Throughput Optimal Scheduling for Wireless Downlinks with Reconfiguration Delay

Vineeth Bala Sukumaran
vineethbs@gmail.com
Department of Avionics, Indian Institute of Space Science and Technology.

***Abstract*—We consider wireless downlinks where the base station dynamically switches between different users in order to transmit data intended for the respective users. When the base station switches from serving one user to another, there is a reconfiguration delay. For such wireless downlinks with reconfiguration delay we consider the problem of throughput optimal scheduling. We propose the 1-lookahead scheduling policy and analytically show that it is throughput optimal. We obtain the 1-lookahead policy by using an approximate solution to a Markov decision process formulation of the scheduling problem. The approximate solution is also used to explain the biased max-weight form of 1-lookahead as well as an existing policy.**

***Index Terms*—Wireless downlink, Random connectivity, Reconfiguration delay, Stability region, Throughput optimality**

I. Introduction

We consider a wireless downlink model where the base station switches between different users in order to transmit data intended for the respective users. When switching between users, the logical links between the user and the base station need to be configured. For example, the user's *state* may need to be changed from an idle to an active state. This incurs a reconfiguration delay, which is the delay between the time at which the base station scheduler decides to serve the user and the time at which the actual data transmission starts. We note that such reconfiguration delay also arises in other cases: such as satellite systems with mechanically steered antennae, electronic beamforming, optical routers [1], and radio transceivers [3]. The other important feature of such systems is that the service of a user is also affected by the connectivity of that user to the service station. For example, the connectivity of the user is through a wireless channel subject to fading, therefore the connectivity is random over time. Motivated by such scenarios, we consider the stability region and throughput optimal scheduling for a wireless downlink model with random connectivity over time and reconfiguration delay (see Fig. 1).

For wireless networks with reconfiguration delay max-weight policies [7] are not throughput optimal since such policies switch between queues very frequently [2]. Prior work in [1], [2], and [4] had proposed heuristic throughput optimal policies for such systems. Celik et al. [2] proposed the variable frame max-weight (VFMW) policy which reduces reconfiguration delay overhead by restricting switching to happen only at the ends of scheduling frames for wireless networks with random connectivity and reconfiguration delay. Hsieh et al. [4] proposed the queue biased max weight (QBMW) policy that is throughput optimal for a queueing system with reconfiguration delay. The average delay performance of VFMW policies was improved by the QBMW policy. However, why the specific biased form of the QBMW policy was needed was not addressed in their paper. In this paper we propose the 1-lookahead policy, which is another throughput optimal policy for systems with reconfiguration delay. In contrast to VFMW and QBMW, we use a formal Markov decision theoretic formulation to analytically motivate the need for the bias term. The same approach can be used to motivate the bias term used for QBMW policies.

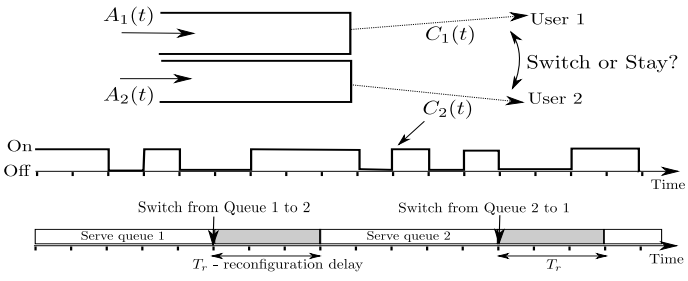

Fig. 1. An example wireless downlink with two users and a base station (BS). The data for the users have random packet arrivals $A_i(t)$ into their packet buffers at the BS. The BS can transmit at most one packet to one user from its queue in a slot. The connections between the BS and the users are randomly on or off across time (slots). When the BS switches from serving one user to another there is a reconfiguration delay of T_r slots.

Outline, Contributions, and Notation: In Section II we discuss the queueing model that we use for analyzing the wireless downlink shown in Fig. 1 as well as its stability region. Our main contribution in this paper is the analytically well-motivated throughput optimal 1-lookahead policy, which we propose in Section III. We then discuss a Markov decision process formulation which is used to motivate the definition of 1-lookahead policies in Section IV. Our first secondary contribution is that we explain the bias term appearing in the QBMW policy which has a similar form as 1-lookahead. Another secondary contribution is in the proof of throughput optimality of 1-lookahead and its relation to the proof of throughput optimality of QBMW; we provide simplifications and corrections to the proof in [4]. In this paper, all vectors are column vectors and vector transposes are denoted by $(.)^T$.

II. System Model and Problem Formulation

We consider a system of N parallel queues served by a single server to model a wireless downlink for N users[1] (see Fig. 1

[1]We note that the model can also apply to a wireless uplink but with additional assumptions on the availability of queue length information at a centralized scheduler.

for an example with $N = 2$). The system evolves in slotted time with the slots indexed by $t \in \{0, 1, 2, \ldots\}$. In each slot t, a random number $A_i(t)$ of packets arrive to the base station (BS) destined for the i^{th} user, for every $i \in \{1, 2, \ldots, N\}$. For every user, these packets are queued in a separate infinite length buffers at the BS. We assume that the random process $(A_i(t), t \geq 0)$ is an independent and identically distributed (IID) process (e.g., $A_i(t)$ can be modelled by a Bernoulli process). The arrival processes to different buffers are also assumed to be independent. We denote the arrival rate $\mathbb{E}A_i(0)$ as λ_i and the column vector $(\lambda_1, \ldots, \lambda_N)^T$ as $\boldsymbol{\lambda}$.

The BS scheduler decides which user's queue is served in a slot; $S_i(t) = 1$ if the scheduler decides to serve user i in slot t and 0 otherwise. We assume that $\sum_{i=1}^{N} S_i(t) = 1$, i.e., at most one user can be served in a slot. We assume that there is a connection random process $(C_i(t) \in \{0, 1\}, t \geq 0)$ associated with user i that models whether the BS is connected to user i in slot t. We assume that the the processes $(C_i(t), t \geq 0)$ are IID. We denote the average connection rate of queue i as μ_i, i.e., $\mathbb{E}C_i(0) = \mu_i$. We also assume that $(C_i(t), t \geq 0)$ and $(C_j(t), t \geq 0)$ for two different queues i and j are independent. We assume that the channel connectivity $C_j(t), \forall j$ is not known to the scheduler at the start of slot t[2]. We assume that if the BS is connected to user i and the scheduler decides to serve user i, then at most one packet is removed from queue i in slot t.

We assume that there is a reconfiguration delay of T_r slots (see Fig. 1) if the server switches from serving one queue to another. We assume that $T_r \geq 1$. We denote the number of slots to finish reconfiguration at time t by $R(t)$. If the scheduler switches from one queue to the another at slot t, $R(t) = T_r$ and then $R(t)$ decrements by one for every slot until $R(t) = 0$ if the scheduler stays with the queue that it has switched to. Once reconfiguration is finished $R(t)$ stays at zero until the scheduler switches to another queue. We note that a packet will be removed from the i^{th} queue in the t^{th} slot only if $C_i(t)S_i(t)\mathbb{I}_{\{R(t)=0\}} = 1$. We define $I_i(t) = C_i(t)S_i(t)\mathbb{I}_{\{R(t)=0\}}$, which is called the service opportunity for queue i in slot t. We denote the number of packets in the i^{th} queue at the beginning of slot t as $Q_i(t)$. Then,

$$Q_i(t+1) = (Q_i(t) - I_i(t))^+ + A_i(t), \tag{1}$$

where $(x)^+ = \max(x, 0)$. We note that the packets which arrive in slot t are assumed to stay in the system for at least one slot.

