The Role of Information in System Stability with Partially Observable Servers

A. Asanjarani[*], Y. Nazarathy[†]

The University of Queensland, St Lucia, Brisbane, QLD, 4072.

June 28, 2021

Abstract

We consider a simple discrete time controlled queueing system, where the controller has a choice of which server to use at each time slot and server performance varies according to a Markov modulated random environment. We explore the role of information in the system stability region. At the extreme cases of information availability, that is when there is either full information or no information, stability regions and maximally stabilizing policies are trivial. But in the more realistic cases where only the environment state of the selected server is observed, only the service successes are observed or only queue length is observed, finding throughput maximizing control laws is a challenge. To handle these situations, we devise a Partially Observable Markov Decision Process (POMDP) formulation of the problem and illustrate properties of its solution. We further model the system under given decision rules, using Quasi-Birth-and-Death (QBD) structure to find a matrix analytic expression for the stability bound. We use this formulation to illustrate how the stability region grows as the number of controller belief states increases.

Our focus in this paper is on the simple case of two servers where the environment of each is modulated according to a two-state Markov chain. As simple as this case seems, there appear to be no closed form descriptions of the stability region under the various regimes considered. Our numerical approximations to the POMDP Bellman equations and the numerical solutions of the QBDs hint at a variety of structural results.

keyword: system stability, POMDP, control, queueing systems, optimal, Bellman equations, QBD, information, Markov models.

---

[*]a.asanjarani@uq.edu.au

[†]y.nazarathy@uq.edu.au

1

1 Introduction

Performance evaluation and control of queueing systems subject to randomly varying environments is an area of research that has received much attention during the past few decades (see for example, [4], [8], [23] and references there-in). This is because numerous situations arise in practice where a controller needs to decide how to best utilise resources, and these are often subject to changing conditions. Examples of such situations arise in wireless communication, supply chain logistics, health care, manufacturing and transportation. In all these situations, it is very common for service rates to vary in a not fully predictable manner. Using Markovian random environments has often been a natural modelling choice due to the tractability and general applicability of Markov models. See for example Section A.1 in [18] for a general discussion on the ubiquity of Markov models.

The bulk of the literature dealing with performance evaluation and control of these types of problems, has considered the situation where the state of the underlying random environment is observable. In such a situation, it is already a non-trivial task to carry out explicit system performance analysis (see [9] as an example). Further, finding optimal or even merely stabilizing control is typically a formidable achievement (see for example [1], [25] or the more recent [5]). But in practice, the actual environment state is often not a directly observed quantity, or is at best only partially observable. The situation is further complicated when control decisions do not only affect instantaneous rewards, but also the observation made. In classic linear-quadratic optimal control settings (e.g. Part III of [28]), the certainty equivalence principle allows to decouple state estimation based on observations and control decisions. However, in more complicated settings such as what we consider here, certainty equivalence almost certainly doesn't hold.

In this paper we augment the body of literature dealing with exploration vs. exploitation trade-offs in systems where a controller needs to choose a server (channel/resource/bandit) at any given time, and the choice influences both the immediate reward (service success) and the information obtained. A general class of such problems, denoted Reward Observing Restless Multi Armed Bandits (RORMAB), is outlined in [11], where much previous literature is surveyed. Key contributions in this area are [10] and the more recent [16]. The former finds the structure of optimal policies from first principles. The latter, generalizes the setting and utilizes the celebrated Whittle Index, [27] for such a partially observable case. Related recent results dealing with RORMAB problems are in [12] and [13]. Of further interest is the latest rigorous account on asymptotic optimality of the Whittle index, [26].

Our focus in this paper, is on a controlled queueing systems, where server environments vary and the controller (choosing servers) only observes partial information. Our aim is to explore the role of information in system stability. For this, we devise what is perhaps the simplest non-trivial model possible: a

single discrete time queue is served by either Server 1 or Server 2 where each server environment is an independent two-state Markov chain. A controller having (potentially) only partial state information, selects one of the two servers at each time instance.

The role of information is explored by considering different observation schemes. At one extreme, the controller has full information of the servers' environment states. At the other extreme, the controller is completely unaware of the servers' environment states. Obviously the stability region of the system in the latter situation is a subset of the former. Our contribution is in considering additional more realistic observation schemes. One such scheme is a situation where the controller only observes the state of the server currently chosen. This type of situation has been widely studied in some of the references mentioned above and surveyed in [11], but most of the literature dealing with this situation does not consider a queue. A more constrained scenario is one where the controller only observes the success/failure of service (from the server chosen) at every time slot. Such a partial observability situation was recently introduced in [21] in the context of stability and analysed in [19] with respect to the Whittle index. In [15], stability of a related multi-server system was analysed.

An additional observation scheme that we consider is one where the server is only aware of the queue size process. In (non-degenerate) continuous time systems, such an observation scheme is identical to the former scheme. But an artefact of our discrete time model is that such a scheme reveals less information to the controller (this is due to the fact that both an arrival and a departure may occur simultaneously, going unnoticed by the controller).

With the introduction of the five observation schemes mentioned above, this work takes first steps to analyse the effect of information on the achievable stability region. A controller of such a system makes use of a belief state implementation. We put forward (simple) explicit belief state update recursions for each of the observation schemes. These are then embedded in Bellman equations describing optimal solutions of associated Partially Observable Markov Decision Processes (POMDP). Numerical solution of the POMDPs then yields insight on structural properties and achievable stability regions. By construction, two-state Markov server environments are more predictable when the mixing times of the Markov chains increase. We quantify this use of channel-memory, through numerical and analytic results.

It is often the case that MDPs (or POMDPs) associated with queueing models, can be cast as QBDs once a class of control policies is found. See for example [20]. We follow this paradigm in the current paper and present a detailed QBD model of the system. The virtue of our QBD based model is that we are able to quantify the effect of a finite state controller on the achievable stability region whose upper bound is given by an elegant matrix analytic expression.

The remainder of this paper is structured as follows. In Section 2, we

3

introduce the system model and different observation schemes. In Section 3, we put forward recursions for belief state updates for the non-trivial observation schemes. In Section 4, we present the myopic policy and the Bellman equations for different observation schemes and present findings from a numerical investigation. In Section 5, we construct a QBD representation of the system, find the stability criterion and put forward the numerical results which are matched with the results of Bellman equations of Section 4. We conclude in Section 6.
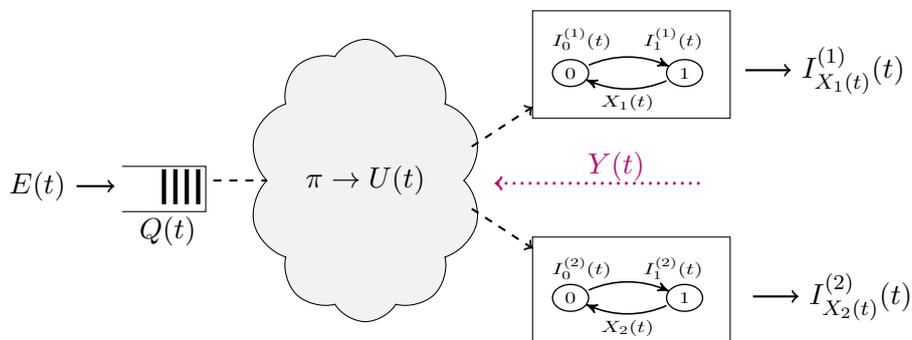
2 System Model

Figure 1: A controller operating under a decision rule, π, decides at each time step, t, if to use server $U(t) = 1$ or $U(t) = 2$ based on previously observed information, $Y(t-1), Y(t-2), \ldots$. The server environments, $X_i(t)$ are Markov modulated.

