authenticated version is available online at: https://doi.org/10.1007/s11750-019-00509-1

A 2-class maintenance model with dynamic server behavior

Kevin Granville[1*], Steve Drekic[1]

[1]Department of Statistics and Actuarial Science, University of Waterloo,
200 University Avenue West, Waterloo, ON, N2L 3G1, Canada
[*]To whom correspondence should be addressed; E-mail: kgranville@uwaterloo.ca

Abstract

We analyze a 2-class maintenance system within a single-server polling model framework. There are $C + f$ machines in the system, where C is the cap on the number of machines that can be turned on simultaneously (and hence, be at risk of failure), and the excess f machines comprise a maintenance float which can be used to replace machines that are taken down for repair. The server's behavior is dynamic, capable of switching queues upon a machine failure or service completion depending on both queue lengths. This generalized server behavior permits the analysis of several classic service policies, including preemptive resume priority, non-preemptive priority, and exhaustive. More complicated polices can also be considered, such as threshold-based ones and a version of the Bernoulli service rule. The system is modelled as a level-dependent quasi-birth-and-death process and matrix analytic methods are used to find the steady-state joint queue length distribution, as well as the distribution for the sojourn time of a broken machine. An upper bound on the expected number of working machines as a function of C is derived, and Little's Law is used to find the relationship between the expected number of working machines and the expected sojourn time of a failed machine when $f = 0$ or $f \geq 1$. Several numerical examples are presented, including how one might optimize an objective function depending on the mean number of working machines, with penalty costs attributed to increasing C or f.

Keywords: Maintenance model · Polling model · Dynamic server · Threshold policy · Switch-in times · Quasi-birth-and-death process

1 Introduction

In this paper, we investigate a closed queueing network tracking a finite population of machines which alternate between being functional or broken. Broken machines are separated into one of two classes depending on their type of failure and service requirement, and we allow for a single mechanic such that only a single machine may receive repairs at any one time, so we elect to represent the system as a polling model. A standard single-server polling model describes a system containing multiple queues which are visited by the server according to a service policy which determines the order of the visits and the limit on how many customers they can serve per visit to a particular queue. Two of the earliest papers analyzing polling models described maintenance problems concerning a patrolling repairman (Mack et al. 1957, Mack 1957). As we are allowing for the possibility to experience delays when the mechanic switches between repairing either class of machine failures, this framework is natural. For an overview on the wider study of polling models, one may refer to the surveys of Boon (2011), Boon et al. (2011), Levy and Sidi (1990), Takagi (1988), and Vishnevskii and Semenova (2006).

Our model assumes a given capacity limiting the number of machines that may be in use at one time (and hence at risk of failure) with an optional inventory of spare machines called a maintenance float which may be immediately turned on to replace failed machines. Some other works concerned with inventories of spare or reserve machines include Buyukkramikli et al. (2015), Gross et al. (1983), Kim and Dshalalow (2003), Liang et al. (2013), and Madu (1988). In particular, both the works of Gross et al. (1983) and Madu (1988) concern closed queueing networks of machines that can suffer two levels of failures having different service requirements; however in Gross et al. (1983), there are repair stations allowing multiple dedicated servers for each failure type while in Madu (1988) there is only a single server in each station and only a single machine may be put to use at a time. Moreover, every distributional assumption is exponential in these two works and a broken machine may require service at both stations prior to being returned to functionality. Some other examples of closed queueing maintenance networks are papers by Abboud (1996), Iravani et al. (2007), Lin et al. (1994), and Righter (2002).

In addition to the inclusion of a maintenance float, this work expands that of Granville and Drekic (2018) by generalizing the server's allowed behaviors. In particular, we allow for the probability of the server switching to the opposite queue at decision epochs after repair completions or machine failure instants to depend on both queue lengths, similar to Iravani et al. (2007) and Liang et al. (2013) within the context of using Markov decision processes to find the optimal server behavior. This dynamic behavior contains as special cases the exhaustive, preemptive resume priority, and non-preemptive priority service policies considered in Granville and Drekic (2018), as well as the (a, b) threshold and smart Bernoulli policies which we define later in Section 3.1.1. Standard threshold policies in a two queue polling model that assign priority to a class of customers once their queue length reaches or exceeds a certain value (e.g., Avram and Gómez-Corral 2006, Boxma et al. 1995, Lee and Sengupta 1993) can allow for a more precise and optimal application of priority than

applying a static priority policy that always favours one queue over the other, and our (a, b) threshold aims to further improve this precision. In our analysis, we employ matrix analytic methods, which can be used to analyze 2-class threshold models as in the recent papers by Avrachenkov et al. (2016) and Perel and Yechiali (2017).

