Dynamic scheduling and maintenance for a two-class queue with a deteriorating server

Jefferson Huang[1], Douglas G. Down[2], Mark E. Lewis[1], and Cheng-Hung Wu[3]

***Abstract*— We consider a queueing control problem motivated by a scheduling and maintenance problem in semiconductor manufacturing. Conditions are given under which it is optimal, relative to both infinite-horizon discounted-cost and average-cost criteria, to perform scheduling according to a priority policy, and maintenance according to a threshold policy. We also provide empirical evidence that, even when the aforementioned conditions do not hold, such policies provide nearly-optimal performance. In addition, the empirical results indicate the importance of performing preventive maintenance.**

I. INTRODUCTION

In this paper, we provide results on the structure of optimal policies, under both discounted-cost and average-cost criteria, for a generalization of a classic scheduling problem in queueing control. Namely, there are two classes of jobs, where each class can be viewed as having its own queue. Arrivals occur according to class-dependent, arbitrary point processes, and the class-dependent service requirements are independent and exponential. In addition, service is performed by a single server, and at most one customer can be in service at any given time. So far, this describes a G/M/1 queue with two customer classes. It is well-known that a simple priority policy is optimal for this problem [5, Theorem 2.1].

This model differs from the one we will consider in two ways. First, we allow the service rate of the single server to vary over time. In particular, at every point in time the server is in one of a finite set of states $\{0,1,\ldots,B\}$. The server's service rate depends on its state; if the current state is s, it can process jobs in queue $i \in \{1,2\}$ at a rate of μ_i^s per unit time. This can be used to model server deterioration by, for instance, letting μ_i^s increase in s for each class i.

The second way our problem differs from the usual two-class G/M/1 scheduling problem is that we allow the decision-maker to exert control over the server state. In particular, at every decision epoch, the decision-maker may elect to perform an action that, with probability one, will make the server transition to state B. In the case of a deteriorating server where state B is the "like-new" state, this can be interpreted as performing server maintenance or replacement; in the rest of the paper, we will also use "maintenance" to refer to exerting control over the server state in general. When there is at least one job in the system, the decision-maker may, in lieu of maintaining the server, elect to work on one of the jobs. In this case, when at least one job of each class is present, the server must also decide on the job class that should be served. There is an immediate cost incurred in moving the server to state B, and class-dependent holding costs for jobs in queue.

When control is not exerted on the server state, the server-state process evolves as follows. If the current state is $s \geq 1$, the next state is $s-1$ with probability one, and if $s = 0$ then the next state is B with probability one. Moreover, the sojourn times in each state are random, and in general need not be independent and identically distributed. However, for the case of a deteriorating server, our results on the structure of optimal maintenance decisions will require this; see Assumption 3 in Section IV.

A motivation for the model described above is the following quality control problem in semiconductor manufacturing; for examples of how queueing theory can be applied to semiconductor manufacturing more generally, see e.g., Shanthikumar et al. [6]. Prior to being shipped, each type of chip that is being produced needs to be tested to confirm that it performs according to certain standards. The tests are performed using a device that can be assigned to at most one type of chip at a time. In the model that we consider, each customer class corresponds to a type of chip that needs to be tested, and the server corresponds to the testing device. In practice, there are typically several such devices working in parallel; extending our results to the case of multiple servers is a promising direction for future work. More generally, our work continues a line of research on applications of queueing theory to problems in machine maintenance, for which an extensive literature exists; see e.g., Cai et al. [2], Celen & Djurdjanovic [3], Kaufman & Lewis [4], Sloan & Shanthikumar [7], and Yao et al. [9]. The structural results we present in Section IV can be viewed as generalizations, to the case of two job classes, of those derived in Kaufman & Lewis [4]. In addition, a model with two job classes that is closely related to ours is studied in Cai et al. [2].

The rest of the paper is organized as follows. In Section II, the queueing control problem is defined. Next, results on the structure of optimal policies are provided in Sections III and IV. Finally, in Section V we present emprical evidence suggesting that policies with the structure considered in Sections III and IV are nearly optimal. The results in Section V also indicate the importance of allowing for preventive maintenance.