Consider a situation as depicted in Figure 1. Jobs arrive into the queue, $Q(t)$, and are potentially served by one of two servers $j = 1$ or $j = 2$, according to some control policy. The system is operating in discrete time steps $t = 0, 1, \ldots$, where in each time step, the following sequence of events occur:

1. An arrival occurs as indicated through $E(t)$: $E(t) = 1$ indicates an arrival and otherwise $E(t) = 0$.

2. The environments of the servers update from $\big(X_1(t-1), X_2(t-1)\big)$ to $\big(X_1(t), X_2(t)\big)$, autonomously. That is, the updating of the environment is not influenced by arrivals, queue length and controller choice.

3. A control decision $u = U(t)$ of which server to select is made based on observations in previous time steps, denoted by $Y(t-1), Y(t-2), \ldots$.

4

This is through a decision rule, π. We consider different observation schemes as described below.

4. The control action is executed and the queue length is updated as follows:

$$Q(t+1) = Q(t) + E(t) - I(t), \qquad \text{with} \qquad I(t) = I^{(u)}_{X_u(t)}(t).$$

Here, $I(t)$ indicates whether there was a service success or not. It is constructed from the primitive sequences,

$$\{I^{(j)}_i(t), \ t = 0, 1, 2, \dots\},$$

for servers $j = 1, 2$ and environment states $i = 0, 1$. Note that when $Q(t) = 0$, we notionally assume that $U(t) = u = 0$, indicating "no action" and in this case denote $I^{(0)}_i \equiv 0$ for $i = 0, 1$.

5. The observation of $Y(t)$ is made and is used in subsequent time steps.

The sequence of events (1)–(5) as above repeats in every time step and fully defines the evolution law of the system. We consider the following distinct observation schemes:

(I) Full observation: The controller knows the state of both servers all the time. In this case

$$Y(t) = \big(X_1(t), \ X_2(t)\big),$$

and further, the sequence of steps above is slightly modified with step (5) taking place between steps (2) and (3) and the policy at step 3 being

$$u = U(t) = \pi\big(Y(t)\big).$$

(II) State observation: The controller observes the state of the selected server at time t, but does not observe the other server at that time. Hence

$$Y(t) = \begin{cases} \big(X_1(t), \emptyset\big) & \text{if} \ \ u = 1, \\ \big(\emptyset, X_2(t)\big) & \text{if} \ \ u = 2. \end{cases}$$

(III) Output observation: The controller observes the success or failure of outputs of the server selected (but gains no information about the other server at that time). Hence

$$Y(t) = \begin{cases} \big(I^{(1)}_{X_1(t)}(t), \emptyset\big) & \text{if} \ \ u = 1, \\ \big(\emptyset, I^{(2)}_{X_2(t)}(t)\big) & \text{if} \ \ u = 2. \end{cases}$$

(IV) Queue observation: The controller only observes the queue length, $Q(t)$, and can thus utilize the differences

$$\Delta Q(t) = Q(t+1) - Q(t) = E(t) - I(t).$$

Note that since the system is operating in discrete time, there is some loss of information compared to case III : If $\Delta Q(t) = 1$ or $\Delta Q(t) = -1$, then it is clear that $I(t) = 0$ or $I(t) = 1$, respectively. But if $\Delta Q(t) = 0$, then since the controller does not observe $E(t)$, there is not a definitive indication of $I(t)$.

(V) No observation: We assume the controller does not observe anything. Nonetheless, as with the other cases, the controller knows the system parameters as described below.

We consider the simplest non-trivial probably model for the primitives. These are $E(t)$, $I_i^{(j)}(t)$ and the environment processes, $X_j(t)$, all assumed mutually independent. The arrivals, $E(t)$, are an i.i.d. sequence of Bernoulli random variables, each with probability of success λ. The service success indicators, for each server $j = 1, 2$ and state $i = 0, 1$, denoted by $\{I_i^{(j)}(t),\ t = 0, 1, 2, \ldots\}$, are each an i.i.d. sequence with

$$I_i^{(j)}(t)\ \sim\ \text{Bernoulli}\big(\mu_i^{(j)}\big).$$

Moreover, we assume

$$\mu_0^{(2)} \leq \mu_0^{(1)} < \mu_1^{(1)} \leq \mu_1^{(2)}. \tag{1}$$

Hence states $i = 1$ for both servers are better than states $i = 0$. Further, the spread of the chance of success for Server 2 is greater or equal to that for Server 1.

For the environment processes, we restrict attention to a two-state Markov chain, sometimes referred to as a Gilbert–Elliot channel [23]. We denote the probability transition matrix for server j as:

$$P^{(j)} = \left[\begin{array}{cc} \overline{p}_j & p_j \\ q_j & \overline{q}_j \end{array} \right] = \left[\begin{array}{cc} 1 - \gamma_j\,\overline{\varrho}_j & \gamma_j\,\overline{\varrho}_j \\ \overline{\gamma}_j\,\overline{\varrho}_j & 1 - \overline{\gamma}_j\,\overline{\varrho}_j \end{array} \right], \tag{2}$$

with $\overline{x} := 1 - x$. In the sequel, we omit the server index j from the individual parameters of $P^{(j)}$. A standard parametrization of this Markov chain uses transition probabilities $p, q \in [0, 1]$. Alternatively, we may specify the stationary probability of being in state 1, denoted by $\gamma \in [0, 1]$, together with the second eigenvalue of P, denoted by $\varrho \in \big[1 - \min(\gamma^{-1}, \overline{\gamma}^{-1}),\ 1\big]$.

Using this parametrization, ϱ quantifies the time-dependence of the chain; when $\varrho = 0$ the chain is i.i.d., otherwise there is memory. If $\varrho > 0$ then environment states are positively correlated, otherwise they are negatively correlated. Our numerical examples in this paper deal with positive correlation as it is often the more reasonable model for channel memory. The relationship between the (γ, ϱ) parametrization and the (p, q) parametrization is given by $p = \gamma\,\overline{\varrho}$, $q = \overline{\gamma}\,\overline{\varrho}$, $\gamma = p/(p + q)$, and $\varrho = 1 - p - q$. See Figure 2.
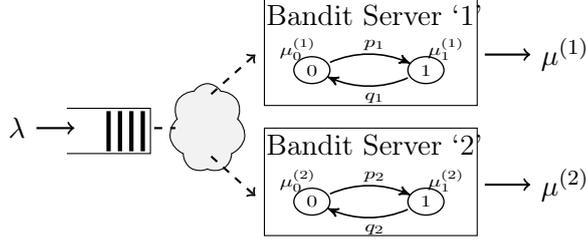
Figure 2: Parameters of the system.

It is instructive to consider the long term behaviour of $I^{(j)}_{X_j(t)}(t)$ for $j = 1, 2$ by assuming the sequence $\{X_j(t),\ t = 0, 1, 2, \ldots\}$ is stationary and thus each $X_j(t)$ is Bernoulli distributed with parameter γ_j. In this case

$$\mathbb{E}[I^{(j)}_{X_j(t)}(t)] = \overline{\gamma}_j\, \mu^{(j)}_0 + \gamma_j\, \mu^{(j)}_1\,,$$

$$\mathrm{Var}[I^{(j)}_{X_j(t)}(t)] = \overline{\gamma}_j\, \mu^{(j)}_0\, \overline{\mu}^{(j)}_0 + \gamma_j\, \mu^{(j)}_1\, \overline{\mu}^{(j)}_1 + \gamma_j\, \overline{\gamma}_j\, \big(\mu^{(j)}_1 - \mu^{(j)}_0\big)^2\,.$$

These quantities are useful for obtaining a rough handle on the performance of the system.