Of course, a Bernoulli service policy (first introduced in the context of a $GI/G/1$ vacation model by Keilson and Servi 1986), which generalizes the exhaustive and 1-limited service policies, can also be used to optimize a polling model (e.g., Blanc and van der Mei 1995). Specifically, a server following a Bernoulli policy serves at least one customer per visit to a queue and assigns varying importance to each queue by way of a class-dependent probability (which may be varied) of the server initiating another service after a completion (should their queue be non-empty) rather than switching away. In Section 5.3, we argue for the optimality of setting one of our smart Bernoulli probabilities to 1 as in Blanc and van der Mei (1995), reducing to a two queue polling model with exhaustive service at one queue and smart Bernoulli at the other. For an example of a 2-class polling model with exhaustive and the standard Bernoulli policy, one can refer to Weststrate and van der Mei (1994). For examples of Bernoulli service in a polling model with a general number of queues, with or without switchover times, see Blanc (1990, 1991).

In the next section, we outline the features and distributional assumptions of the maintenance system. This is followed by the formal definition of the state space and the server's decision epoch switching probabilities (including the specification of several special cases), the derivation of the steady-state probabilities treating the system as a level-dependent quasi-birth-and-death (QBD) process, as well as the derivation of the continuous phase-type distributed sojourn time distribution in Section 3. We examine several results about the expected number of working machines in Section 4, concerning the impact of increasing the number of machines in the system capacity as well as the connection to the mean sojourn time of a failed machine. Finally, we present a series of numerical examples in Section 5, followed by some concluding remarks in Section 6.

2 The Maintenance System

We consider a maintenance system of $C + f$ identical machines, where $C \in \mathbb{Z}^+$ is the system's capacity, or the cap on how many machines may be in use at once (and hence at risk of failure), and $f \in \mathbb{N}$ denotes the number of machines in the maintenance float. The float provides an extra inventory of functional machines that replace machines that are taken down for repair after suffering a failure. It is assumed that a machine is not at risk of failure while turned off and stored in the float, and that they can instantaneously be put to use and turned on when needed. Following a machine repair, it is instantly turned on if the number of working machines immediately prior to the repair completion was less than C; otherwise, it is stored in the maintenance float.

The system is modelled as a 2-class polling model attended to by a lone mechanic (or server), where each class represents a grouping of one or more types of failure, and the service time distributions for each type of failure are allowed to be different. Let α_i, $i = 1, 2$, be

the total exponential class-i failure rate, such that each machine, when turned on, has an effective failure rate of $\alpha = \alpha_1 + \alpha_2$. It is assumed that the failure times of the machines are independent, machines fail individually, and a machine may only suffer one type of failure at a time. This last assumption may be worked around if the types of failure are within the same class by defining a combination of failure types as a new type of failure (to be included in the same class).

Upon experiencing a class-i failure (and henceforth being referred to as a *class-i machine* until it is repaired), a class-i machine waits in the i^{th} queue to receive service on a first-come-first-served basis with respect to other class-i machines in the same queue. To contrast the two classes of failures, we denote functional machines (either in use or stored in the float) as being of class 0. When every machine is class 0, rather than waiting at class 1 or class 2, the mechanic moves to a neutral third location, similarly named class 0.

It is assumed that class-i service times are strictly positive and follow a continuous phase-type distribution with representation $\text{PH}(\underline{\beta}_i, B_i)$ of order b_i (e.g., see He 2014, p. 10). This is inherently a more restrictive assumption than generally distributed service times, although it is possible to approximate a (non-negative) non-phase-type distribution by fitting a phase-type one (most notably via the classic EM algorithm outlined by Asmussen et al. 1996). However, they have a difficult time approximating some distributions well (particularly heavy-tailed ones), and increasing the number of phases to improve the fit can introduce computational issues due to the impact on the size of the state space of the model.

Fortunately, phase-type distributions do have many appealing features. Since phase-type distributions are closed under finite mixtures, it is straightforward to construct the underlying class-i service time distribution from the individual continuous phase-type distributions corresponding to each type of failure within the same class. Depending on the assigned behavior of the mechanic, it may be possible for a service time to be interrupted. In these cases, the service progress is not lost as the service phase is tracked to allow the mechanic to resume service where it left off, after eventually returning to that queue. Each service time is assumed to be independent of other services, as well as machine failure times.