[1]Jefferson Huang and Mark E. Lewis are with the School of Operations Research and Information Engineering, Cornell University, Ithaca, NY 14850, USA jh2543@cornell.edu

[2]Douglas G. Down is with the Department of Computing and Software, McMaster University, Hamilton, Ontario L8S 4L7, Canada

[3]Cheng-Hung Wu is with the Institute of Industrial Engineering, National Taiwan University, 1, Sec. 4, Roosevelt Rd., Taipei 106, Taiwan

II. MODEL DESCRIPTION

Jobs arrive at queues 1 and 2 at random times, according to independent point processes on the set of nonnegative real numbers $[0,\infty)$. Each job arriving at queue $k \in \{1,2\}$ has a service requirement that is exponentially distributed with rate 1, and incurs holding costs while it occupies that queue at a rate of $h_k \in [0,\infty)$ per unit time. The jobs in both queues are to be completed using a single server whose state at time $t = 0$ is $B \in \{1,2,\dots\}$. When the state of the server is $s \in \{0,1,\dots,B\}$, it can complete jobs in queue $k \in \{1,2\}$ at a rate of μ_k^s per unit time. We assume that the server is unusable while it is in state 0, i.e., $\mu_1^0 = \mu_2^0 = 0$. Moreover, the random times at which the server changes state are described by a point process on $[0,\infty)$.

The decision-maker may choose to initiate the maintenance of the server when it is in a state $s \geq 0$, in which case a fixed cost $K(s) > 0$ is incurred. At each *decision epoch* (i.e., each arrival, service completion, and change in server state), the decision-maker must decide whether to serve queue 1 or 2, or to perform server maintenance. We assume that jobs undergoing service may be preempted. In the sequel, we will consider the existence and structure of optimal policies for this server scheduling and maintenance problem under various additional assumptions.

A. Policies and optimality criteria

Let $\mathbb{N} := \{0,1,\dots\}$, and denote the *state set* by

$$\mathbb{X} := \mathbb{N} \times \mathbb{N} \times \{0,1,\dots,B\},$$

where state $(i,j,s) \in \mathbb{X}$ indicates that there are i jobs in queue 1, j jobs in queue 2, and the server is in state s. If the system is in state $(i,j,s) \in \mathbb{X}$, the set of available actions is

$$A(i,j,s) := \begin{cases} \{1,2,R\}, & i,j \geq 1,\ s \geq 1, \\ \{1,R\}, & i \geq 1,\ j = 0,\ s \geq 1, \\ \{2,R\}, & i = 0,\ j \geq 1,\ s \geq 1, \\ \{0,R\}, & i = j = 0,\ s \geq 1, \\ \{R\}, & s = 0. \end{cases}$$

Taking action $k \in \{1,2\}$ corresponds to assigning the server to queue k, action 0 corresponds to idling the server, and action R corresponds to maintaining (i.e., replacing or repairing) the server. Note that we are only considering policies that do not idle the server when there are jobs in the system, i.e., *non-idling policies*. Let Π denote the set of all non-idling policies that are *non-anticipating*, i.e., that do not make use of future arrival times, service completion times, and deterioration times. Of particular interest will be the policies in Π that are *stationary*, which are identified with functions $f : \mathbb{X} \to \mathbb{A}$ where $f(i,j,s) \in A(i,j,s)$ for all $(i,j,s) \in \mathbb{X}$; under such a policy, the action $f(i,j,s)$ is selected whenever the current state is (i,j,s). Let $\mathbb{F} \subset \Pi$ denote the set of all stationary policies.

To define the optimality criteria under consideration, fix $\pi \in \Pi$ and $t \in [0,\infty)$. Let $N^\pi(t)$ denote the number of decision epochs that occur during $[0,t]$ under π, and for $n = 0,1,\dots$ let t_n^π denote the time of the n^{th} decision epoch under π, where $t_0^\pi := 0$. In addition, let $S^\pi(t)$ denote the state of the server at time t under the policy π, let $U^\pi(t)$ denote the action selected at time t under π, and for $k = 1,2$ let $Q_k^\pi(t)$ denote the number of jobs in queue k at time t under π.