The *stability region* of the system is defined as the set of arrival rates for which there exist a decision rule, π, under which the Markov chain describing the system is positive recurrent[1]. Note that this Markov chain, encapsulates $Q(t)$, but may also include additional state variables. We postulate that stability regions are of the form

$$\{\lambda\ :\ \lambda < \mu^*\},$$

where the *stability bound* μ^* varies according to the observation schemes I–V above and is identical to the maximal throughput rate that may be obtained in a system without a queue (but rather with an infinite supply of jobs). Further, by the construction of the information observation schemes, we have

$$\mu^*_{\mathrm{no}} \leq \mu^*_{\mathrm{queue}} \leq \mu^*_{\mathrm{output}} \leq \mu^*_{\mathrm{state}} \leq \mu^*_{\mathrm{full}}\,, \tag{3}$$

145 where μ^*_{no} corresponds to case V, μ^*_{queue} corresponds to case IV and so forth.

The lower and upper bounds, μ^*_{no} and μ^*_{full}, are easily obtained as we describe now. However, the other cases are more complicated and are analysed in the sections that follow. For the lower and upper bounds we have

$$\mu^*_{\mathrm{no}} = \max\big\{\mathbb{E}[I^{(1)}_{X_1(t)}(t)],\ \mathbb{E}[I^{(2)}_{X_2(t)}(t)]\big\},$$

$$\mu^*_{\mathrm{full}} = \overline{\gamma}_1\overline{\gamma}_2\mu^{(1)}_0 + \gamma_2\mu^{(2)}_1 + \gamma_1\overline{\gamma}_2\mu^{(1)}_1\,.$$

---

[1]There may be different Markov chain representations of this system. One concrete description is $Z(t)$ in Section 5.

7

The lower bound, μ^*_{no} is trivially achieved with a control policy that always uses the server with the higher mean throughput. The upper bound, is achieved with a control policy that uses the best server at any given time. Under the ordering in (1), the throughput in this case is calculated as follows: If both servers are in state 0, then since $\mu_0^{(2)} \leq \mu_0^{(1)}$, the controller selects Server 1. This situation occurs at a long term proportion, $\overline{\gamma}_1 \overline{\gamma}_2$, hence we obtain the first term of μ^*_{full}. The other terms of μ^*_{full} are obtained with a similar argument. Note that when $\gamma_1 = \gamma_2 = \gamma$ and $\mu_i^{(1)} = \mu_i^{(2)} = \mu_i$ for $i = 0, 1$, the expression is reduced to $\mu^*_{\text{full}} = \overline{\gamma}^2 \mu_0 + (1 - \overline{\gamma}^2)\mu_1$ and can be obtained by a Binomial argument.

As a benchmark numerical case, all the examples we present use

$$\gamma = 0.5\,, \ \ \mu_0 = 0.2\,, \ \ \mu_1 = 0.8\,, \tag{4}$$

for both servers. Under these parameters

$$\mu^*_{\text{no}} = 0.5\,, \qquad \text{and} \qquad \mu^*_{\text{full}} = 0.65\,.$$

Hence in the examples that follow, we explore how μ^*_{queue}, μ^*_{output} and μ^*_{state} vary within the interval $[0.5, 0.65]$ as ϱ_j, $j = 1, 2$ varies.

3 Belief States

In implementing a controller for each of the observation schemes, the use of *belief states* reduces both the complexity of the controller and the related analysis. The idea is to summarize the history of observations, $Y(t-1), Y(t-2), \ldots$, into *sufficient statistics* that are updated by the controller. For our model, a natural choice for the belief state of server j is

$$\omega_j(t) = \mathbb{P}\big(X_j(t) = 1 \,|\, \text{Prior knowledge to time } t\big).$$

As we describe now, it is a simple matter to recursively update this sequence in a Bayesian manner. Denoting $\omega_j(t)$ by ω, the believed chance of success is

$$r(\omega) := \overline{\omega}\mu_0 + \omega\mu_1.$$

The updating algorithms (different for each observation scheme) make use of the following:

$$\tau_n(\omega) := \omega\varrho + \gamma\overline{\varrho}\,, \qquad \tau_f(\omega) := \frac{\overline{q}\,\overline{\mu}_1\,\omega + p\,\overline{\mu}_0\,\overline{\omega}}{\overline{r}(\omega)}\,,$$

$$\tau_s(\omega) := \frac{\overline{q}\,\mu_1\,\omega + p\,\mu_0\,\overline{\omega}}{r(\omega)}, \qquad \tau_c(\omega) := \lambda\tau_s(\omega) + \overline{\lambda}\tau_f(\omega)\,. \tag{5}$$

Note that in the above, superscripts j are omitted for clarity. The probabilistic meaning of these functions is described in the sequel. These are used

8

to define recursions for updating the belief state. Each observation scheme entails a different type of recursion:

(II) State observation:

$$\big(\omega_1(t+1), \omega_2(t+1)\big) = \begin{cases} \big(X_1(t), \tau_n^{(2)}(\omega_2(t))\big), & U(t) = 1, \\ \big(\tau_n^{(1)}(\omega_1(t)), X_2(t)\big), & U(t) = 2. \end{cases}$$

(III) Output observation:

$$\big(\omega_1(t+1), \omega_2(t+1)\big) = \begin{cases} \big(\tau_f^{(1)}(\omega_1(t)), \tau_n^{(2)}(\omega_2(t))\big), & I_{X_1(t)}^{(1)}(t) = 0, \\ & \hspace{2cm} U(t) = 1, \\ \big(\tau_s^{(1)}(\omega_1(t)), \tau_n^{(2)}(\omega_2(t))\big), & I_{X_1(t)}^{(1)}(t) = 1, \\[2mm] \big(\tau_n^{(1)}(\omega_1(t)), \tau_f^{(2)}(\omega_2(t))\big), & I_{X_2(t)}^{(2)}(t) = 0, \\ & \hspace{2cm} U(t) = 2. \\ \big(\tau_n^{(1)}(\omega_1(t)), \tau_s^{(2)}(\omega_2(t))\big), & I_{X_2(t)}^{(2)}(t) = 1, \end{cases}$$

(IV) Queue observation:

$$\big(\omega_1(t+1), \omega_2(t+1)\big) = \begin{cases} \big(\tau_f^{(1)}(\omega_1(t)), \tau_n^{(2)}(\omega_2(t))\big), & \Delta Q(t) = 1, \\ \big(\tau_c^{(1)}(\omega_1(t)), \tau_n^{(2)}(\omega_2(t))\big), & \Delta Q(t) = 0, \quad U(t) = 1, \\ \big(\tau_s^{(1)}(\omega_1(t)), \tau_n^{(2)}(\omega_2(t))\big), & \Delta Q(t) = -1, \\[2mm] \big(\tau_n^{(1)}(\omega_1(t)), \tau_f^{(2)}(\omega_2(t))\big), & \Delta Q(t) = 1, \\ \big(\tau_n^{(1)}(\omega_1(t)), \tau_c^{(2)}(\omega_2(t))\big), & \Delta Q(t) = 0, \quad U(t) = 2. \\ \big(\tau_n^{(1)}(\omega_1(t)), \tau_s^{(2)}(\omega_2(t))\big), & \Delta Q(t) = -1, \end{cases}$$

Upon applying the recursions above, we indeed track the belief state as needed:

Proposition 3.1. *For each of the observation schemes, assume that at $t = 0$, $\omega_j(0) = \mathbb{P}\big(X_j(0) = 1\big)$. Then upon implementing the recursion above, based on the observations, it holds that*

$$\omega_j(t) = \mathbb{P}\big(X_j(t) = 1 \,|\, Y(t), Y(t-1), \ldots, Y(0)\big), \qquad t = 1, 2, \ldots.$$