A policy μ is the sequence of decisions $(\mathbf{S}(1), \mathbf{S}(2), \ldots)$, where $\mathbf{S}(t)$ is the vector $(S_1(t), \ldots, S_N(t))^T$. The time average total queue length under a policy μ is $\limsup_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=0}^{T-1}\sum_{i=1}^{N} Q_i(t)\right]$ and is denoted as $\overline{q}(\mu)$. The queueing system with an arrival rate $\boldsymbol{\lambda}$ is said to be stable under the policy μ if the time average total queue length $\overline{q}(\mu)$ is finite. The stability region Λ of the system is the set of arrival rate vectors $\boldsymbol{\lambda}$, where for each $\boldsymbol{\lambda}$ there exists a policy μ (possibly dependent on $\boldsymbol{\lambda}$) such that the queueing system is stable.

From [3], we have that if $\boldsymbol{\lambda} \in \Lambda$ then there exists $\beta_j, \forall j$ such that $\beta_j \geq 0$, $\sum_j \beta_j \leq 1$ such that

$$\boldsymbol{\lambda} \leq \sum_j \beta_j I^{(j)} \mu_j,$$

where $I^{(j)}$ is a column vector of length N with 1 at the j^{th} position and zero otherwise. It is known that the VFMW policy from [2] can be used to achieve any point in the stability region Λ. We note that both VFMW and QBMW are policies that are throughput optimal for the above queueing system. In the next section, we propose another throughput optimal policy 1-lookahead; the form of this policy is analytically motivated using a Markov decision theoretic formulation.

III. The 1-lookahead policy

For the queueing model, any scheduling policy at the start of a slot, needs to decide whether to keep serving the current queue or to switch to any other queue. Intuitively, we should serve a queue with large queue length, so that the queue length can be reduced, and largest possible throughput, since the reduction in queue length would be the most. This is the intuition behind the decision rule that is proposed as the 1-lookahead policy. We define our 1-lookahead policy in Algorithm 1. We assume that the server is serving queue i at $t-1$ in Algorithm 1. For the 1-lookahead policy, the weights $W_i(t)$ (or $W_j(t)$) can be calculated as the expected sum throughput that can be obtained in the immediate future consisting of $T_r + 1$ slots under the decision of staying with queue i (or switching to the j^{th} queue). Then, if we stay with queue i at t, the expected sum throughput, i.e., $W_i(t)$ is given by $(1 + T_r)\mu_i$. If we decide to switch to queue j at t, then the expected sum throughput, i.e., $W_j(t)$ is given by μ_j. We find that a heuristic modification to the weight $W_i(t)$ is required to show that the 1-lookahead policy is in fact throughput optimal. We redefine $W_i(t)$ as $\left(1 + \frac{T_r}{F(\mathbf{Q}(t))}\right)\mu_i$, where $F(\mathbf{Q}(t)) = \max(1, (\sum_i Q_i(t))^\alpha)$, where $\alpha \in (0, 1)$.

Algorithm 1 1-lookahead policy

1: Calculate the weights $W_j(t)$, $j \in \{1, 2, \ldots, N\}$.
2: If $W_i(t)Q_i(t) < W_j(t)Q_j(t)$, then switch to the j^{th} queue, else stay with the i^{th} queue.

A natural question that arises in the definition of 1-lookahead policies is why we are looking at the expected total throughput for $T_r + 1$ slots and not $T_r + m$ slots for some $m \geq 1$. In the next section we motivate this and the exact form for the 1-lookahead policy by considering a Markov decision process (MDP) formulation for the problem of minimizing time average total queue length for the queueing system discussed in our paper. We show that the 1-lookahead policy arises from a heuristic approximate solution to the average cost optimality equation for a Markov decision process formulated for the queueing system in our paper. The reason why $m = 1$ is related to the need for keeping the policy specification simple.

[2]Since $(C_i(t))$ are IID, this information is not useful.

We now show that the 1-lookahead policy has finite average queue length for any arrival rate vector within the stability region of the system and is therefore throughput optimal.

Theorem III.1. *For IID channel connection processes, for any arrival rate that is within the stability region, the 1-lookahead policy has finite average queue length.*

The proof of this theorem is presented in Appendix A. We note that the proof of this theorem borrows ideas from the proof of the throughput optimality of QBMW in [4].

Comparison with QBMW: We note that the form of 1-lookahead policy is similar to that of the QBMW policy. However, for the QBMW policy the weight $W_i(t)$ is calculated as $1 + \frac{T_r}{F(\mathbf{Q}(t_k))}$ where t_k is the time of the last switch before t. Since the form of 1-lookahead is almost the same as that of QBMW, the derivation of the weight terms for the 1-lookahead given in the next section also motivates the form of the QBMW policy. We also note that the proof of throughput optimality of QBMW relies upon showing that the drift of a Lyapunov function, defined as the sum of squared queue lengths in a slot, is negative when considered over frames (multiple slots). We expect that during reconfiguration delay after a switch, since there is no service, the Lyapunov drift would not be negative. Hsieh et al. [4] define a *frame size parameter* $\tilde{T}_k$ in order to show that the drift is indeed negative when considered over $\tilde{T}_k$ slots which include the reconfiguration delay. However, the proof in [4] does not consider what happens when $\tilde{T}_k < T_r$. We address this problem in our paper.

IV. Markov decision process formulation

We consider a two queue system here for ease of exposition. We note that instead of finding a scheduling policy such that the average queue length under that policy is just finite we reformulate our objective to find a scheduling policy that minimizes

$$\limsup_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\left[Q_1(t) + Q_2(t)\right]. \tag{2}$$

In order to find a scheduling policy that minimizes (2) defined above, we use a Markov decision process (MDP) formulation (see [8], [9]). The state of the MDP is defined to be $\mathbf{S}(t) = (Q_1(t), Q_2(t), R(t), M(t))$, where $M(t) \in \{0, 1\}$ indicates whether the first queue or second queue has been in service at slot $t-1$ respectively. The state space of the MDP is the Cartesian product of the state spaces of the individual components, i.e., $\mathbb{Z}_+ \times \mathbb{Z}_+ \times \{0, \ldots, T_r\} \times \{0, 1\}$. We denote a specific state vector value as $\mathbf{s} = (q_1, q_2, r, m)^T$. The action taken at slot t is $\gamma(t)$ which is defined as either staying with the current queue ($\gamma(t) = 0$) or switching to the other queue ($\gamma(t) = 1$). The evolution of the MDP's state from slot to slot, i.e., $\mathbf{S}(t)$ to $\mathbf{S}(t+1)$, is defined in terms of its components as follows:

1) if $\gamma(t) = 0$ and $R(t) > 0$ then $R(t+1) = R(t) - 1$ and since $T_r \geq 1$ there is no service from the $M(t)^{th}$ queue; we denote this as $I_{M(t)}(t) = 0$,
2) if $\gamma(t) = 0$ and $R(t) = 0$ then we have $I_{M(t)}(t) = 1$,
3) if $\gamma(t) = 1$ then $R(t) = T_r$ and $I_{M(t)}(t) = 0$; $M(t)$ changes to the queue that was switched to,
4) $Q_i(t+1) = \max(Q_i(t) - I_i(t), 0) + A_i(t)$.