Consider a discount rate $\theta \geq 0$, a time horizon $T \in [0,\infty)$, and a policy $\pi \in \Pi$. The expected total θ-discounted cost incurred up to time T under the policy π, when the initial state is $(i,j,s) \in \mathbb{X}$, is

$$g_{T,\theta}^\pi(i,j,s) := \mathbb{E}_{(i,j,s)} \left\{ \sum_{n=0}^{N^\pi(T)-1} e^{-\theta t_n^\pi} K(S^\pi(t_n^\pi)) \mathbf{1}\{U^\pi(t_n^\pi) = R\} + \int_0^T e^{-\theta t} [h_1 Q_1^\pi(t) + h_2 Q_2^\pi(t)]\ dt \right\},$$

where $\mathbf{1}\{\cdot\}$ denotes the indicator function. A policy π_* is *θ-optimal for the finite horizon T* if for every $(i,j,s) \in \mathbb{X}$, $g_{T,\theta}^{\pi_*}(i,j,s) \leq g_{T,\theta}^\pi(i,j,s)$ for all $\pi \in \Pi$.

The expected total θ-discounted cost incurred by the N^{th} decision epoch under $\pi \in \Pi$, starting from the initial state $(i,j,s) \in \mathbb{X}$, is

$$v_{N,\theta}^\pi(i,j,s) := \mathbb{E}_{(i,j,s)} \left\{ \sum_{n=0}^{N-1} \left[e^{-\theta t_n^\pi} K(S^\pi(t_n^\pi)) \mathbf{1}\{U^\pi(t_n^\pi) = R\} + \int_{t_n^\pi}^{t_{n+1}^\pi} e^{-\theta t} [h_1 Q_1^\pi(t) + h_2 Q_2^\pi(t)]\ dt \right] \right\}$$

For a discount rate $\theta > 0$, the infinite-horizon expected total θ-discounted cost incurred under the policy $\pi \in \Pi$, starting from the initial state $(i,j,s) \in \mathbb{X}$, is

$$v_\theta^\pi(i,j,s) := \lim_{N \to \infty} v_{N,\theta}^\pi(i,j,s).$$

A policy $\pi_* \in \Pi$ is *θ-optimal* if $v_\theta^{\pi_*}(i,j,s) = \inf_{\pi \in \Pi} v_\theta^\pi(i,j,s) =: v_\theta(i,j,s)$ for all $(i,j,s) \in \mathbb{X}$. In addition, the long-run expected average cost per unit time incurred under π, when the initial state is $(i,j,s) \in \mathbb{X}$, is

$$\rho^\pi(i,j,s) := \limsup_{T \to \infty} \frac{g_{T,0}^\pi(i,j,s)}{T}.$$

A policy $\pi_* \in \Pi$ is *average-optimal* if $\rho^{\pi_*}(i,j,s) = \inf_{\pi \in \Pi} \rho^\pi(i,j,s) =: \rho(i,j,s)$ for all $(i,j,s) \in \mathbb{X}$.

When the server state does not change over time, it is well-known that it is optimal to schedule the server according to the so-called μh-rule, under which one prioritizes any queue i for which $\mu_i h_i$ is maximized; see [1], [5]. Assumption 1 below will be used to ensure that it is optimal to schedule the server according to an analogous priority rule when the server state randomly changes over time. In the sequel, we will assume without loss of generality that the classes are numbered so that there is a server state $\ell \in \{1,\dots,B\}$ satisfying $\mu_1^\ell h_1 \geq \mu_2^\ell h_2$.

Assumption 1 (Constant Ratio): For $s = 1,\dots,B$, the equality $\mu_1^{s-1}\mu_2^s = \mu_2^{s-1}\mu_1^s$ holds.

Assumption 1 states that the *ratio* of the service rates for class 1 and class 2 jobs remains constant as the server

changes state. Observe that, if Assumption 1 holds, then $\mu_1^s h_1 \geq \mu_2^s h_2$ for $s = 0, 1, \ldots, B$. In other words, under Assumption 1 the rate at which holding costs are reduced is better when queue 1 is served, uniformly in the server states. While it is tempting to conjecture that prioritizing queue 1 when $\mu_1^s h_1 \geq \mu_2^s h_2$ for $s = 0, 1, \ldots, B$ is always optimal, the following example shows that this conjecture is false.

Example 1: The server state can be 0, 1, or 2. When the server is in state 2, the respective service rates for class 1 and 2 are $\mu_1^2 = 10$ and $\mu_2^2 = 2$, while the corresponding service rates are $\mu_1^1 = 10$ and $\mu_2^1 = 1$ when the server state is 1. Suppose that the sojourn times of the server in states 1 and 2 are independent and exponentially distributed with the same rate. Moreover, maintenance occurs instantaneously; in other words, whenever the server reaches state 0 it immediately jumps to state 2. Finally, letting $h_1 = h_2 = 1$, observe that $\mu_1^1 h_1 = 10 > 1 = \mu_2^1 h_2$ and $\mu_1^2 h_1 = 10 > 2 = \mu_2^2 h_2$. Moreover, the holding cost corresponds to the number of jobs in the system.