Proof. The proof and derivation of the operators in (5) follows from elementary conditional probabilities and induction. We illustrate this for the

output observation case here. It holds that

$$\mathbb{P}\Big(X(t) = 1 \mid I(t-1) = 0\Big) = \frac{\mathbb{P}\left(X(t) = 1, I(t-1) = 0\right)}{\mathbb{P}\left(I(t-1) = 0\right)}$$

$$= \frac{\mathbb{P}\left(X(t) = 1, I = 0 \mid X = 1\right)\mathbb{P}\left(X = 1\right) + \mathbb{P}\left(X(t) = 1, I = 0 \mid X = 0\right)\mathbb{P}\left(X = 0\right)}{\mathbb{P}\left(I = 0 \mid X = 1\right)\mathbb{P}\left(X = 1\right) + \mathbb{P}\left(I = 0 \mid X = 0\right)\mathbb{P}\left(X = 0\right)},$$

where we denote $X = X(t-1)$ and $I = I(t-1)$. Since $X(t)$ and $I(t-1)$ are conditionally independent given $X(t-1)$, the above numerator can be written as:

$$\mathbb{P}\left(X(t) = 1 \mid X = 1\right)\mathbb{P}\left(I = 0 \mid X = 1\right)\mathbb{P}\left(X = 1\right)$$
$$+ \mathbb{P}\left(X(t) = 1 \mid X = 0\right)\mathbb{P}\left(I = 0 \mid X = 0\right)\mathbb{P}\left(X = 0\right) = \overline{q}\,\overline{\mu}_1\,\omega + p\,\overline{\mu}_0\,\overline{\omega}.$$

Similarly, for the denominator we have:

$$\mathbb{P}\left(I = 0 \mid X = 1\right)\mathbb{P}\left(X = 1\right) + \mathbb{P}\left(I = 0 \mid X = 0\right)\mathbb{P}\left(X = 0\right) = \overline{\mu}_1\omega + \overline{\mu}_0\overline{\omega},$$

which is equal to $\overline{r}(\omega)$. Hence as expected, we find that

$$\mathbb{P}\Big(X(t) = 1 \mid I(t-1) = 0\Big) = \tau_f(\omega).$$

The derivation of $\tau_n(\omega)$ and $\tau_s(\omega)$ follows similar lines. For the queue observation case, notice that

$$\mathbb{P}\Big(X(t) = 1 \mid \Delta Q(t) = 1\Big) = \tau_f(\omega), \qquad \mathbb{P}\Big(X(t) = 1 \mid \Delta Q(t) = -1\Big) = \tau_s(\omega),$$
$$\mathbb{P}\Big(X(t) = 1 \mid \Delta Q(t) = 0\Big) = \lambda\tau_s(\omega) + \overline{\lambda}\tau_f(\omega) = \tau_c(\omega).$$

165 The state observation case follows similar lines. $\qquad\square$

Note that the fixed point of τ_n is the stationary probability γ. The fixed points of τ_f and τ_s are also of interest. When $\varrho \neq 0$ and $\mu_0 \neq \mu_1$, τ_f and τ_s are (real) hyperbolic Möbius transformations of the form $(a\omega+b)/(c\omega+d)$ for $\omega \in [0, 1]$. As such, they each have two distinct fixed points, one stable and

170 one unstable. Here, excluding trivialities where $p, q \in \{0, 1\}$, the stable fixed point of each lies in $(0, 1)$ and is of the form $\left(a - d + \sqrt{(a-d)^2 + 4bc}\right)/2c$ (see also Lemma 2 and 3 of [17]). For τ_f, we have $a = \overline{q}\,\overline{\mu}_1 - p\,\overline{\mu}_0$, $b = p\,\overline{\mu}_0$, $c = \overline{\mu}_1 - \overline{\mu}_0$, and $d = \overline{\mu}_0$. Fixed point of τ_s comes from the same formula by replacing $\overline{\mu}_i$ by μ_i.

Denote by $\omega_i^{(j)}$ the stable fixed point of τ_i for $i = 0, 1$ and $j = 1, 2$. Then

$$\Omega = \Omega_1 \times \Omega_2 \subset [0, 1] \times [0, 1],$$

where we put

$$\Omega_j = [\min(\omega_0^{(j)}, \omega_1^{(j)}), \ \max(\omega_0^{(j)}, \omega_1^{(j)})],$$

175 is the *belief state space* and the limit of any infinite subsequence of the mappings τ_n, τ_f, τ_s and τ_c (for $\omega_1, \omega_2 \in [0, 1]$) lies within Ω, see [21] for more details.

10

4   Maximal Throughput

Having defined sufficient statistics for the belief state and their evolution, the problem of finding a maximally stabilizing control can be posed as a Partially Observable Markov Decision Process (POMDP), see for example [2] or the historical reference [24]. The objective for the POMDP is

$$\mu^* = \sup_\pi \liminf_{T \to \infty} \frac{1}{T} \mathbb{E}_\pi \Big[\sum_{t=0}^{T-1} I(t) \Big],$$

where $U(t) = \pi\big(\omega_1(t), \omega_2(t)\big)$ influences the $I(t)$ as outlined in Section 2. A formal treatment of the POMDP, relating it to the maximal stability region of the system can be carried out. We now first introduce the myopic policy for the POMDP and then move onto optimality equations.

The Myopic Policy

One specific policy is the *myopic policy* given by:

$$\pi(\omega_1, \omega_2) = \begin{cases} \text{Server 2} & \text{if} \quad \omega_2 \geq \frac{\mu_1^{(1)} - \mu_0^{(1)}}{\mu_1^{(2)} - \mu_0^{(2)}} \, \omega_1 + \frac{\mu_0^{(1)} - \mu_0^{(2)}}{\mu_1^{(2)} - \mu_0^{(2)}}, \\ \text{Server 1} & \text{if} \quad \text{otherwise.} \end{cases} \tag{6}$$

The affine threshold in this policy is obtained by comparing the immediate expected mean throughput for any given pair (ω_1, ω_2) and choosing the server that maximizes it. Such a policy is attractive in that it is easy to implement. Further, when the servers are symmetric (all parameters are identical), it holds from symmetry that it is optimal. In this case it can be represented as

$$\pi(\omega_1, \omega_2) = \mathrm{argmax}_{i=1,2} \, \omega_i,$$

and we refer to it as the *symmetric myopic policy*.

Simulation Result

Figure 3 demonstrates results obtained through a Monte Carlo simulation[2] of the model for observation schemes (I)–(V). We use the parameters in (4) and vary ϱ with $\varrho_1 = \varrho_2 = \varrho$ in the range $[0, 1]$ with steps of 0.01. The policy used is the symmetric myopic policy and is optimal since the servers are identical.

As we see from the figure, the ordering (3) appears to hold. Further, at the i.i.d. case $\varrho = 0$, historical observations are not useful and the throughput of all observations schemes, except for full observation, is at 0.5. At the other extreme, when $\varrho \to 1$, we have that the state observation scheme converges to a throughput identical to that of the full observation scheme. This is

---

[2]Simulation details: We run the process for $t = 5,000,000$ time units, using common random numbers for each run and recording average throughput.
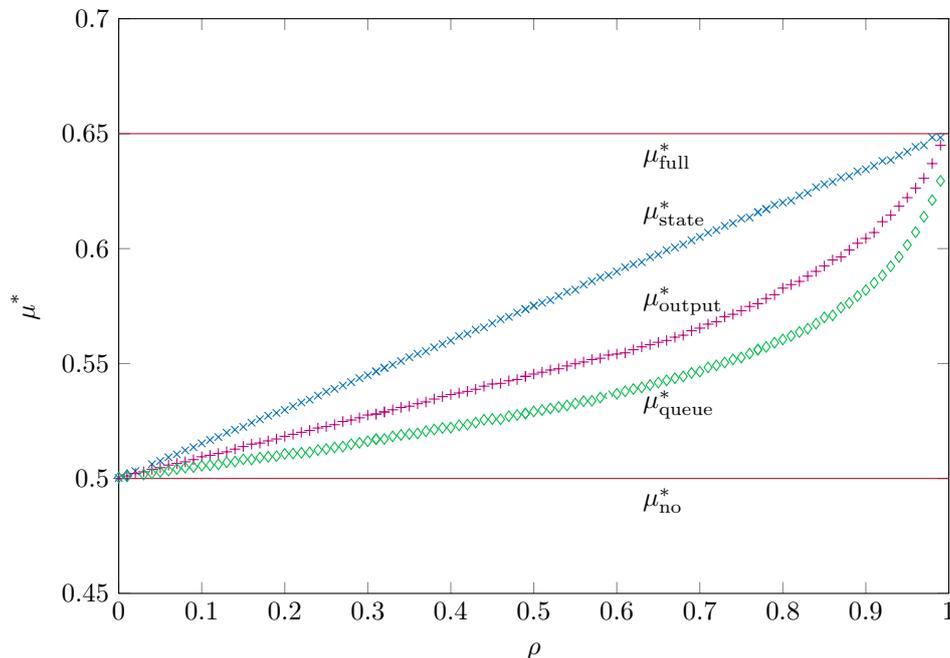
Figure 3: The stability bound is displayed as a function of ϱ for the various observation schemes. This plot is based on simulation results using the parameters in (4) and $\varrho_1 = \varrho_2 = \varrho$.