Since we are interested in minimizing the time average expected total queue length, the single stage cost of the MDP at slot t is chosen as $Q_1(t) + Q_2(t)$. Suppose the average cost optimality equation (ACOE) [9, Chapter 6] exists[3] for the MDP. The ACOE is of the form [9, Chapter 6, Theorem 6.3.1]

$$h(\mathbf{s}) = \min_{\gamma\in\{0,1\}} \left\{ q_1 + q_2 - g^* + \mathbb{E}\left[h(\mathbf{S}^{(+\mathbf{1})})|\mathbf{s}\right] \right\}, \tag{3}$$

where $h(\mathbf{s})$ is the relative value function; which is a function of the state $\mathbf{s} = (q_1, q_2, r, m)^T$, g^* is the optimal minimum average cost (or sum of average queue lengths) and $\mathbf{S}^{(+\mathbf{1})}$ is the state that the MDP evolves to in one step starting from $\mathbf{s}$ according to the MDP evolution described above. We note that $q_1 + q_2$ is the single stage cost when the state of the MDP is $\mathbf{s}$.

The possible actions which can be taken when the state is $\mathbf{s}$ is: (a) 0 (stay with the current queue) or (b) 1 (switch to the other queue). The optimal policy is a stationary policy that chooses an action $\gamma \in \{0, 1\}$ in order to minimize the expression within the minimization in the RHS of (3). This optimal stationary policy prescribes an action γ as a function $\gamma(\mathbf{s})$ of the state $\mathbf{s}$. Note that if the function $h(\mathbf{s})$ is known then the optimal stationary policy can be completely characterized. However, in (3) both $h(\mathbf{s})$ and g^* are not known. In most cases, the above functional equation cannot be solved analytically for $h(\mathbf{s})$ and g^*.

Value iteration [9, Section 6.6] is an iterative procedure that can be used to obtain a solution to the ACOE. We let $V_1(\mathbf{s}) = q_1 + q_2$. We define

$$V_{n+1}(\mathbf{s}) = \min_{\gamma\in\{0,1\}} \left\{ q_1 + q_2 + \mathbb{E}\left[V_n(\mathbf{S}^{(+1)})|\mathbf{s}\right] \right\}. \tag{4}$$

We recall that $\mathbf{S}^{(+1)}$ is the state that the MDP evolves to in a single slot starting from state $\mathbf{s}$. We note that $V_n(\mathbf{s})$, called the value function, is the minimum expected cumulative sum of queue lengths when the system evolution is considered over n slots starting with state $\mathbf{s}$ (or $\mathbf{S}(0) = \mathbf{s}$) (i.e. $\mathbb{E}\left[\sum_{t=0}^{n-1}\sum_{i=1}^{N} Q_i(t)|\mathbf{S}(0) = \mathbf{s}\right]$). From [9, eq 6.6.7] for large enough n we have that the γ which attains the minimum in the above value iteration equation (4) for every $\mathbf{s}$ is an *approximately-optimal* stationary policy $\gamma(\mathbf{s})$ for the average cost problem. Furthermore, again from [9, eq. 6.3.6] we have that for large n,

$$V_n(\mathbf{s}) = h(\mathbf{s}) + ng^*. \tag{5}$$

The motivating idea behind the definition of the 1-lookahead policy is that a *good* policy can be obtained from the ACOE (3)

[3] We note that one of the sufficient conditions for the existence of the ACOE is that there exists a policy under which (2) is finite. Since the VFMW policy has a finite average queue length if the arrival rate vector is within Λ, the above sufficient condition is satisfied. Other sufficient conditions (which deal with irreducibility of the queue length Markov chain) for the existence of the ACOE can be shown to hold under appropriate assumptions on the distribution of the arrival random variables. Since this informal discussion is to motivate the form of the 1-lookahead policy, these details are not included here.

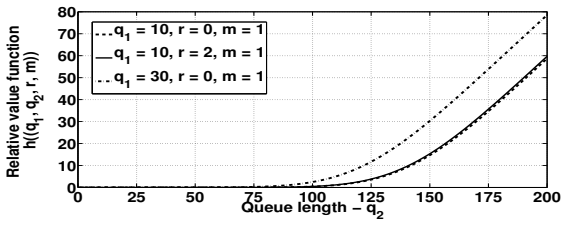

Fig. 2. The relative value function $h(\mathbf{s})$, where $\mathbf{s} = (q_1, q_2, r, m)$ plotted as a function of q_2 for different q_1, r, and m. The relative value function has been obtained from value iteration carried out for a system with buffer size for both queues truncated to 1000, channel connectivity parameters $\mu_1 = \mu_2 = 0.5$ and arrival rates $\lambda_1 = 0.15$ and $\lambda_2 = 0.2$. We observe that $h(\mathbf{s})$ need to be approximated by a non-linear function of q_2.

or the value iteration (4) if a *good* approximation can be found for $h(\mathbf{s})$ or $V_n(\mathbf{s})$. The approximations for $h(\mathbf{s})$ or $V_n(\mathbf{s})$ can be substituted into the RHS of the ACOE (3) and (4) and the minimizing γ can be proposed as a candidate policy. Suppose an approximation to $h(\mathbf{s})$ is $\tilde{h}(\mathbf{s})$. We note that except for a constant $\tilde{h}(\mathbf{s})$ is also an approximation for $V_n(\mathbf{s})$ from (5).

We first use a fluid model [6, Chapter 10] of the queueing system in order to motivate that $\tilde{h}(\mathbf{s}) = \sum_i q_i^2$ is a reasonable approximation to the function $h(\mathbf{s})$. Under the optimal policy let $\overline{\mu}_i$ be the time average of the service opportunity which is given to queue i. The fluid model of the queueing system models each queue length as a deterministic function $q_i(t)$ which evolves according to the differential equation

$$\frac{dq_i(t)}{dt} = -(\overline{\mu}_i - \lambda_i).$$

We assume that the initial state $q_i(0)$ of queue i is q_i (the component of our state $\mathbf{s}$). We then obtain that $q_i(t) = \max(q_i - (\overline{\mu}_i - \lambda_i)t, 0)$. The area under the $q_i(t)$ function or the cumulative queue length for queue i is proportional to q_i^2. The total cumulative queue length, which is then proportional $\sum_i q_i^2$, is the value function for the fluid model. From [6, Theorem 10.0.3] we have that the value function for the fluid model is the same as the relative value function $h(\mathbf{s})$, where the queue length components of $\mathbf{s}$ are q_1 and q_2 used in the fluid model, in an asymptotic regime where $q_1, q_2 \to \infty$. So we approximate the relative value function $h(\mathbf{s})$ with $\sum_i q_i^2$.