Suppose class 1 jobs are given priority, and that $\lambda_1 = 5$. Recalling that $\mu_1^1 = \mu_1^2 = 10$, it follows that half of the server's capacity will be occupied with class 1 jobs. Since the server spends half of its time on average in state 1 and the other half in state 2, the system is stable (i.e., the expected average queue lengths do not grow without bound over time) if and only if $\lambda_2 < 0.5(0.5 \cdot 2 + 0.5 \cdot 1) = 0.75$.

To complete the example, observe that if class 1 jobs are prioritized in server state 1 and class 2 jobs are prioritized in server state 2, then the system is stable if $\lambda_2 < 1$. This implies that if $\lambda_2 \in (0.75, 1)$, then the long-run average number of jobs in the system under this state-dependent prioritization is finite, while the long-run average number of class 2 jobs in the system is infinite if class 1 is always given priority. Since the holding costs accrue linearly at the constant rate $h_1 = h_2 = 1$, it follows that prioritizing class 1 jobs in every state is strictly suboptimal.

On the other hand, under Assumption 1 it is possible to adapt the interchange argument used in [5] to our setting. To this end, a key part of Assumption 1 is statement 2), which states that the fraction by which the service rate changes between server states is the same for both job classes. In addition, in Sections IV-C and IV-D a necessary and sufficient condition for stability under Assumption 1, stated as Assumption 4 below, will be used for results on the structure of average-optimal maintenance decisions. It is possible to remove the dependence of these results on Assumption 1, by replacing Assumption 4 with a sufficient, but not necessary, stability condition; see Assumption 5 in Section IV.

III. SCHEDULING UNDER A FIXED MAINTENANCE POLICY

In this section, we consider the structure of optimal server scheduling decisions under a fixed maintenance policy that is independent of the queue lengths. For example, the server could be maintained whenever its state is at or below a certain threshold, or a calendar-based policy could be followed (see e.g., [2]). Theorem 1 and Corollary 1 in this section are also relevant to the case of joint scheduling and preventive maintenance; see Section IV-A.

Assumption 2 (Fixed Maintenance Policy): The decision-maker follows a fixed policy for deciding whether or not to begin maintaining the server, i.e, to take the action R, that is independent of the queue lengths.

If Assumption 2 holds, then if server maintenance is not initiated at a given decision epoch and both queues are nonempty, it is up to the decision-maker to decide which queue to serve. To state the results in this section, a policy is said to *prioritize* queue $k \in \{1, 2\}$ if, whenever one of the queues is to be served, queue k is served if it is nonempty.

Theorem 1 (Finite-Horizon): Suppose Assumptions 1 and 2 hold. Let π_* denote the policy that prioritizes queue 1. Then for any $\theta \in [0, \infty)$ and $T \in [0, \infty)$, the policy $\pi_* \in \Pi$ is θ-optimal for the finite horizon T.

Theorem 1 can be proved using an interchange argument analogous to the one used in [5, Proof of Theorem 2.1]. Moreover, the optimality under Assumption 1 of the policy that prioritizes queue 1 for any finite horizon implies the following corollary.

Corollary 1 (Infinite-Horizon): Suppose Assumptions 1 and 2 hold. Let π_* denote the policy that prioritizes queue 1. Then π_* is both θ-optimal for any $\theta \in (0, \infty)$, and average-optimal.

IV. JOINT SCHEDULING AND PREVENTIVE MAINTENANCE

We now consider the structure of optimal policies when both scheduling and preventive maintenance decisions may be made. For this we will assume the following memory-lessness condition, under which the decision problem can be formulated as a semi-Markov decision process (SMDP).

Assumption 3 (SMDP):

1) Jobs arrive to queues 1 and 2 according to indpendent Poisson processes with rates λ_1 and λ_2, respectively.
2) If the current state of the server is $s \in \{1, \ldots, B\}$, then the time D_s until it transitions to state $s-1$ is exponentially distributed with rate m_s.
3) The time D_0 required for the server to be repaired is independent of the arrivals, service times, and deterioration times $D_1, \ldots, D_B$.