because at that regime, the server state environments rarely change. Thus, from a throughput perspective, the controller behaves as though it has full information. On the other hand, even at $\varrho = 1$, the output observation scheme and queue observation scheme still performs at a lower throughput. Finally, for $\varrho \in (0,1)$ it is evident that there is a gap in performance for each observation scheme. This gap quantifies the value of information in controlling our model and motivates further analysis.

Bellman Equations

It is well-known that the optimal policy that maximizes the throughput follows from the average reward Bellman equations. See [22] for background on Markov Decision Processes (MDP) or [7] for a discussion of average reward optimality with such state spaces. The Bellman equation is then

$$\mu^* + h(\omega_1, \omega_2) = \max\left\{h^{(1)}(\omega_1, \omega_2),\, h^{(2)}(\omega_1, \omega_2)\right\},$$

where h is the relative value function and the individual components $h^{(j)}(\cdot, \cdot)$, vary as follows:

12

(II) State observation:

$$h^{(1)}(\omega_1, \omega_2) := r^{(1)}(\omega_1) + \left[\overline{\omega}_1\, h\big(p_1, \tau_n^{(2)}(\omega_2)\big) + \omega_1\, h\big(\overline{q}_1, \tau_n^{(2)}(\omega_2)\big)\right],$$

$$h^{(2)}(\omega_1, \omega_2) := r^{(2)}(\omega_2) + \left[\overline{\omega}_2\, h\big(\tau_n^{(1)}(\omega_1), p_2\big) + \omega_2\, h\big(\tau_n^{(1)}(\omega_1), \overline{q}_2\big)\right].$$

(III) Output observation:

$$h^{(1)}(\omega_1, \omega_2) := r^{(1)}(\omega_1) + \left[\overline{r}^{(1)}(\omega_1)\, h\big(\tau_f^{(1)}(\omega_1), \tau_n^{(2)}(\omega_2)\big) + r^{(1)}(\omega_1)\, h\big(\tau_s^{(1)}(\omega_1), \tau_n^{(2)}(\omega_2)\big)\right],$$

$$h^{(2)}(\omega_1, \omega_2) := r^{(2)}(\omega_2) + \left[\overline{r}^{(2)}(\omega_2)\, h\big(\tau_n^{(1)}(\omega_1), \tau_f^{(2)}(\omega_2)\big) + r^{(2)}(\omega_2)\, h\big(\tau_n^{(1)}(\omega_1), \tau_s^{(2)}(\omega_2)\big)\right].$$

(IV) Queue observation:

$$h^{(1)}(\omega_1, \omega_2) := r^{(1)}(\omega_1) + \Big[\lambda \overline{r}^{(1)}(\omega_1)\, h\big(\tau_f^{(1)}(\omega_1), \tau_n^{(2)}(\omega_2)\big) + \overline{\lambda} r^{(1)}(\omega_1)\, h\big(\tau_s^{(1)}(\omega_1), \tau_n^{(2)}(\omega_2)\big)$$
$$+ \Big(\overline{\lambda}\overline{r}^{(1)}(\omega_1) + \lambda r^{(1)}(\omega_1)\Big)\Big(h(\tau_c^{(1)}(\omega_1), \tau_n^{(2)}(\omega_2))\Big)\Big],$$

$$h^{(2)}(\omega_1, \omega_2) := r^{(2)}(\omega_2) + \Big[\lambda \overline{r}^{(2)}(\omega_2)\, h\big(\tau_n^{(1)}(\omega_1), \tau_f^{(2)}(\omega_2)\big) + \overline{\lambda} r^{(2)}(\omega_2)\, h\big(\tau_n^{(1)}(\omega_1), \tau_s^{(2)}(\omega_2)\big)$$
$$+ \Big(\overline{\lambda}\overline{r}^{(1)}(\omega_2) + \lambda r^{(1)}(\omega_2)\Big)\Big(h(\tau_n^{(1)}(\omega_1), \tau_c^{(2)}(\omega_2))\Big)\Big].$$

The optimal decision is then to choose Server 1 if and only if $h^{(1)}(\omega_1, \omega_2) \geq h^{(2)}(\omega_1, \omega_2)$, breaking ties arbitrarily. Since $\tau_n^{(j)}(0) = \tau_s^{(j)}(0) = \tau_f^{(j)}(0) = p_j$ and $\tau_n^{(j)}(1) = \tau_s^{(j)}(1) = \tau_f^{(j)}(1) = \overline{q}_j$, for all three aforementioned cases we have:

$$\mu^* + h(0,0) = \max\left\{\mu_0^{(1)} + h(p_1, p_2), \mu_0^{(2)} + h(p_1, p_2)\right\},$$

$$\mu^* + h(1,1) = \max\left\{\mu_1^{(1)} + h(\overline{q}_1, \overline{q}_2), \mu_1^{(2)} + h(\overline{q}_1, \overline{q}_2)\right\}. \tag{7}$$

From the ordering in (1), the above equations imply that at point $(\omega_1, \omega_2) = (0,0)$, choosing Server 1 is optimal and at point $(\omega_1, \omega_2) = (1,1)$ choosing Server 2 is optimal. This observation gives some initial insight into the structure of optimal policies. We now purse these further numerically.

210 Numerical Investigation of Optimal Policies

A solution to the above Bellman equations can be obtained numerically using relative value iteration and discretization of the belief state space, Ω. Here we consider that each interval $[0,1]$ for ω_1 and ω_2 is partitioned to 1000 equal sub-intervals. We then run relative value iteration with an accepted 215 error set to $\epsilon = 0.0001$.

Our various numerical experiments indicate the following:

1. The ordering in (3) holds.

13

2. Increasing (positive) ϱ_j always yields an increase in μ^*.

3. Though the myopic policy does not appear to be generally optimal, when both servers are identical, the optimal policy is the symmetric myopic policy.

4. In all cases, the optimal policy is given by a non-decreasing switching curve within Ω. That is, there exists a function $\omega_2^*(\omega_1)$ where the optimal policy is

$$\pi(\omega_1, \omega_2) = \begin{cases} \text{Server 2} & \text{if} & \omega_2 \geq \omega_2^*(\omega_1), \\ \text{Server 1} & \text{if} & \text{otherwise.} \end{cases}$$

5. When the ordering in (1) has strict inequalities, $\omega_2^*(0) > 0$ and $\omega_2^*(1) < 1$.

6. For identical servers, it holds that the switching curve for the output observation case is sandwiched between the switching curve of the state observation case and the myopic switching line (6).

7. The switching curve for the queue observation case depends on λ. Further, when λ is at either of the extreme points ($\lambda = 0$ or $\lambda = 1$), the queue observation case agrees with the output observation case.

As one illustration of some of the above properties, consider Table 1 based on the parameters of (4) and various values of ϱ_1 and ϱ_2. The results in the table further affirm comments 1 and 3 above and contains values that agree with Monte Carlo simulation results, similar to those of Figure 3.

