}(t_k) || | \mathbf{Q}(t_k)) \\
&\leq \mu_{\max} \left(\sum_{\ell} Q_{\ell}(t_k) \right) \mathbb{P}(|R^*(t_k) - R_{\chi_k}(t_k)| \geq \delta_2 || \mathbf{Q}(t_k) || | \mathbf{Q}(t_k)) \\
&\quad + \delta_2 || \mathbf{Q}(t_k) || \tag{30}
\end{aligned}$$

where we bound the first expectation by $\sum_{\ell} Q_{\ell}(t_k)$ by using $\|\mathbf{r}^*\| < \mu_{\max}$, the second expectation by $\delta_2 || \mathbf{Q}(t_k) ||$, and the second probability by 1. By Schwartz inequality, we have

$$\begin{aligned}
& \mathbb{P}(|R^*(t_k) - R_{\chi_k}(t_k)| \geq \delta_2 || \mathbf{Q}(t_k) || | \mathbf{Q}(t_k)) \\
&\quad \leq \mathbb{P}(|\mathbf{r}^* - \hat{\mathbf{r}}_{\chi_k}(t_k)| \geq \delta_2 || \mathbf{Q}(t_k) ||). \tag{31}
\end{aligned}$$

Using (28) and (31) in (30), we have that there exist constants c_1 and c_2 such that

$$\begin{aligned}
& \mathbb{E}[|R^*(t_k) - R_{\chi_k}(t_k)| | \mathbf{Q}(t_k)] \\
&\quad \leq \left(\sum_{\ell} Q_{\ell}(t_k) \right) c_1 e^{-c_2 \delta_2^2 \chi_k} + \delta_2 || \mathbf{Q}(t_k) ||
\end{aligned}$$

for all $\chi_k \geq 1$ and $\delta_2 > 0$. Hence, using $|| \mathbf{Q}(t_k) || \leq \sum_{\ell} Q_{\ell}(t_k)$, we bound (29) as

$$\begin{aligned}
\frac{\Delta_{\chi_k}(t_k)}{2\chi_k} &\leq \frac{LB\chi_k}{2} + \sum_{\ell} Q_{\ell}(t_k)\lambda_{\ell} \\
&\quad - \sum_{\ell} Q_{\ell}(t_k)r_{\ell}^* + \left(\sum_{\ell} Q_{\ell}(t_k) \right) \left(c_1 e^{-c_2 \delta_2^2 \chi_k} + \delta_2 \right).
\end{aligned}$$

Therefore, calling $\delta(\chi_k) \doteq c_1 e^{-c_2 \delta_2^2 \chi_k} + \delta_2$, we have

$$\begin{aligned}
\frac{\Delta_{\chi_k}(t_k)}{2\chi_k} &\leq \frac{LB\chi_k}{2} + \sum_{\ell} Q_{\ell}(t_k)\lambda_{\ell} \\
&\quad - \sum_{\ell} Q_{\ell}(t_k)r_{\ell}^* + \delta \sum_{\ell} Q_{\ell}(t_k).
\end{aligned}$$

Now for $\boldsymbol{\lambda}$ strictly inside the stability region $\mathbf{\Lambda}$, there exist a small $\xi' > 0$ such that $\boldsymbol{\lambda} + \xi' \cdot \mathbf{1} = \mathbf{r}$, for some $\mathbf{r} \in \mathbf{\Lambda}_s$. Utilizing this and the fact that $\sum_{\ell} Q_{\ell}(t)(r_{\ell} - r_{\ell}^*) \leq 0$ by definition of the FBDC policy in Algorithm 3, we have

$$\frac{\Delta_{\chi_k}(t_k)}{2\chi_k} \leq \frac{LB\chi_k}{2} - \xi' \sum_{\ell} Q_{\ell}(t_k) + \delta \sum_{\ell} Q_{\ell}(t_k). \tag{32}$$

For $\delta_2 < \xi$ and the frame length χ_k large enough, we have $\delta(\chi_k) < \xi' + \xi$ where $\xi > 0$. Therefore, we have

$$\Delta_{\chi_k}(t_k) \leq LB\chi_k^2 - 2\xi\chi_k \sum_{\ell} Q_{\ell}(t_k). \tag{33}$$

Therefore, similar to (23), we have a negative drift for large queue lengths if χ_k is a sublinear function of the sum-queue

lengths. For example, for $\chi_k = T_r + \left(\sum_{\ell} Q_{\ell}(t_k) \right)^{\alpha}$, we have that there exists a constant B_1 such that

$$\Delta_{\chi_k}(t_k) \leq B_1 - \xi \left(\sum_{\ell} Q_{\ell}(t_k) \right)^{1+\alpha}. \tag{34}$$

The rest of the proof follows similar to the proof of VFMW Algorithm in Appendix A.