Under Assumption 3, we will also assume without loss of generality that

$$\Psi := \sum_{k=1}^{2} \left(\lambda_k + \sum_{s=1}^{B} \mu_k^s \right) + \sum_{s=1}^{B} m_s = 1$$

Namely, multiplying all transition rates not associated with the maintenance action R by a constant does not affect the optimality of a policy.

We will consider the structure of optimal policies under both the discounted-cost and average-cost criterion. For the latter, we employ stability conditions to ensure that the underlying SMDP satisfies the conditions proposed in Sennott [8]. According to [8], the latter conditions in turn

guarantee the existence of stationary average-cost optimal policies via the existence of a suitable solution to a set of average-cost optimality inequalities. For the stability conditions, let $1/m_0$ denote the expected time required for the server to be repaired.

Assumption 4 (Stability Under Assumption 1):

1) Assumption 1 holds.
2) Let $r = \mu_1^B/\mu_2^B$. There is a server state $s^* \in \{1,\dots,B\}$ such that

$$\lambda_1 + r\lambda_2 < \left(\frac{1}{m_0} + \sum_{s=s^*}^{B} \frac{1}{m_s}\right)^{-1} \sum_{s=s^*}^{B} \frac{\mu_1^s}{m_s}.$$

The intuition that this assumption implies stability is as follows. Due to Assumption 1, $\mu_1^s/\mu_2^s = r$ for all s. If the machine is repaired in state s^*, the average service rate over a cycle from repair to repair is

$$\bar{\mu}_i^{s^*} = \left(\frac{1}{m_0} + \sum_{s=s^*}^{B} \frac{1}{m_s}\right)^{-1} \sum_{s=s^*}^{B} \frac{\mu_i^s}{m_s}.$$

Stability follows from the sufficient condition

$$\frac{\lambda_1}{\bar{\mu}_1^{s*}} + \frac{\lambda_2}{\bar{\mu}_2^{s*}} < 1$$

and the definition of r.

The following sufficient condition for stability, which does not rely on Assumption 1, also suffices to imply that the underlying SMDP satisfies the conditions in [8] that imply the existence of a stationary average-cost optimal policy.

Assumption 5: There is a server state $s^* \in \{1,\dots,B\}$ such that

$$\frac{\lambda_1}{\sum_{s=s^*}^{B} \frac{\mu_1^s}{m_s}} + \frac{\lambda_2}{\sum_{s=s^*}^{B} \frac{\mu_2^s}{m_s}} < \left(\frac{1}{m_0} + \sum_{s=s^*}^{B} \frac{1}{m_s}\right)^{-1}.$$

A. Structure of optimal scheduling decisions

We first consider the structure of optimal scheduling decisions. The main result, stated as Theorem 2 below, provides conditions under which it suffices to consider policies that prioritize queue 1. In particular, under the hypotheses of Theorem 2, the control problem reduces to that of finding an optimal maintenance policy.

Theorem 2 (Optimal Scheduling): If Assumptions 1 and 3 hold, and the decision-maker is restricted to maintenance policies that are independent of the queue lengths, then for $\theta > 0$ there exists a θ-optimal policy with the following property: If the decision is made to serve one of the queues, the server is assigned to queue 1 if it is nonempty. Moreover, if Assumption 4 also holds, then there exists an average-optimal policy with this property.

Theorem 2 can be proved by using Theorem 1 to show that, for any optimal policy, there exists a policy that prioritizes queue 1 and performs at least as well.

B. Structure of optimal maintenance decisions

The remainder of Section IV concerns the structure of optimal maintenance decisions. In particular, we provide conditions under which it is optimal to maintain the server according to a threshold policy. Under such a policy, maintenance is performed whenever the server state reaches a certain threshold state.

We will consider two sets of assumptions on the maintenance times and costs, each of which leads to a different interpretation of the maintenance action. In Section IV-C, we consider a situation where maintenance can be interpreted as repairing the server, which requires a random amount of time. In Section IV-D, the maintenance action can be viewed as corresponding to replacing the server with a new one.

The results in Sections IV-C and IV-D rely on a monotonicity property of the discounted-cost value function v_θ (Proposition 1), which states that one can do better in states with fewer jobs and higher service rates. The validity of this property in turn depends on Assumption 6 below, which states that server states with smaller indices are worse in terms of service rate.