The second motivation for the use of $\sum_i q_i^2$ as an approximation comes from numerically solving the ACOE using value iteration. In Fig. 2, we plot $h(\mathbf{s})$ as a function of q_2 for different q_1, r, and m. where $h(\mathbf{s})$ has been obtained using value iteration for 1000 iterations and for queue length state space truncated to 1000. The other parameters used are indicated in Fig. 2. We find that a second order polynomial (i.e., $c_1 q_2^2 + c_2 q_2 + c_3$) is a *good* choice for fitting the observed function. When higher degree polynomials are used, the coefficients of higher degree terms are seen to be close to zero. We note that we exclude linear terms as well as cross terms (i.e., $q_1 q_2$) in our approximation for the sake of analytical simplicity.

We now use the sum of squares approximation to motivate the form of the 1-lookahead policy. From (4), we have that a policy that achieves the minimum in

$$\min_{\gamma \in \{0,1\}} \left\{ q_1 + q_2 + \mathbb{E}\left[V_n(\mathbf{S}^+) | \mathbf{s} \right] \right\},$$

for large enough n is approximately optimal for the average queue length minimization problem. Suppose we use the sum of squares approximation $\sum_i q_i^2$ for $V_n(\mathbf{s})$ (motivated by (5)). Then, we are considering the policy that chooses γ to achieve the minimum as follows

$$\min_{\gamma \in \{0,1\}} \left\{ q_1 + q_2 + \mathbb{E}\left[Q_1(1)^2 + Q_2(1)^2 | \mathbf{s} \right] \right\}.$$

Here $Q_1(1)$ and $Q_2(1)$ are the queue lengths that the MDP evolves to under an action γ according to the MDP evolution described above starting with $Q_1(0) = q_1$ and $Q_2(0) = q_2$.

Let us consider the case where $\mathbf{s}$ is such that queue 1 is currently being served (a similar discussion holds for the case where queue 2 is being served). For $\gamma = 0$ we have that

$$\mathbb{E}\left[Q_1(1)^2 + Q_2(1)^2 | \mathbf{s} \right] = \mathbb{E}\left[(q_1 - C_1 + A_1)^2 + (q_2 + A_2)^2 \right]$$

and for $\gamma = 1$ we have that

$$\mathbb{E}\left[Q_1(1)^2 + Q_2(1)^2 | \mathbf{s} \right] = \mathbb{E}\left[(q_1 + A_1)^2 + (q_2 + A_2)^2 \right],$$

since we have $T_r \geq 1$. We see that $\gamma = 1$ will never be chosen; this happens of course because our approximation does not capture the possible service that can happen for queue 2 after T_r slots. The approximation does not capture the possible service since the approximation, although simple, does not include other state variables especially r. In order to capture this potential service after T_r slots we proceed by writing $V_{n+1}(\mathbf{s})$ as:

$$V_{n+1}(\mathbf{s}) = \min_{\gamma \in \{0,1\}} \left\{ \mathbb{E}\left[\sum_{\tau=0}^{T_r+m-1} (Q_1(\tau) + Q_2(\tau)) | \mathbf{s} \right] + \mathbb{E}\left[V_{n-(T_r+m-1)}(\mathbf{S}^{+(T_r+m)}) | \mathbf{s} \right] \right\}.$$

Here $Q_1(0) = q_1$ and $Q_2(0) = q_2$ where q_1 and q_2 are the queue length components of $\mathbf{s}$. Also $Q_i(\tau)$ is the queue length that the MDP evolves to in the τ^{th} slot after taking action γ in the first slot and then the optimal actions in all slots. Then, specially note that the expectation $\mathbb{E}\left[\sum_{\tau=0}^{T_r+m-1} (Q_1(\tau) + Q_2(\tau)) | \mathbf{s} \right]$ is computed and the state $\mathbf{S}^{+(T_r+m)}$ is what the MDP evolves to in $T_r + m$ slots under the use of γ in the first slot and the optimal policy after that. We note that the above alternate expression for $V_{n+1}(\mathbf{s})$ can be written for any $n > T_r + m$, but we are interested in sufficiently large values of n. We now explicitly indicate that the optimal actions are taken for the first $T_r + m - 1$ slots by using a minimization over action variables $\gamma_1, \ldots, \gamma_{T_r+m-1}$. We have that $V_{n+1}(\mathbf{s})$ is

$$\min_{\gamma, \gamma_1, \ldots, \gamma_{T_r+m-1}} \left\{ \mathbb{E}\left[\sum_{\tau=0}^{T_r+m-1} (Q_1(\tau) + Q_2(\tau)) | \mathbf{s} \right] + \mathbb{E}\left[V_{n-(T_r+m-1)}(\mathbf{S}^{+(T_r+m)}) | \mathbf{s} \right] \right\}. \quad (6)$$

We note that since n is large the sum of squares approximation can also be applied to $V_{n-(T_r+m-1)}(\mathbf{s})$.

We now consider the case where $m = 1$. For $m = 1$ we note that

$$V_{n+1}(\mathbf{s}) = \min_{\gamma,\gamma_1,\ldots,\gamma_{T_r}} \left\{ \mathbb{E}\Big[\sum_{\tau=0}^{T_r} (Q_1(\tau) + Q_2(\tau)) | \mathbf{s} \Big] + \mathbb{E}\left[V_{n-T_r}(\mathbf{S}^{+(T_r+1)}) | \mathbf{s} \right] \right\}. \quad (7)$$

Assuming that $\mathbf{s}$ is such that queue 1 is being served, the possible action sequences can be divided into three:

A1: $\gamma = \gamma_1 = \cdots = \gamma_{T_r} = 0$ or stay with queue 1 for T_r slots,

A2: $\gamma = 1, \gamma_1 = \cdots = \gamma_{T_r} = 0$ or switch to queue 2 in the first slot and then stay with queue 2 for the rest of T_r slots,

A3: Any other action sequence.

In the following, in order to differentiate between queue length evolution under different action sequences, we use $Q_i^j(t)$ to denote the queue length for queue i under action sequence Aj (e.g. $Q_1^3(t)$ for queue length of queue 1 under action sequence $A3$). For the action sequence A1, we have for all $\tau \in \{0, \ldots, T_r + 1\}$

$$\begin{aligned} Q_1^1(\tau) &= q_1 - \sum_{l=0}^{\tau-1} C_1(l) + \sum_{l=0}^{\tau-1} A_1(l), \text{ and} \\ Q_2^1(\tau) &= q_2 + \sum_{l=0}^{\tau-1} A_2(l). \end{aligned}$$