REFERENCES

- [1] S. Ahmad, L. Mingyan, 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.
- [2] S. Ahmad and M. Liu, "Multi-channel opportunistic access: A case of restless bandits with multiple plays," in *Proc. Allerton*, Oct. 2009, pp. 1361–1368.
- [3] I. F. Akyildiz and X. Wang, *Wireless Mesh Networks*. Hoboken, NJ, USA: Wiley, 2009.
- [4] E. Altman, P. Konstantopoulos, and Z. Liu, "Stability, monotonicity and invariant quantities in general polling systems," *Queueing Syst.*, vol. 11, pp. 35–57, Mar. 1992.
- [5] D. P. Bertsekas, A. Nedic, and A. E. Ozdaglar, *Convex Analysis and Optimization*. Boston, MA, USA: Athena Scientific, 2003.
- [6] L. Blake and M. Long, *Antennas: Fundamentals, Design, Measurement*. Herndon, VA, USA: SciTech, 2009.
- [7] D. Bertsimas and J. Tsitsiklis, *Introduction to Linear Optimization*. Belmont, MA, USA: Athena Scientific, 1997.
- [8] A. Brzezinski and E. Modiano, "Dynamic reconfiguration and routing algorithms for IP-over-WDM networks with stochastic traffic," *J. Lightw. Technol.*, vol. 23, no. 10, pp. 3188–3205, Oct. 2005.
- [9] A. Brzezinski, "Scheduling algorithms for throughput maximization in data networks," Ph.D. dissertation, MIT, Cambridge, MA, USA, 2007.
- [10] P. Chaporkar, K. Kar, and S. Sarkar, "Throughput guarantees through maximal scheduling in wireless networks," in *Proc. Allerton*, Sep. 2005.
- [11] G. Celik, "Scheduling algorithms for throughput maximization in time-varying networks with reconfiguration delays," Ph.D. dissertation, MIT, Cambridge, MA, USA, 2012.
- [12] G. D.Çelik and E. Modiano, "Scheduling in networks with time-varying channels and reconfiguration delay," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 990–998.
- [13] G. D.Çelik, L. B. Le, and E. Modiano, "Scheduling in parallel queues with randomly varying connectivity and switchover delay," in *Proc. IEEE INFOCOM Mini-Conf.*, Apr. 2011, pp. 316–320.
- [14] G.Çelik, S. Borst, P. Whiting, and E. Modiano, "Variable frame based max-weight algorithms for networks with switchover delay," in *Proc. IEEE ISIT*, Jun. 2011, pp. 2537–2541.
- [15] A. Eryilmaz, A. Ozdaglar, and E. Modiano, "Polynomial complexity algorithms for full utilization of multi-hop wireless networks," in *Proc. IEEE INFOCOM*, May 2007, pp. 499–507.
- [16] R. G. Gallager, *Discrete Stochastic Processes*. Norwell, MA, USA: Kluwer, 1996.
- [17] L. Georgiadis, M. Neely, and L. Tassiulas, *Resource Allocation and Cross-Layer Control in Wireless Networks*. Delft, The Netherlands: Now, 2006.
- [18] Y. Hawar, E. Farag, S. Vanakayala, R. Pauls, X. Yang, S. Subramanian, P. Sadhanala, L. Yang, B. Wang, Z. Li, H. Chen, Z. Lu, D. Clark, T. Fosket, P. Mallela, M. Shelton, D. Laurens, T. Salaun, L. Gougeon, N. Aubourg, H. Morvan, N. Le Henaff, G. Prat, F. Charles, C. Creach, Y. Calvez, and P. Butel, "3G UMTS wireless system physical layer: Baseband processing hardware implementation perspective," *IEEE Commun. Mag.*, vol. 44, no. 9, pp. 52–58, Sep. 2006.
- [19] J. Padhye, V. N. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," *Wireless Netw.*, vol. 11, pp. 471–487, Sep. 2005.
- [20] K. Jagannathan, S. Mannor, I. Menache, and E. Modiano, "A state action frequency approach to throughput maximization over uncertain wireless channels," in *Proc. IEEE INFOCOM Mini-Conf.*, Apr. 2011, pp. 491–495.
- [21] K. Kar, X. Luo, and S. Sarkar, "Throughput-optimal scheduling in multichannel access point networks under infrequent channel measurements," in *Proc. IEEE INFOCOM*, May 2007, pp. 1640–1648.
- [22] L. B. Le, E. Modiano, C. Joo, and N. B. Shroff, "Longest-queue-first scheduling under SINR interference model," in *Proc. ACM MobiHoc*, Sep. 2010, pp. 41–50.