Assumption 6 (Deterioration): The service rates are nondecreasing in the server state, i.e., $\mu_k^0 = 0 < \mu_k^1 \le \cdots \le \mu_k^B < \infty$ for $k = 1,2$.

Proposition 1 below, which is useful for proving the structural results in Sections IV-C and IV-D, can be proved via a sample path argument similar to the one used in [4, Proof of Proposition 3.3].

Proposition 1 (Monotonicity): Suppose Assumptions 3 and 6 hold. Let $\theta > 0$, and consider $(i,j,s),(i',j',s') \in \mathbb{X}$ where $i \le i'$, $j \le j'$ and $s \ge s'$. Then $v_\theta(i,j,s) \le v_\theta(i',j',s')$.

C. Repair model

We first present results for the case where action R corresponds to initiating the repair of the server, which takes a random amount of time and incurs a cost that does not depend on the state from which the repair was initiated.

Assumption 7 (Repair Model):

1) The mean repair time $\mathbb{E}[D_0]$ is positive and finite.
2) The fixed repair cost is a constant $K \in (0,\infty)$.

A stationary policy $f \in \mathbb{F}$ is *monotone in the server state* s if, for every fixed number of jobs in the two queues, the repair action R is taken under f for all sufficiently low (i.e., "bad") server states. In other words, $f \in \mathbb{F}$ is monotone in the server state s if $f(i,j,s+1) = R$ implies $f(i,j,s) = R$, for all $(i,j) \in \mathbb{N} \times \mathbb{N}$ and $s \in \{0,1,\dots,B-1\}$. Theorem 3 below provides conditions under which every optimal policy under the discounted-cost criterion is monotone in the server state. According to Theorem 4, the same result holds under the average-cost criterion if the stability condition stated as Assumption 4 holds.

Theorem 3 (Discounted Costs): Suppose Assumptions 3, 6, and 7 hold. Then for $\theta > 0$ there exists a θ-optimal stationary policy, and every such policy is monotone in the server state s.

Theorem 4 (Average Costs): Suppose Assumptions 3, 6, and 7 hold. If Assumption 4 or 5 also holds, then there exists an average-optimal stationary policy that is monotone in the server state s.

Theorem 3 can be proved by analyzing the discounted-cost optimality equation (see e.g., [8, Theorem 1]), and using the

monotonicity property of v_θ stated in Proposition 1. Similarly, Theorem 4 follows from the existence of a solution to the average-cost optimality inequality (see [8, Theorem 2]), and the monotonicity of this solution that is implied by Proposition 1.

D. Replacement model

In this section, taking action R can be interpreted as making the decision to replace the server with a new one. In particular, we will assume that this replacement occurs instantaneously, and that the costs $K(s)$ incurred by this action when the current server state is s satisfy certain convexity and monotonicity properties.

Assumption 8 (Replacement Model):

1) The maintenance action R represents an instantaneous *replacement*, i.e., $D_0 \equiv 0$.
2) The replacement cost function $K : \{0,1,\ldots,B\} \to (0,\infty)$ is *convex*, i.e.,
$$K(s+1)+K(s-1) \geq 2K(s), \quad s = 1,\ldots,B-1.$$
3) Given $\theta \geq 0$, one of the following two conditions holds for $s = 2,\ldots,B-1$.
   a) $K(s) \leq K(s-1)$ and $m_{s-1} \geq m_s + \theta$.
   b) $K(s) \geq K(s-1)$ and $m_{s-1} \leq m_s + \theta$.

The above assumptions are the same as those considered in [4, Proposition 4.10] and an analogous result, stated as Theorems 5 and 6 below, can be proved via a value-iteration argument.

Theorem 5 (Discounted Costs): Suppose Assumptions 3, 6, and 8 hold. Then for $\theta > 0$ there exists a θ-optimal stationary policy, and every such policy is monotone in the server state s.

Theorem 6 (Average Costs): Suppose Assumptions 3, 6, 4, and 8 hold. Then there exists an average-optimal stationary policy that is monotone in the server state s.

Theorems 5 and 6, along with Theorems 3 and 4 for the repair model, provide conditions under which it suffices to restrict attention to policies where, for every $i, j \in \mathbb{N}$, there is a threshold $\phi(i,j) \in \{0,1,\ldots,B\}$ such that action R is taken if and only if the current server state $s \leq \phi(i,j)$.