For the action sequence A2, we have for all $\tau \in \{0, \ldots, T_r + 1\}$

$$Q_1^2(\tau) = q_1 + \sum_{l=0}^{\tau-1} A_1(l),$$

and for all $\tau \in \{0, \ldots, T_r\}$,

$$Q_2^2(\tau) = q_2 + \sum_{l=0}^{\tau-1} A_2(l).$$

We note that there is no service under A2 for the first T_r slots, but at the $(T_r + 1)^{th}$ we do have a service, i.e,

$$Q_2^2(T_r + 1) = q_2 - C_2(T_r) + \sum_{l=0}^{T_r} A_2(l).$$

We note that the action sequence corresponding to A3 above is of the form $(\gamma = 0, \gamma_1 = 0, \ldots, \gamma_{\tau_s - 1} = 1, \ldots)$. That is, we switch to queue 2 from queue 1 in the τ_s^{th} slot where $\tau_s > 1$. We note that for $m = 1$, after τ_s there cannot be any more service in the system in the first T_r slots since there is a reconfiguration delay of T_r slots. Then, we have that for A3 and any $\tau \in \{0, 1, \ldots, T_r + 1\}$

$$\begin{aligned} Q_1^3(\tau) &= q_1 - \sum_{l=0}^{\min(\tau,\tau_s)-1} C_1(l) + \sum_{l=0}^{\tau-1} A_1(l), \text{ and} \\ Q_2^3(\tau) &= q_2 + \sum_{l=0}^{\tau-1} A_2(l). \end{aligned}$$

We note that for the same sample path of arrivals $A_1(l), A_2(l)$ and channel connectivity $C_1(l)$ and $C_2(l)$ and for any $\tau \in \{0, \ldots, T_r + 1\}$, $Q_1^1(\tau) \leq Q_1^3(\tau)$ and $Q_2^1(\tau) = Q_2^3(\tau)$. Also under the assumption of the sum of squares form for $V_{n-T_r}(.)$ we have that under A1

$$\mathbb{E}\left[V_{n-T_r}(\mathbf{S}^{+(T_r+1)}) | \mathbf{s} \right] = \mathbb{E}\left[\sum_i (Q_i^1(T_r + 1))^2 \big| \mathbf{s} \right],$$

which is less than the following corresponding value

$$\mathbb{E}\left[V_{n-T_r}(\mathbf{S}^{+(T_r+1)}) | \mathbf{s} \right] = \mathbb{E}\left[\sum_i (Q_i^3(T_r + 1))^2 \big| \mathbf{s} \right],$$

under A3. Therefore, we conclude that for $m = 1$ and a sum of squares approximation for $V_{n-T_r}(\mathbf{s})$, the optimal sequence of actions is either A1 or A2 in the above list but not any action sequence in A3. Thus, for $m = 1$ with the sum of squares form for $V_{n-T_r}(.)$ we have that the optimal choice is between A1 and A2; which incidentally is a choice between staying with the current queue or switching to the other queue. This choice forms the basis for the definition of the 1-lookahead (note that $m = 1$) policy.

From (7) and the fact that the optimal sequence is either A1 or A2 we have the following decision rule. We choose $\gamma = 0$ if

$$\begin{aligned} &\mathbb{E}\left[\sum_i \sum_{\tau=0}^{T_r} Q_i^1(\tau) + \sum_i Q_i^1(T_r + 1))^2 \big| \mathbf{s} \right] < \\ &\mathbb{E}\left[\sum_i \sum_{\tau=0}^{T_r} Q_i^2(\tau) + \sum_i Q_i^2(T_r + 1))^2 \big| \mathbf{s} \right] \end{aligned} \quad (8)$$

and $\gamma = 1$ otherwise. We have that

$$\begin{aligned} &\mathbb{E}\left[\sum_{\tau=0}^{T_r} (Q_1^1(\tau) + Q_2^1(\tau)) | \mathbf{s} \right] = \\ &\mathbb{E}\left[\sum_{\tau=0}^{T_r} \left[\sum_i (q_i + \sum_{l=0}^{\tau-1} A_i(l)) - \sum_{l=0}^{\tau-1} C_1(l) \right] \Big| \mathbf{s} \right], \end{aligned} \quad (9)$$

and $\mathbb{E}\left[\sum_i (Q_i^1(T_r + 1))^2 | \mathbf{s} \right]$ is

$$\mathbb{E}\left[(q_1 + \sum_{\tau=0}^{T_r} (A_1(\tau) - C_1(\tau)))^2 + (q_2 + \sum_{\tau=0}^{T_r} A_2(\tau))^2 | \mathbf{s} \right]$$

which can be simplified as

$$\begin{aligned} &= \mathbb{E}\left[\left(\sum_{\tau=0}^{T_r} C_1(\tau) \right)^2 | C_1(0) \right] + \mathbb{E}\left[\left(\sum_{\tau=0}^{T_r} A_1(\tau) \right)^2 \right] - \\ &2q_1 \mathbb{E}\left[\sum_{\tau=0}^{T_r} C_1(\tau)) | C_1(0) \right] + 2q_1 \mathbb{E}\left[\sum_{\tau=0}^{T_r} A_1(\tau) \right] \\ &-2\mathbb{E}\left[\sum_{\tau=0}^{T_r} C_1(\tau)) | C_1(0) \right] \mathbb{E}\left[\sum_{\tau=0}^{T_r} A_1(\tau) \right] + q_1^2 + q_2^2 + \\ &\mathbb{E}\left[\left(\sum_{\tau=0}^{T_r} A_2(\tau) \right)^2 \right] + 2q_2 \mathbb{E}\left[\sum_{\tau=0}^{T_r} A_2(\tau) \right]. \end{aligned} \quad (10)$$

Similarly we have that

$$\mathbb{E}\left[\sum_{\tau=0}^{T_r}(Q_1^2(\tau)+Q_2^2(\tau))|\mathbf{s}\right] = \quad (11)$$
$$\mathbb{E}\left[\sum_{\tau=0}^{T_r}(q_1+\sum_{l=0}^{\tau-1}A_1(l)+q_2+\sum_{l=0}^{\tau-1}A_2(l))|\mathbf{s}\right],$$

and

$$\mathbb{E}\left[(Q_1^2(T_r+1))^2+(Q_2^2(T_r+1))^2|\mathbf{s}\right] =$$
$$\mathbb{E}\left[(q_1+\sum_{\tau=0}^{T_r}A_1(\tau))^2+(q_2-C_2(T_r)+\sum_{\tau=0}^{T_r}A_2(\tau))^2|\mathbf{s}\right]$$
$$= q_1^2+\mathbb{E}\left[\left(\sum_{\tau=0}^{T_r}A_1(\tau)\right)^2\right]+2q_1\mathbb{E}\left[\sum_{\tau=0}^{T_r}A_1(\tau)\right]+q_2^2+$$
$$\mathbb{E}\left[\left(\sum_{\tau=0}^{T_r}A_2(\tau)\right)^2\right]+2q_2\mathbb{E}\left[\sum_{\tau=0}^{T_r}A_2(\tau)\right]$$
$$+\mathbb{E}\left[(C_2(T_r))^2|C_2(0)\right]-2q_2\mathbb{E}\left[C_2(T_r)|C_2(0)\right]-$$
$$2\mathbb{E}\left[C_2(T_r)|C_2(0)\right]\mathbb{E}\left[\sum_{\tau=0}^{T_r}A_2(\tau)\right]. \quad (12)$$

We note that the comparison in (8) is equivalent to the comparison

$$(9)+(10)<(11)+(12).$$

Several terms are common in (9) + (10) and (11) + (12). We cancel out those common terms and keep only those terms which have a queue length term appearing in it. For large values of queue lengths, these terms would dominate other constant terms. Furthermore, such a choice would result in the decision rule for 1-lookahead to have a simple biased max-weight form. Then, we get a heuristic decision rule of choosing $\gamma=0$ if

$$-2q_1\mathbb{E}\left[\sum_{\tau=0}^{T_r}C_1(\tau))|C_1(0)\right] < -2q_2\mathbb{E}\left[C_2(T_r)|C_2(0)\right], \text{ or,}$$
$$q_1\mathbb{E}\left[\sum_{\tau=0}^{T_r}C_1(\tau))|C_1(0)\right] > q_2\mathbb{E}\left[C_2(T_r)|C_2(0)\right],$$

and $\gamma=1$ otherwise. We note that this is the motivation behind the definition of the biased max-weight 1-lookahead policy.