ϱ_1	ϱ_2	μ_{state}^*	μ_{output}^*	μ_{queue}^*
0.2	0.5	0.5543	0.5314	0.5190
0.4	0.5	0.5673	0.5400	0.5231
0.6	0.5	0.5823	0.5489	0.5289
0.8	0.5	0.6009	0.5647	0.5360

Table 1: Stability region bounds for observations schemes (II)-(IV) for various ϱ_1 and ϱ_2 values. Note the queue observation case is with $\lambda = 0.5$.

As a further illustration, Figure 4 shows switching curves, $w_2^*(\cdot)$ for the parameters of (4) with $\varrho_1 = 0.5$ and $\varrho_2 = 0.7$. In the figure, the red dotted line is the myopic policy line (suboptimal). The blue solid curve is the switching curve for the output observation case. The green loosely dashed line is related to the queue observation case and the orange densely dashed curve is the switching curve for the state observation case. These curves were obtained by finding the optimal decision for every (discretized) element of Ω and then observing the switching curve structure.
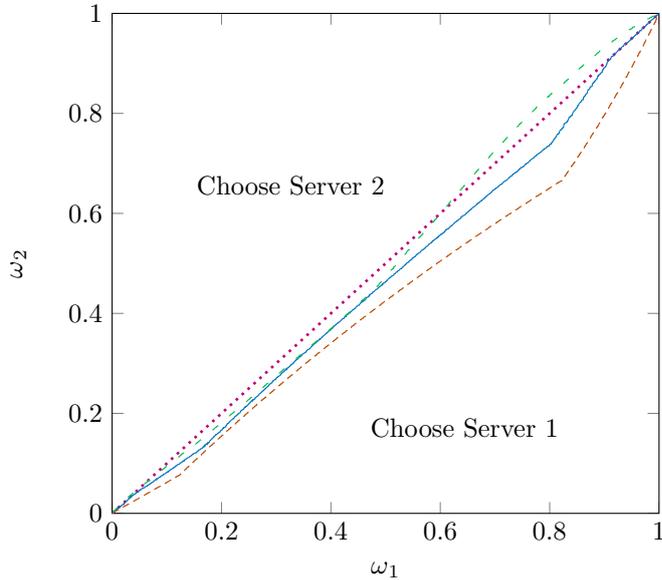
····· Myopic threshold – – Opt. threshold IV ——Opt. threshold III – – Opt. threshold II

Figure 4: Myopic and optimal policies for the state observation Case II, output observation Case III and queue observation Case IV ($\lambda = 0.5$). This is for a system with $\varrho_1 = 0.5$ and $\varrho_2 = 0.7$.

5 QBD Structured Models for Finite State Controllers

Here we illustrate how Matrix-Analytic Modelling (MAM) can be used to analyse a finite state controller that approximates an optimal controller. Our analysis is for the output observation scheme (Case IV). Similar analysis can be applied to the other observation schemes as well.

A finite state controller operates by using a finite discrete belief state $\widetilde{\Omega}$, representing a discrete grid in Ω. With such a controller, we consider the whole system as a Quasi-Birth-and-Death (QBD) process (for more details about QBD process see for example [14]). Using the QBD structure, we find a matrix analytic expression for μ^*_{output} (denoted by μ^* in this section).

Take $\widetilde{\Omega} = \{1, \ldots, M\}^2$ and define the controller state at time t by $(\psi_1(t), \psi_2(t)) \in \widetilde{\Omega}$. In doing so, we treat $\psi_j(t)$ as $\lceil M\omega_j(t) \rceil$. The controller action is (potentially) randomized based on a matrix of probabilities C so that Server 2 is chosen with probability $C_{(\psi_1(t), \psi_2(t))}$ and otherwise the choice is Server 1. That is, the matrix C is a randomized control policy. Such a

15

policy encodes information as in Figure 4.

The controller state is updated in a (potentially) randomized manner based on the $M \times M$ stochastic matrices $N^{(j)}, S^{(j)}, F^{(j)}$ for $j = 1, 2$ as follows: if Server 1 was not selected (either because there were no jobs in the queue, or because the other server was selected), the distribution of the new state is $\big(N^{(1)}_{\psi_1(t),1}, \ldots, N^{(1)}_{\psi_1(t),M}\big)$; that is taken from the row indexed by $\psi_1(t)$. Similarly, if Server 1 was chosen and service was successful $(I = 1)$, the distribution of the new state is $\big(S^{(1)}_{\psi_1(t),1}, \ldots, S^{(1)}_{\psi_1(t),M}\big)$. Finally, if Server 1 was chosen and the service failed $(I = 0)$, the distribution of the new state is $\big(F^{(1)}_{\psi_1(t),1}, \ldots, F^{(1)}_{\psi_1(t),M}\big)$. Similarly, for Server 2, we have $\big(N^{(2)}_{1,\psi_2(t)}, \ldots, N^{(2)}_{M,\psi_2(t)}\big)$, $\big(S^{(2)}_{1,\psi_2(t)}, \ldots, S^{(2)}_{M,\psi_2(t)}\big)$ and $\big(F^{(2)}_{1,\psi_2(t)}, \ldots, F^{(2)}_{M,\psi_2(t)}\big)$, respectively. Therefore, the rows of matrices $N^{(j)}, S^{(j)}$ and $F^{(j)}$ for $j = 1, 2$ indicate how to (potentially randomly) choose the next controller state. Here, S stands for Success, F for Failure and N for No service.

We construct the matrices $N^{(j)}, S^{(j)}, F^{(j)}$ based on a discretization of $\tau_n^{(j)}, \tau_s^{(j)}$ and $\tau_f^{(j)}$, respectively. For example, construction of S from τ_s is as follows: construct the elements of S by

$$
\begin{cases}
S_{i,j} = 1 & j \text{ is an integer} \\
S_{i,\lfloor j \rfloor} = 1 & 1 \le \lfloor j \rfloor \le M \\
S_{i,\lceil j \rceil} = 1 & otherwise
\end{cases}
\qquad \text{with} \qquad j = M\tau_s\Big(\frac{i-1}{M}\Big),
$$

and $S_{i,k} = 0$ for all other elements with $i, k = 1, \ldots, M$. After this, we ensure irreducibility of this matrix by fixing $\epsilon > 0$ (e.g. $= 0.001$ as in our numerical examples) and adding ϵ/M to each elements of the matrix and then renormalizing it.

The matrices F and N are constructed in a similar way based on τ_f and τ_n, respectively. This is simply a mechanism to encode the transition operators over the finite grid. Hence the matrices $N^{(j)}, S^{(j)}, F^{(j)}$ describe propagation of ψ_j through the belief operators, similarly to the propagation of ω through their continuous counterparts.

Now, given such a controller with

$$
\text{Controller parameters} = \Big(N^{(1)}, S^{(1)}, F^{(1)}, N^{(2)}, S^{(2)}, F^{(2)}, C\Big),
$$

we construct a Markov chain, $Z(t)$ for the system. The state of this model at time t is given by the queue length, server environment state, and controller state as follows:

$$
Z(t) = \Big(\underbrace{Q(t)}_{\text{Level}}, \big(\overbrace{(X_1(t), X_2(t))}^{\text{Servers}}, \overbrace{(\psi_1(t), \psi_2(t))}^{\text{Controller}} \big) \Big) \in \{0, 1, \ldots\} \times \{1, 2\}^2 \times \{1, \ldots, M\}^2.
$$

Explicit QBD Construction

When the states of $Z(t)$ are lexicographically ordered, with first component countably infinite (levels) and the other components finite (phases), the resulting (infinite) probability transition matrix is of the QBD form:

$$
A = \begin{bmatrix}
\widetilde{A}_0 & \widetilde{A}_1 & & & 0 \\
A_{-1} & A_0 & A_1 & & \\
& A_{-1} & A_0 & A_1 & \\
& & A_{-1} & A_0 & A_1 \\
0 & & & \ddots & \ddots & \ddots
\end{bmatrix}, \tag{8}
$$

where each of $\widetilde{A}_0$, $\widetilde{A}_1$, A_{-1}, A_0, and A_1 is a block matrix of order $4M^2$ as we construct below.