V. EMPIRICAL PERFORMANCE

In Sections III and IV, sufficient conditions were provided for the optimality of performing scheduling according to a priority policy. This priority policy satisfies the following: If both queues are nonempty, and the current state is (i,j,s), then any queue i maximizing $\mu_i^s h_i$ is served. In this section, we will numerically evaluate the performance that results from following this scheduling rule. In particular, we will compare two versions of this scheduling rule (described in Section V-A.1) to a naïve first-come first-served policy described in Section V-A.2, and to an optimal policy obtained via dynamic programming that is described in Section V-A.3.

We consider the case of a deteriorating server, where the server can be in one of three states: "failed", "worn", or "new". The arrival processes will be assumed to be Poisson, and the service and deterioration times will be assumed to be exponential. However, three key assumptions, on which our results in the preceding sections depend, will be relaxed. First, statement 2) of Assumption 1, under which the service rates always change by a constant fraction, does not hold. Second, the rate at which the server deteriorates depends on both the current server state s and the class i of the job currently in service; this rate will be denoted by α_i^s. Finally, the rate at which the server returns to the best state varies according to the state from which the corresponding maintenance action was initiated. These conditions are more representative of the problem in semiconductor manufacturing that motivated our model, and a possible future research direction involves extending the theoretical results presented in Sections III and IV to this setting.

A. Policies

1) State-Dependent μh-Rule: In Section IV, conditions were provided under which there exists an optimal policy with the following property: If the decision is made to serve one of the queues while both queues are nonempty and the state of the server is s, the server should be assigned to queue 1 if $\mu_1^s h_1 \geq \mu_2^s h_2$ and to queue 2 otherwise. In Section V-C below, we will consider the empirical performance of two policies that have this structure. One of them, denoted by μh−NPM, serves according to the aforementioned state-dependent μh-rule without performing preventive maintenance. The other, denoted by μh−PM, is obtained by solving the dynamic programming formulation of the problem where it is assumed that the server is scheduled according to the (state-dependent) μh-rule.

2) First-Come, First-Served: To indicate the value of taking queue-dependent holding costs and the state of the server into account, we will compare μh−NPM and μh−PM to a simple first-come first-served policy. Under this benchmark policy, denoted by FCFS, the arriving customers are simply served in the order in which they arrive.

3) Optimal Policy via Dynamic Programming: Finally, we will compare the performance of μh−NPM and μh−PM with two types of policies obtained via dynamic programming. One of these, denoted by DP−NPM, is obtained by solving the dynamic programming formulation of the problem where preemptive maintenance is not permitted. The other, denoted by DP−PM, is an optimal policy for the dynamic programming formulation of the original problem. The latter serves as a benchmark that will suggest how far the other policies are from optimality.

B. Model Parameters

The policies described in Section V-A are evaluated by varying the parameters and simulating the following version of the model described in Section II. Arrivals to queues 1 and 2 occur according to independent Poisson processes with rates λ_1 and λ_2, respectively. The service requirement for every arrival is exponential with rate 1, and incur holding costs at a rate of $h_1 = h_2 = 1$. The set of possible server states is $\{0,1,2\}$, and the server state deteriorates according to a continuous-time Markov chain. The service rates and

Server State s	μ_1^s	μ_2^s	α_1^s	α_2^s
2	6	6	1/15	1/0
1	4.8	3	1/12	1/8

TABLE I

SERVICE RATES AND DETERIORATION RATES

λ_1	λ_2	μh−NPM	μh−PM	FCFS	DP−NPM
1.65	1.65	2.45 (0.045)	**0.99 (0.028)**	3.50 (0.040)	3.12 (0.040)
1.65	1.82	2.84 (0.073)	**1.07 (0.024)**	4.03 (0.040)	3.38 (0.049)
1.65	1.99	3.13 (0.036)	**1.07 (0.027)**	5.08 (0.055)	3.98 (0.034)
1.99	1.65	2.92 (0.039)	**1.00 (0.031)**	4.36 (0.046)	3.34 (0.042)
1.99	1.82	3.46 (0.048)	**1.02 (0.040)**	5.17 (0.043)	3.84 (0.044)
1.99	1.99	4.12 (0.056)	**1.09 (0.032)**	6.99 (0.080)	4.35 (0.029)

TABLE II

POLICY PERFORMANCE: THE REPORTED VALUES FOR EACH POLICY ARE THE RATIO OF ITS AVERAGE COST WITH THAT OF THE OPTIMAL POLICY, AND THE CORRESPONDING COEFFICIENT OF VARIATION IN PARENTHESES.

deterioration rates, which depend on the server state and the class of job being served, are given in Table I. When the maintenance action is taken while the current server state is $s = 1$, the time needed for the server to return to state $B = 2$ is exponentially distributed with rate $2/3$. On the other hand, the corresponding rate for $s = 0$ is $1/3$.