We now consider the case where $m>1$. Using a similar sequence of steps as for $m=1$ it is possible to derive a decision rule comparing A1 with A2. This comparison between A1 and A2 leads to the rule of choosing $\gamma=0$ if

$$q_1\mathbb{E}\left[\sum_{\tau=0}^{T_r+m-1}C_1(\tau))|C_1(0)\right] > q_2\mathbb{E}\left[\sum_{\tau=T_r}^{T_r+m-1}C_2(\tau)|C_2(0)\right].$$

However, for $m>1$ we note that it is not possible to conclude that the optimal sequence of actions is either A1 or A2 but not any in A3. We illustrate this via an example here. Consider (6) for $T_r=2$ and $m=3$. Assuming that we are serving queue 1, a possible sequence of decisions is $(\gamma=0,\gamma_1=1,\gamma_2=0,\gamma_3=1,\gamma_4=0)$, i.e., we stay with queue 1, then switch to queue 2, wait for 2 slots, switch back to queue 1 and wait for 2 slots. Such a sequence of decisions might lead to service for both queue 1 and queue 2. We note that the simple decision rule that we have proposed in this case just compares A1 and A2 and not any policy in A3 (such as the one above). If we include other comparisons to action sequences of type A3, the simplicity of the definition of 1-lookahead policies does not carry over. Since, 1-lookahead policies are throughput optimal and have a simple specification, we restrict our attention to the case of $m=1$. We note that the correction to the weight or bias, using $\frac{T_r}{F(\mathbf{Q}(t))}$, is motivated by the proof of throughput optimality.

V. Conclusions and Future Work

In this paper, we considered queueing system models with reconfiguration delay. We proposed a new 1-lookahead policy using an approximate solution to a Markov decision process formulation. The biased max-weight form of the 1-lookahead policy is analytically well motivated. Furthermore, since the form of 1-lookahead is similar to that of QBMW, the bias term which appears in QBMW can also be explained using this analytical development. We also prove that 1-lookahead is throughput optimal. We note that the Markov decision process formulation can be extended to the case of correlated channel connectivity, by including channel states in the MDP's state. It is possible to again define a 1-lookahead policy for the correlated channel case (see [5]); proving the throughput optimality of such a policy would be part of our future work.

References

[1] Güner D Celik, Sem C Borst, Philip A Whiting, and Eytan Modiano. Dynamic scheduling with reconfiguration delays. *Queueing Systems*, 83(1-2):87–129, 2016.

[2] Güner D Celik, Long B Le, and Eytan Modiano. Dynamic server allocation over time-varying channels with switchover delay. *IEEE Transactions on Information Theory*, 58(9):5856–5877, 2012.

[3] Güner D Celik and Eytan Modiano. Scheduling in networks with time-varying channels and reconfiguration delay. *IEEE/ACM Transactions on Networking*, 23(1):99–113, 2015.

[4] Ping-Chun Hsieh, I Hou, and Xi Liu. Delay-optimal scheduling for queueing systems with switching overhead. *arXiv preprint arXiv:1701.03831*, 2017.

[5] Amit Kumar and Vineeth B. S. Scheduling policies for wireless downlink with correlated random connectivity and multislot reconfiguration delay. *Accepted to IEEE Communications Letters*, 2017.

[6] S. P. Meyn. *Control Techniques for Complex Networks*. Cambridge University Press, 2007.

[7] Michael J Neely. Stochastic network optimization with application to communication and queueing systems. *Synthesis Lectures on Communication Networks*, 3(1):1–211, 2010.

[8] Linn I Sennott. *Stochastic dynamic programming and the control of queueing systems*, volume 504. John Wiley & Sons, 2009.

[9] C. H. Tijms. *A first course in stochastic models*. Wiley, 2003.

Appendix A
Proof of Theorem III.1

We prove the stability of the modified 1-lookahead policy in this section. Suppose t_k is the slot at which the k^{th} switch happens. Let $T_k=t_{k+1}-t_k$ be the duration of the k^{th} frame. Let us assume that in the k^{th} frame the i^{th} queue is being

served. Then we note that at t_k, $\mu_i Q_i(t_k) > \mu_j Q_j(t_k)$ for every j. Furthermore, we have that there exists a j such that

$$\left(1+\frac{T_r}{F(\mathbf{Q}(t_k+T_k))}\right)Q_i(t_k+T_k)\mu_i < Q_j(t_k+T_k)\mu_j.$$

We first prove the following lemma which states that for large enough queue lengths, the duration of the frame T_k is large. The proof is similar to Lemma 3 in [4].

Lemma A.1. *The length T_k of the k^{th} frame for every k is such that*

$$T_k^{1+\alpha} > \frac{T_r \sum_l Q_l(t_k)\mu_l}{N\left(1+\left(A_{max}N+\sum_l Q_l(t_k)\mu_l\right)^{\alpha}\right)(1+T_r+A_{max})},$$

where $\alpha \in (0,1)$ is as in the definition of $F(\mathbf{Q}(t))$.

Proof. From the definition of T_k (or t_{k+1}) we have that there exists a j such that

$$\left(1+\frac{T_r}{F(\mathbf{Q}(t_k+T_k))}\right)Q_i(t_k+T_k)\mu_i < Q_j(t_k+T_k)\mu_j.$$

We note that

$$\begin{aligned} Q_i(t_k+T_k) &\geq Q_i(t_k)-T_k, \\ Q_j(t_k+T_k) &\leq Q_j(t_k)+A_{max}T_k, \end{aligned}$$

where we have used the fact that in T_k slots the maximum amount of service for queue i is T_k and the maximum number of arrivals for queue j is $A_{max}T_k$. We also note that the first lower bound could be negative. Substituting these bounds we obtain that

$$\begin{aligned} \left(1+\frac{T_r}{F(\mathbf{Q}(t_k+T_k))}\right)(Q_i(t_k)-T_k)\mu_i < \\ (Q_j(t_k)+A_{max}T_k)\mu_j, \\ \left(1+\frac{T_r}{F(\mathbf{Q}(t_k+T_k))}\right)Q_i(t_k)\mu_i - \mu_j Q_j(t_k) < \\ T_k((1+T_r)\mu_i + A_{max}\mu_j), \end{aligned}$$

where we have used $F(\mathbf{Q}(t_k+T_k)) \geq 1$. Since μ_i and $\mu_j \leq 1$ and $\mu_i Q_i(t_k) \geq \mu_j Q_j(t_k)$ we have that

$$T_k > \frac{T_r Q_i(t_k)\mu_i}{F(\mathbf{Q}(t_k+T_k))(1+T_r+A_{max})}. \tag{13}$$