The matrix A_{-1} represents the phase transition where there is a one level decrease. Similarly, the matrix A_1 represents phase transition where there is a one level increase and A_0 represents the phase transition where the level remains the same. The blocks are constructed as follows:

$$
\widetilde{A}_0 = \overline{\lambda}\widetilde{N}, \quad \widetilde{A}_1 = \lambda\widetilde{N},
$$

$$
A_{-1} = \overline{\lambda}\widetilde{S}, \quad A_0 = \overline{\lambda}\widetilde{F} + \lambda\widetilde{S}, \quad A_1 = \lambda\widetilde{F}, \tag{9}
$$

285 where the matrices, $\widetilde{S}, \widetilde{F}, \widetilde{N}$ (each of order $4M^2$) denote the change of phase together with a service success, service failure or no service attempt, respectively. For instance, the (i,j)-th entry of $\widetilde{S}$ is the chance of a service success together with a change of phase from i to j (note that i and j are each 4-tuples). The sum $\widetilde{S} + \widetilde{F}$ is a stochastic matrix (as is evident from the
290 construction below). Similarly, $\widetilde{N}$ is a stochastic matrix. Hence the overall transition probability (infinite) matrix A is stochastic as well.

To construct $\widetilde{S}, \widetilde{F}$ and $\widetilde{N}$, we define $M^2 \times M^2$ matrices $\widetilde{S}\_{k\ell}$, $\widetilde{F}\_{k\ell}$ and $\widetilde{N}\_{k\ell}$ for $k, \ell = 0, 1$. Taking $\widetilde{S}\_{k\ell}$ as an example, its (i,j)-th entry (each represented as a 2-tuple), describes the chance of a success together with a transition of belief state from i to j, when the environment of the first server is in state k and that of the second server is in state ℓ. Here i and j, each represent the overall system belief state in lexicographic order. That is, we should refer to i as (i_1, i_2) and similarly to j. A similar interpretation holds for $\widetilde{F}\_{k\ell}$ and $\widetilde{N}\_{k\ell}$. These aforementioned matrices are constructed (for $k, \ell = 0, 1$) as follows:

$$
\widetilde{S}\_{k\ell} = \mu_\ell^{(2)}\left(diag\big(vec(C')\big)\right)(N^{(1)} \otimes S^{(2)}) + \mu_k^{(1)}\left(diag\big(vec(\overline{C}')\big)\right)(S^{(1)} \otimes N^{(2)}),
$$

$$
\widetilde{F}\_{k\ell} = \overline{\mu}_\ell^{(2)}\left(diag\big(vec(C')\big)\right)(N^{(1)} \otimes F^{(2)}) + \overline{\mu}_k^{(1)}\left(diag\big(vec(\overline{C}')\big)\right)(F^{(1)} \otimes N^{(2)}),
$$

$$
\widetilde{N}\_{k\ell} = (N^{(1)} \otimes N^{(2)}),
$$

17

where $diag(\cdot)$ is an operation taking a vector and resulting in a diagonal matrix with the vector in the diagonal, $vec(\cdot)$ is an operation taking a matrix and resulting in a vector with the columns of the matrix stacked up one by one, and $\otimes$ is the standard Kronecker product.

To see the above, let us consider (for e.g.) an element of the matrix $\widetilde{S}_{k\ell}$ at coordinate $i = (i_1, i_2)$ and $j = (j_1, j_2)$. This describes the probability of the event

$W = \{$Success of service together with a transition to belief state $(j_1, j_2)\}$,

where $X_1 = k$, $X_2 = \ell$, $\psi_1 = i_1$, and $\psi_2 = i_2$. The event W can be partitioned into W_1 (service attempt was on 1) and W_2 (service attempt was on 2). The chance of W_2 is C_{i_1,i_2}. With choosing Server 2 the success probability is $\mu_\ell^{(2)}$. Then under the event W_2, the belief state of Server 1 will be updated according to $N^{(1)}$ and the belief state of Server 2 with $S^{(2)}$. The $M^2 \times M^2$ matrix $diag(vec(C'))$ is a diagonal matrix where its diagonal elements are the rows of the matrix C, each represent the chance of $U = 2$.

With the matrices $\widetilde{S}_{k\ell}$, $\widetilde{F}_{k\ell}$ and $\widetilde{N}_{k\ell}$ (for $k, \ell = 0, 1$) in hand, we construct the matrices $\widetilde{S}, \widetilde{F}$ and $\widetilde{N}$ as:

$$\widetilde{S} = (P^{(1)} \otimes P^{(2)}) \circledast \begin{bmatrix} \widetilde{S}_{00} & 0 & 0 & 0 \\ 0 & \widetilde{S}_{01} & 0 & 0 \\ 0 & 0 & \widetilde{S}_{10} & 0 \\ 0 & 0 & 0 & \widetilde{S}_{11} \end{bmatrix},$$

$$\widetilde{F} = (P^{(1)} \otimes P^{(2)}) \circledast \begin{bmatrix} \widetilde{F}_{00} & 0 & 0 & 0 \\ 0 & \widetilde{F}_{01} & 0 & 0 \\ 0 & 0 & \widetilde{F}_{10} & 0 \\ 0 & 0 & 0 & \widetilde{F}_{11} \end{bmatrix},$$

$$\widetilde{N} = (P^{(1)} \otimes P^{(2)}) \circledast \begin{bmatrix} \widetilde{N}_{00} & 0 & 0 & 0 \\ 0 & \widetilde{N}_{01} & 0 & 0 \\ 0 & 0 & \widetilde{N}_{10} & 0 \\ 0 & 0 & 0 & \widetilde{N}_{11} \end{bmatrix},$$

where $P^{(j)}$ for $j = 1, 2$ are the 2×2 probability transition matrices of the servers given by (2) and operation $\circledast$ is defined as

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \circledast \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix} = \begin{bmatrix} a_{11}A & a_{12}A \\ a_{21}B & a_{22}B \end{bmatrix}.$$

Putting all of the above components together yields the probability transition matrix of $Z(t)$, A.

Stability Criterion

18

A well-known sufficient condition for positive recurrence (stability) of QBDs such as $Z(t)$ is

$$\pi_\infty \big(A_1 - A_{-1}\big)\mathbf{1} < 0,$$

where π_∞ is the stationary distribution of the (finite) stochastic matrix $A_{-1} + A_0 + A_1$ and $\mathbf{1}$ is a column vector of ones. From (9), we see that this is also the stationary distribution of $\widetilde{S} + \widetilde{F}$ which does not depend on λ. This property of our QBD allows us to represent the stability criterion as

$$\lambda < \mu^* = \pi_\infty \widetilde{S}\mathbf{1}, \qquad (10)$$

310 with μ^* depending on the controller and system parameters but not depending on λ.

In addition to the stability criteria, a further virtue of modelling the system as a QBD is that we can use the vast body of MAM knowledge and algorithms for analysing the system and ultimately optimizing controllers.
315 Nevertheless, our focus in this paper is on stability.