- [23] C. Li and M. Neely, "On achievable network capacity and throughput-achieving policies over Markov on/off channels," in *Proc. WiOpt*, Jun. 2010.
- [24] X. Lin and N. B. Shroff, "The impact of imperfect scheduling on cross-layer rate control in wireless networks," in *Proc. IEEE INFOCOM*, Mar. 2005, pp. 1804–1814.
- [25] S. Mannor and J. N. Tsitsiklis, "On the empirical state-action frequencies in Markov decision processes under general policies," *Math. Oper. Res.*, vol. 30, no. 3, pp. 545–561, Aug. 2005.
- [26] N. McKeown, A. Mekkittikul, V. Anantharam, and J. Walrand, "Achieving 100% throughput in an input-queued switch," *IEEE Trans. Commun.*, vol. 47, no. 8, pp. 1260–1272, Aug. 1999.
- [27] E. Modiano and R. Barry, "A novel medium access control protocol for WDM-based LAN's and access networks using a master/slave scheduler," *J. Lightw. Technol.*, vol. 18, no. 4, pp. 461–468, Apr. 2000.
- [28] E. Modiano, D. Shah, and G. Zussman, "Maximizing throughput in wireless networks via gossip," in *Proc. ACM SIGMETRICS/Perform.*, Jun. 2006, pp. 27–38.
- [29] M. J. Neely, *Stochastic Network Optimization With Application to Communication and Queuing Systems*. San Rafael, CA, USA: Morgan & Claypool, 2010.
- [30] 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.
- [31] M. J. Neely, E. Modiano, and C. E. Rohrs, "Power allocation and routing in multi-beam satellites with time-varying channels," *IEEE Trans. Netw.*, vol. 11, no. 1, pp. 138–152, Feb. 2003.
- [32] M. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Hoboken, NJ, USA: Wiley, 2005.
- [33] V. Raman and N. Vaidya, "Short: A static-hybrid approach for routing real time applications over multichannel, multihop wireless networks," in *Proc. WWIC*, Jun. 2006, pp. 77–94.
- [34] D. Shah and D. J. Wischik, "Optimal scheduling algorithms for input-queued switches," in *Proc. IEEE INFOCOM*, Mar. 2006, pp. 1–11.
- [35] A. Pantelidou, A. Ephremides, and A. Tits, "A cross-layer approach for stable throughput maximization under channel state uncertainty," *Wireless Netw.*, vol. 15, no. 5, pp. 555–569, Jul. 2009.
- [36] A. L. Stolyar, "Maxweight scheduling in a generalized switch: State space collapse and workload minimization in heavy traffic," *Ann. Appl. Probab.*, vol. 14, no. 1, pp. 1–53, 2004.
- [37] L. Tassiulas, "Adaptive back-pressure congestion control based on local information," *IEEE Trans. Autom. Control*, vol. 40, no. 2, pp. 236–250, Feb. 1995.
- [38] L. Tassiulas and S. Papavassiliou, "Optimal anticipative scheduling with asynchronous transmission opportunities," *IEEE Trans. Autom. Control*, vol. 40, no. 12, pp. 2052–2062, Dec. 1995.
- [39] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. Autom. Control*, vol. 37, no. 12, pp. 1936–1948, Dec. 1992.
- [40] 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.
- [41] A. Tolkachev, V. Denisenko, A. Shishlov, and A. Shubov, "High gain antenna systems for millimeter wave radars with combined electronical and mechanical beam steering," in *Proc. IEEE Symp. Phased Array Syst. Technol.*, Oct. 2006, pp. 266–271.
- [42] R. Vedantham, S. Kakumanu, S. Lakshmanan, and R. Sivakumar, "Component based channel assignment in single radio, multichannel ad hoc networks," in *Proc. MobiCom*, Sep. 2006, pp. 378–389.
- [43] V. M. Vishnevskii and O. V. Semenova, "Mathematical methods to study the polling systems," *Autom. Remote Control*, vol. 67, no. 2, pp. 173–220, Feb. 2006.
- [44] X. Wu and R. Srikant, "Bounds on the capacity region of multi-hop wireless networks under distributed greedy scheduling," in *Proc. IEEE INFOCOM*, Mar. 2006.
- [45] L. Ying and S. Shakkottai, "On throughput-optimality with delayed network-state information," in *Proc. ITA*, Jan. 2008, pp. 339–344.
- [46] M. Yun, Y. Zhou, A. Arora, and H. Choi, "Channel-assignment and scheduling in wireless mesh networks considering switching overhead," in *Proc. IEEE ICC*, Jun. 2009, pp. 1–6.
- [47] Q. Zhao, B. Krishnamachari, and K. Liu, "On myopic sensing for multi-channel opportunistic access: Structure, optimality, and performance," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 5431–5440, Dec. 2008.