We note that

$$\begin{aligned} F(\mathbf{Q}(t_k+T_k)) &\leq 1+\left(\sum_l (Q_l(t_k)+A_{max}T_k)\mu_l\right)^{\alpha}, \\ &\leq 1+\left(A_{max}T_kN+\sum_l Q_l(t_k)\mu_l\right)^{\alpha}, \end{aligned}$$

where we have used that there are N queues and $\mu_l \leq 1, \forall l$. Then we have that

$$\begin{aligned} F(\mathbf{Q}(t_k+T_k)) &\leq 1+T_k^{\alpha}\left(A_{max}N+\sum_l Q_l(t_k)\mu_l\right)^{\alpha}, \\ &\leq T_k^{\alpha}\left(1+\left(A_{max}N+\sum_l Q_l(t_k)\mu_l\right)^{\alpha}\right). \end{aligned}$$

where we have used $T_k \geq 1$. Using this upper bound on $F(.)$ in (13) we obtain that

$$T_k^{1+\alpha} > \frac{T_r Q_i(t_k)\mu_i}{\left(1+\left(A_{max}N+\sum_l Q_l(t_k)\mu_l\right)^{\alpha}\right)(1+T_r+A_{max})}.$$

We also note that $Q_i(t_k)\mu_i \geq Q_l(t_k)\mu_l$ and hence $Q_i(t_k)\mu_i \geq \frac{\sum_l Q_l(t_k)\mu_l}{N}$. Hence,

$$T_k^{1+\alpha} > \frac{T_r \sum_l Q_l(t_k)\mu_l}{N\left(1+\left(A_{max}N+\sum_l Q_l(t_k)\mu_l\right)^{\alpha}\right)(1+T_r+A_{max})}.$$

□

An important corollary of this lemma is stated below.

Corollary A.2. *Given a $\Delta \in \mathbb{Z}_+$ there is a finite set $\mathcal{Q}_\Delta$ of queue length vectors such that $T_k \geq T_r+\Delta$ for $\mathbf{Q}(t_k) \in \mathcal{Q}_\Delta^c$.*

Proof. From Lemma A.1 we have that $T_k >$

$$\left(\frac{T_r \sum_l Q_l(t_k)\mu_l}{N\left(1+\left(A_{max}N+\sum_l Q_l(t_k)\mu_l\right)^{\alpha}\right)(1+T_r+A_{max})}\right)^{\frac{1}{1+\alpha}}.$$

We note that the lower bound on T_k is an increasing function of $\sum_l Q_l(t_k)\mu_l$. So for a given Δ, there is a finite δ such that $T_k \geq T_r+\Delta$ for $\sum_l Q_l(t_k)\mu_l > \delta$. There is a finite set of queue length vectors $\mathcal{Q}^{\Delta}$ such that $\sum_l Q_l(t_k)\mu_l \leq \delta$ for which the above lower bound on T_k does not guarantee that $T_k > T_r+\Delta$. □

We now prove the stability of the 1-lookahead policy using Lyapunov drift arguments. The Lyapunov function is chosen to be $L(t) = \sum_l Q_l(t)^2$. From Corollary A.2, we note that at t_k for all queue length vectors except for a finite set $\mathcal{Q}_\Delta$ we have that $T_k \geq T_r+\Delta$. We first consider the case of the complement set of $\mathcal{Q}_\Delta$. So $T_k \geq T_r+\Delta$ in the discussion below.

In the stability proof, we will consider the Lyapunov drift of the system over a frame, first between t_k and $t_k+T_r+\Delta$ and then for every slot in $\{t_k+T_r+\Delta+1, t_k+T_k\}$. We note that in [4] the drift is considered over t_k and $t_k+\tilde{T}_k$ instead of between t_k and $t_k+T_r+\Delta$. In [4], $\tilde{T}_k$ is defined as $\min(T_k, F(\mathbf{Q}(t_k)))$. It is not clear why $\tilde{T}_k$ would be greater than T_r; this property is essential to the drift arguments that follow in [4]. In contrast, we consider $\tilde{T}_k$ to be a constant $T_r+\Delta$ but only for those t_k for which the queue length vector lies in $\mathcal{Q}_\Delta^c$. Then, we show that whatever Lyapunov drift that we obtain for the case where $\mathbf{Q}(t_k) \in \mathcal{Q}_\Delta^c$ can be applied to the case where $\mathbf{Q}(t_k) \in \mathcal{Q}_\Delta$ but with a different constant factor in the drift expression.

The Lyapunov drift between t_k and $t_k+T_r+\Delta$ is

$$\sum_l Q_l(t_k+T_r+\Delta)^2 - \sum_l Q_l(t_k)^2.$$

We note that

$$\begin{aligned} Q_l(t_k+T_r+\Delta) \leq \max\left(Q_l(t_k)-\sum_{t=0}^{T_r+\Delta-1} I_l(t_k+t), 0\right)+ \\ \sum_{t=0}^{T_r+\Delta-1} A_l(t_k+t). \end{aligned}$$

With Γ_l defined as $\max\left(Q_l(t_k) - \sum_{t=0}^{T_r+\Delta-1} I_l(t_k+t), 0\right) - \left(Q_l(t_k) - \sum_{t=0}^{T_r+\Delta-1} I_l(t_k+t)\right)$, we have that

$$Q_l(t_k+T_r+\Delta) \leq Q_l(t_k) - \sum_{t=0}^{T_r+\Delta-1} I_l(t_k+t) + \Gamma_l + \sum_{t=0}^{T_r+\Delta-1} A_l(t_k+t).$$

Then (14) can be bounded above as

$$\sum_l \left(Q_l(t_k) - \sum_{t=0}^{T_r+\Delta-1} I_l(t_k+t) + \Gamma_l + \sum_{t=0}^{T_r+\Delta-1} A_l(t_k+t) \right)^2 - Q_l(t_k)^2,$$
$$\leq B_0^\Delta + 2\sum_l Q_l \left(\sum_{t=0}^{T_r+\Delta-1} (A_l(t_k+t) - I_l(t_k+t)) \right).$$