C. Performance

We consider the average-cost criterion. A summary of the performance of the policies described in Section V-A, relative to the optimal policy DP−PM and over a range of arrival rates, is given in Table II. The policy μh−PM, which was obtained by fixing the scheduling to follow the μh-rule as described in Section V-A.1 and computing the optimal maintenance decisions, achieved performance similar to the optimal policy DP−PM over this range, with some deterioration in performance under higher arrival rates. Moreover, there are significant distinctions in performance between μh−NPM and DP−NPM, compared to their counterparts that allow for preventive maintenance. This suggests the importance of allowing for preemptive repairs/maintenance of a deteriorating server.

VI. CONCLUSION AND FUTURE WORK

In this paper, we have provided conditions that imply the optimality of a natural priority rule for scheduling a possibly deteriorating server in the context of a 2-class queue. In particular, when Assumption 1 holds and a fixed maintenance policy is used, Theorem 1 implies that such a scheduling policy is optimal under general arrival and server-state processes. When an optimal policy for the joint scheduling and maintenance problem exists, e.g., under the hypotheses of Theorem 2, under Assumption 1 it follows that at least one such policy schedules the server according to the aforementioned priority policy. In addition, according to Theorems 3,4 and 5,6, a natural monotonicity property of optimal maintenance decisions for the joint scheduling and maintenance problem holds under additional assumptions on the server state process.

There are a number of promising directions for future research. For instance, Statement 2) of Assumption 1 is fairly restrictive, and has been observed to be violated in the context of our motivating problem in semiconductor manufacturing. On the other hand, we have found empirical evidence that following the priority policy suggested by our results can still achieve optimal or nearly-optimal performance when Assumption 1 is violated. This suggests two possible research directions. One involves understanding the extent to which the structural results in Sections III and IV-A hold when Assumption 1 is weakened. A second, closely related direction is to investigate whether it is possible to bound the performance of the priority policy alluded to above when this policy is not necessarily optimal. Of course, it would also be of interest to extend our structural results to the case of more than two job classes and more than one server, and to investigate the efficacy of priority scheduling in these settings.

REFERENCES

[1] C. Buyukkoc, P. Varaiya, and J. Walrand. The $c\mu$ rule revisited. *Advances in Applied Probability*, 17(1): 237–238, 1985.

[2] Y. Cai, J. J. Hasenbein, E. Kutanoglu, and M. Liao. Single-machine multiple-recipe predictive maintenance. *Probability in the Engineering and Informational Sciences*, 27(2): 209-235, 2013.

[3] M. Celen and D. Djurdjanovic. Integrated maintenance decision-making and product sequencing in flexible manufacturing systems. *Journal of Manufacturing Science and Engineering*, 137(4): 041006-041006–15, 2015.

[4] D. L. Kaufman and M. E. Lewis. Machine maintenance with workload considerations. *Naval Research Logistics*, 54(7):750–766, 2007.

[5] P. Nain. Interchange arguments for classical scheduling problems in queues. *Systems & Control Letters*, 12(2):177–184, 1989.

[6] J. G. Shanthikumar, S. Ding, and M. T. Zhang. Queueing theory for semiconductor manufacturing systems: a survey and open problems. *IEEE Transactions on Automation Science and Engineering*, 4(4): 513–522, 2007.

[7] T. W. Sloan and J. G. Shanthikumar. Combined production and maintenance scheduling for a multiple-product, single-machine production system. *Production and Operations Management*, 9(4): 379–399, 2000.

[8] L. I. Sennott. Average cost semi-Markov decision processes and the control of queueing systems. *Probability in the Engineering and Informational Sciences*, 3(2):247–272, 1989.

[9] X. Yao, X. Xie, M. C. Fu, S. I. Marcus. Optimal joint preventive maintenance and production policies. *Naval Research Logistics*, 52(7): 668-681.