Güner D.Çelik (S'06–M'12) received the B.Sc. degree in electrical and electronics engineering from the Middle East Technical University (METU), Ankara, Turkey, with second highest ranking in 2005, and the S.M. and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, in 2007 and 2012, respectively, under the supervision of Prof. Eytan Modiano.

In 2010, he was a Visiting Researcher with the Mathematics of Networks and Complex Systems Research Group, Bell Labs, Murray Hill, NJ, USA, and in 2007, he joined the Microsoft Research (MSR) Systems and Networking Group, Cambridge, U.K., as a Research Intern. His research interests are in network control and queuing theory. In particular, he is currently working on scheduling and resource allocation for network control and optimization and dynamic server scheduling in queuing systems.



Eytan Modiano (S'90–M'93–SM'00–F'12) received the B.S. degree in electrical engineering and computer science from the University of Connecticut, Storrs, CT, USA, in 1986, and the M.S. and Ph.D. degrees in electrical engineering from the University of Maryland, College Park, MD, USA, in 1989 and 1992, respectively.

He was a Naval Research Laboratory Fellow between 1987 and 1992 and a National Research Council Post Doctoral Fellow from 1992 to 1993. Between 1993 and 1999, he was with the Massachusetts Institute of Technology (MIT) Lincoln Laboratory, Lexington, MA, USA, where he was a project leader for MIT Lincoln Laboratory's Next Generation Internet (NGI) project. Since 1999, he has been on the faculty with MIT; where he is a Professor with the Department of Aeronautics and Astronautics and the Laboratory for Information and Decision Systems (LIDS). His research is on communication networks and protocols with emphasis on satellite, wireless, and optical networks.

Prof. Modiano is currently an Associate Editor for the IEEE/ACM TRANSACTIONS ON NETWORKING. He had served as Associate Editor for the IEEE TRANSACTIONS ON INFORMATION THEORY, and as Guest Editor for the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS special issue on WDM network architectures; the *Computer Networks* special issue on broadband Internet access; the *Journal of Communications and Networks* special issue on wireless ad hoc networks; and the JOURNAL OF LIGHTWAVE TECHNOLOGY special issue on optical networks. He was the Technical Program co-chair for IEEE Wiopt 2006, IEEE INFOCOM 2007, and ACM MobiHoc 2007. He is the co-recipient of the SIGMETRICS 2006 Best Paper Award for the paper "Maximizing Throughput in Wireless Networks via Gossiping," and the WiOpt 2005 Best Student Paper Award for the paper "Minimum Energy Transmission Scheduling Subject to Deadline Constraints."