Here B_0^Δ is a constant which is obtained by bounding $I_l(t_k+t)$ above by 1, $A_l(t_k)$ above by A_{max}, and the fact that Γ_l is at most $(T_r+\Delta)$ and is non-zero only if $Q_l \leq T_r+\Delta$. We note that B_0^Δ is a function of Δ. The expected Lyapunov drift (which is conditioned on $\mathbf{Q}(t_k)$) is then

$$\leq \quad B_0^\Delta + 2\sum_l Q_l \left((T_r+\Delta)\lambda_l - \Delta \mathbb{I}_{\{l=i\}}\mu_i \right),$$

where $\mathbb{I}_{\{l=i\}}$ is 1 if $l=i$ and 0 otherwise, since in the k^{th} frame there is service only for the i^{th} queue. Furthermore, in the first T_r slots after t_k we do not have service since the system is in reconfiguration. We now write the expected Lyapunov using vector notation as being

$$\leq \quad B_0^\Delta + 2\left(\boldsymbol{\lambda} - \frac{\Delta}{T_r+\Delta}\mathbf{I}^{(i)}\mu_i\right)^T \mathbf{Q}(t_k), \qquad (14)$$

where all vectors are column vectors and $\mathbf{I}^{(i)}$ is a vector of zeros except with 1 at the i^{th} position. Suppose we consider a $\boldsymbol{\lambda}$ in the stability region, then we have that there exists a set of β_j ($\sum_j \beta_j \leq 1$) such that

$$\boldsymbol{\lambda} \leq \sum_j \beta_j \mathbf{I}^{(j)}\mu_j,$$

and we can write $\boldsymbol{\lambda} = (1-\epsilon)\sum_j \beta_j \mathbf{I}^{(j)}\mu_j$ for some $\epsilon > 0$. Furthermore we have that $\mu_j(I^{(j)})^T\mathbf{Q}(t_k) \leq \mu_i(I^{(i)})^T\mathbf{Q}(t_k)$ since we have switched to i at t_k. Then we have that (14) is

$$\leq \quad B_0^\Delta + 2\left(\left(1-\epsilon-\frac{\Delta}{T_r+\Delta}\right)\mathbf{I}^{(i)}\mu_i\right)^T \mathbf{Q}(t_k),$$

For any $\epsilon > 0$ we choose Δ such that $1-\epsilon-\frac{\Delta}{T_r+\Delta} = -\epsilon_1$ for some $\epsilon_1 > 0$. Then we have that the expected Lyapunov drift between $t_k+T_r+\Delta$ and t_k is

$$\leq \quad B_0^\Delta - 2\epsilon_1\mu_i \left(\mathbf{I}^{(i)}\right)^T \mathbf{Q}(t_k).$$

Now since $\mu_j(I^{(j)})^T\mathbf{Q}(t_k) \leq \mu_i(I^{(i)})^T\mathbf{Q}(t_k)$ we have that

$$\leq \quad B_0^\Delta - 2\frac{\epsilon_1}{N}\sum_l \mu_l Q_l(t_k).$$

We note that $Q_l(t_k+t) \leq Q_l(t_k) + A_{max}t$ or

$$\sum_{t=0}^{T_r+\Delta-1} \mu_l Q_l(t_k+t) - \frac{\mu_l A_{max}(T_r+\Delta)(T_r+\Delta+1)}{2} \leq (T_r+\Delta)\mu_l Q_l(t_k).$$

Then the expected Lyapunov drift between t_k and $t_k+T_r+\Delta$ is

$$\leq \quad B_1^\Delta - 2\frac{\epsilon_1}{(T_r+\Delta)N}\sum_l \mu_l \sum_{t=0}^{T_r+\Delta-1} Q_l(t_k+t),$$

where B_1^Δ is a constant dependent on Δ.

We now consider the slot to slot drift of the Lyapunov functions for slots $t \in \{t_k+T_r+\Delta+1, t_k+T_k\}$. We note that we are still restricting to those t_k at which $\mathbf{Q}(t_k) \in \mathcal{Q}_\Delta^c$. The Lyapunov drift is

$$\sum_l Q_l(t+1)^2 - \sum_l Q_l(t)^2$$

As in the previous case we have that

$$Q_l(t+1) = Q_l(t) - I_l(t) + A_l(t) + \Gamma_l,$$

and it can be shown that the Lyapunov drift is

$$\leq \quad B_2 + 2\sum_l Q_l(t)\left(A_l(t) - I_l(t)\right).$$

The expected Lyapunov drift conditioned on $\mathbf{Q}(t)$ is

$$\leq \quad B_2 + 2\sum_l Q_l(t)\left(\lambda_l - \mathbb{I}_{\{l=i\}}\mu_i\right).$$

Here we note that there is only service to queue i in frame k and we are considering slots after the reconfiguration delay. In vector notation we have

$$\leq \quad B_2 + 2\left(\boldsymbol{\lambda} - \mathbf{I}^{(i)}\mu_i\right)^T \mathbf{Q}(t). \qquad (15)$$

Since $\boldsymbol{\lambda}$ is in the stability region we again have that (15) is

$$\leq \quad B_2 + 2\left((1-\epsilon)\sum_j \beta_j \mathbf{I}^{(j)}\mu_j - \mathbf{I}^{(i)}\mu_i\right)^T \mathbf{Q}(t).$$

We note that at t, $\mu_j(\mathbf{I}^{(j)})^T\mathbf{Q}(t) \leq \left(1+\frac{T_r}{F(\mathbf{Q}(t))}\right)\mu_i(\mathbf{I}^{(i)})^T\mathbf{Q}(t)$. Substituting we get the conditional drift as

$$\leq \quad B_2 + 2\left((1-\epsilon)\left(1+\frac{T_r}{F(\mathbf{Q}(t))}\right) - 1\right)\mu_i(\mathbf{I}^{(i)})^T\mathbf{Q}(t),$$
$$= \quad B_2 + 2\left(-\epsilon + (1-\epsilon)\left(\frac{T_r}{F(\mathbf{Q}(t))}\right)\right)\mu_i(\mathbf{I}^{(i)})^T\mathbf{Q}(t)$$
$$= \quad B_2 - 2\epsilon\mu_i(\mathbf{I}^{(i)})^T\mathbf{Q}(t) + 2(1-\epsilon)\left(\frac{T_r}{F(\mathbf{Q}(t))}\right)\mu_i(\mathbf{I}^{(i)})^T\mathbf{Q}(t).$$

We note that $\mu_i(\mathbf{I}^{(i)})^T\mathbf{Q}(t) \leq \sum_l \mu_l(\mathbf{I}^{(l)})^T\mathbf{Q}(t)$. Also $\mu_i(\mathbf{I}^{(i)})^T\mathbf{Q}(t) \geq \frac{1}{N(1+T_r)}\sum_l \mu_l(\mathbf{I}^{(l)})^T\mathbf{Q}(t)$. There we have the conditional drift as

$$\leq \quad B_2 - \frac{2\epsilon}{N(1+T_r)}\sum_l \mu_l(\mathbf{I}^{(l)})^T\mathbf{Q}(t) + 2(1-\epsilon)\left(\frac{T_r}{F(\mathbf{Q}(t))}\right)\sum_l \mu_l(\mathbf{I}^{(l)})^T\mathbf{Q}(t).$$

We note that there would exist a finite set $\mathcal{Q}_\alpha$ in the complement of which the above drift would be negative since the dominant term in the above bound is the second term. By defining a new constant B_3, we can therefore write that the expected drift is

$$\leq \quad B_3 - \frac{2\epsilon_2}{N(1+T_r)} \sum_l \mu_l (\mathbf{I}^{(l)})^T \mathbf{Q}(t).$$

We note that the above drifts were defined for t_k such that $\mathbf{Q}(t_k) \notin \mathcal{Q}_\Delta$. By redefining the constants in the drift expressions we have that for all $\mathbf{Q}(t_k)$ the above drift conditions hold. Then the rest of the proof proceeds as in [4] (from equation 127 onwards) and it can be shown that the average queue length is finite.