Numerical Illustration

We now use our QBD model and the stability criterion (10) to explore the performance of finite state controllers.
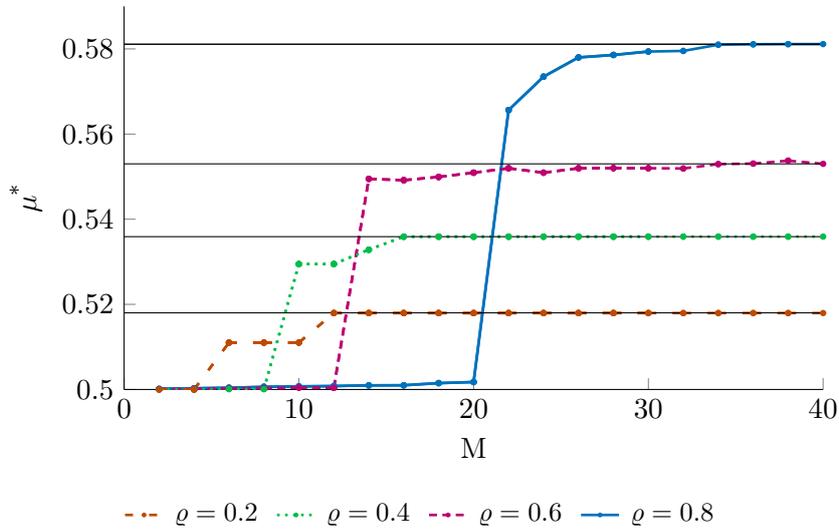

Figure 5: Stability bound achieved by finite state controllers for observation scheme III with increasing M, computed by (10). The limiting horizontal lines are at μ^* computed by means of relative value iteration of Bellman equations.

In doing so, we consider the parameters as in (4) with $\varrho_1 = \varrho_2 = \varrho$. Since in this situation, the servers are identical, the symmetric myopic policy is

19

optimal and we thus restrict attention to a matrix C with

$$C_{i,j} = \begin{cases} 1, & i < j, \\ 0.5, & i = j, \\ 0, & i > j. \end{cases}$$

Using these parameters, we evaluated (10) for increasing M and for various values of ϱ. The results are in Figure 5. As expected, the performance of the finite state controller converges to that found by numerical solution of the Bellman equations as in the previous section. The sudden increase in performance as M increases (e.g. at $M = 20$ for $\varrho = 0.8$) can be attributed to discretization phenomena. For reference, the values of μ^* obtained by Bellman equation (as well as the QBD when $M \to \infty$) are 0.5179, 0.5359, 0.5539 and 0.5815 for $\varrho = 0.2$, 0.4, 0.6 and 0.8, respectively.

6 Outlook

This paper described some results from a research effort attempting to handle control of stochastic systems with partial observations where the control decision influences the observation made. Explicit analysis of such systems is extremely challenging as is evident by both the complicated Bellman equations and the QBD structure that we put forward (even for a simple system as we consider). Nevertheless, insights obtained on the role of information, e.g. the effect of the observation scheme (I–V) on system stability are of interest.

Our model and numerical results, pave the way for explicit proofs of some of the structural properties outlined above. Moreover, the analysis remains to be extended to more general server environment models, as well as systems with more queues and control decisions. Related work is in [21], an earlier paper that leads to this work. An aspect in [21] that remains to be further considered is the networked case where the authors investigated (through simulation) cases in which the relationship of stability and throughput is not as immediate as in our current paper. A further related (recent) paper, [19], deals with a situation similar to our output observation case (III). In that paper, the authors consider the Whittle index applied to a similar system (without considering a queue and stability). Relating the Whittle index and system stability, is a further avenue that requires investigation.

Acknowledgement

20

References

[1] F. Baccelli and M. M. Armand, Stability and bounds for single server queues in random environment. Stochastic Models 2(2):281–291, 1986.

[2] N. Bäuerle and U. Rieder, Markov decision processes with applications to finance. Springer Science & Business Media, 2011.

[3] D. A. Bini, B. Meini, S. Steffé, and B. Van Houdt, Structured Markov chains solver: software tools. Proceeding from SMCTools, 14, 2006.

[4] F. Cecchi and P. Jacko, Nearly-optimal scheduling of users with Markovian time-varying transmission rates. Performance Evaluation, 99 (C):16–36, Elsevier Science Publishers BV, 2016.

[5] H. Halabian, I. Lambadaris, and C. H. Lung, Explicit characterization of stability region for stationary multi-queue multi-server systems. IEEE Trans. Autom. Control, 59(2):355–370, 2014.

[6] Q. M. He, Fundamentals of matrix-analytic methods. Springer, 2014.

[7] O. Hernández-Lerma and J. B. Lasserre, *Discrete-time Markov Control Processes: Basic Optimality Criteria*. Springer, 2012.

[8] L. A. Johnston and K. Vikram, Opportunistic file transfer over a fading channel: A POMDP search theory formulation with optimal threshold policies. IEEE Transactions on Wireless Communications 5(2): 394–405, 2006.

[9] O. Kella and W. Whitt, A storage model with a two-state random environment. Operations Research 40(3) supplement-2:257–262, 1992.

[10] G. Koole, Z. Liu, and R. Righter. Optimal transmission policies for noisy channels. Operations Research, 49(6):892–899, 2001.

[11] J. Kuhn and Y. Nazarathy, Wireless Channel Selection with Reward-Observing Restless Multi-armed Bandits, draft book chapter, submitted, 2015.

[12] M. Larranaga, M. Assaad, A. Destounis, and G. S. Paschos, Asymptotically optimal pilot allocation over Markovian fading channels, arXiv preprint arXiv:1608.08413, 2016.

[13] M. Larrañaga, U. Ayesta, and I. M. Verloop, Index policies for a multi-class queue with convex holding cost and abandonments. ACM SIGMETRICS Performance Evaluation Review, ACM, 42(1):125–137, 2014.

[14] G. Latouche and V. Ramaswami, Introduction to Matrix Analytic Methods in Stochastic Modeling. SIAM, 1999.

21

[15] C. P. Li and M. J. Neely, Network utility maximization over partially observable Markovian channels. Performance Evaluation, 70(7):528–548, 2013.

[16] K. Liu and Q. Zhao, Indexability of restless bandit problems and optimality of Whittle index for dynamic multichannel access. IEEE Trans. Inf. Theory, 56(11):5547–5567, 2010.

[17] I. M. MacPhee and B. P. Jordan, Optimal search for a moving target. Probab. Eng. Inform. Sc., 9(2):159–182, 1995.

[18] S. Meyn, Control techniques for complex networks. Cambridge University Press, 2008.

[19] R. Meshram, D. Manjunath, and A. Gopalan, On the Whittle index for restless multi-armed hidden Markov bandits. arXiv preprint arXiv:1603.04739, 2016.

[20] A. Mészáros and M. Telek Markov decision process and linear programming based control of MAP/MAP/N queues. European Workshop on Performance Engineering, 179–193, Springer, 2014.

[21] Y. Nazarathy, T. Taimre, A. Asanjarani, J. Kuhn, B. Patch, and A. Vuorinen, The challenge of stabilizing control for queueing systems with unobservable server states. 5th Australian Control Conference (AUCC), IEEE, 342–347, 2015.

[22] M. L. Puterman, Markov decision processes: discrete stochastic dynamic programming. John Wiley & Sons, 2014.

[23] P. Sadeghi, R. A. Kennedy, P. B. Rapajic, and R. Shams, Finite-state Markov modelling of fading channels: A survey of principles and applications. IEEE Signal Process. Mag., 25(5):57–80, 2008.

[24] R. D. Smallwood and E. J. Sondik, The optimal control of partially observable Markov processes over a finite horizon. Operations Research, 21(5):1071–1088, 1973.

[25] L. Tassiulas and A. Ephremides, Dynamic server allocation to parallel queues with randomly varying connectivity. IEEE Trans. Inf. Theory, 39(2):466–478, 1993.

[26] I. M. Verloop, Asymptotically optimal priority policies for indexable and non-indexable restless bandits. Annals of Applied Probability 26(4):1947–1995, 2016.

[27] P. Whittle, Restless bandits: Activity allocation in a changing world. Journal of Applied Probability, 25:287–298, 1988.

[28] P. Whittle, Optimization over time. John Wiley Sons, Inc., 1982.