Similarly, the time it takes the mechanic to "switch" from class j to class i (henceforth referred to as a *class-i switch-in*) is assumed to follow a continuous phase-type distribution with representation $\text{PH}(\underline{\gamma}_{ji}, S_i)$ of order s_i, where the rate matrix S_i depends only on the destination class, while the initial probability row vector $\underline{\gamma}_{ji}$ may also depend on the departure class. Switch-ins are also assumed to be independent of other switch-ins, as well as machine service and failure times. A switch-in having positive duration may, for example, represent any combination of the times necessary for the mechanic to change their instruments, retrieve spare parts, or physically relocate themselves to a different queue. If the time required to complete these tasks not directly related to serving an individual machine are insignificant, then it may make sense to allow the switch-in times to be identically zero. We let $\gamma_{ji}^{[0]} = 1 - \underline{\gamma}_{ji}\underline{e}'$ denote the probability of a class-i switch-in (from class j) being equal to zero in duration, where $\underline{e}'$ represents an appropriately-dimensioned column vector of ones. Henceforth, the notation $'$ will represent matrix transpose (such that $\underline{e}$ is a row vector).

As the mechanic may be allowed to preempt a switch-in within this system (if, say, one

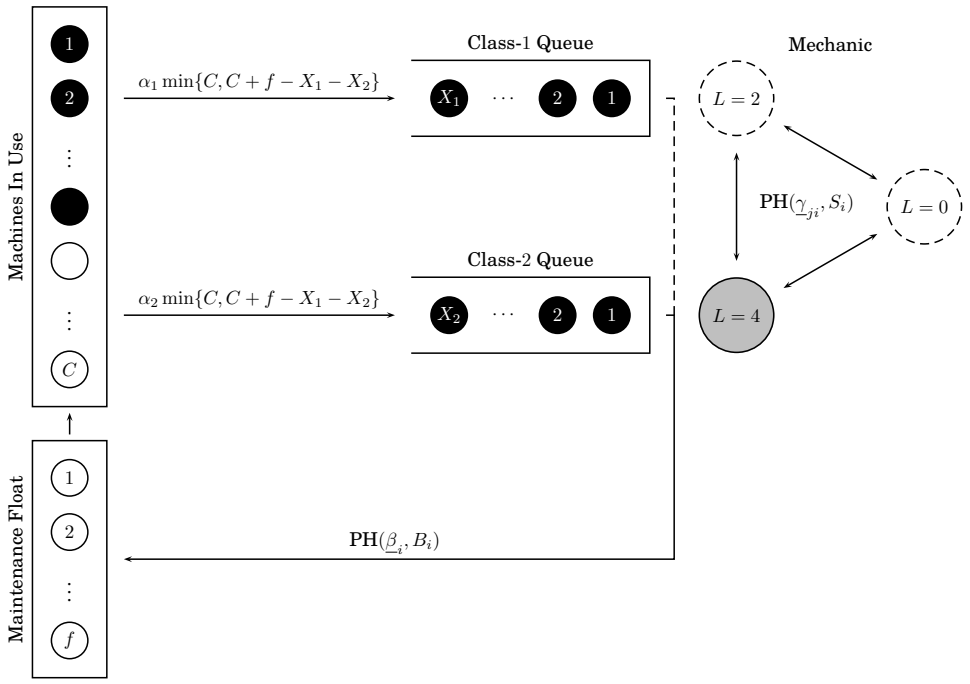

Figure 1: Depiction of the maintenance system with a maintenance float and the server at queue 2.

class has higher priority over the other at a given combination of queue lengths), we make the assumption that switching out of a class-i switch-in is the same as beginning a switch-in after the completion (or preemption) of a class-i service time. That is, for example, if class 1 has a higher priority than class 2 and the mechanic observes a class-1 failure while conducting a switch from class 0 to class 2, then they will start a new class-1 switch-in with initial probability vector $\underline{\gamma}_{21}$. We remark that a class-0 switch-in will always be interrupted if the mechanic observes a machine failure from either class.

We provide a depiction of the maintenance system as described above in Figure 1. Note that the notation X_1, X_2, and L are as they will be defined in Section 3.1, representing the first and second queue lengths, and the position of the server, respectively. Machines are represented by solid black circles, while slots that machines may take within class 0 (whether to be put in use or in the maintenance float) are represented by empty circles. Similarly, the larger solid grey circle and dashed empty circles represent current and potential locations where the server either works or idles, with the grey circle in this example implying that the mechanic is currently serving class-2 machines. As defined above, the distribution of the time between service completions is $\mathrm{PH}(\underline{\beta}_i, B_i)$ (in this example, we would have $i = 2$), and if the server switches between the three locations, the time to complete the switch has a $\mathrm{PH}(\underline{\gamma}_{ji}, S_i)$ distribution. Repaired machines are brought to the maintenance float, where they will automatically be put to use if there are any open slots for functional machines.

Figure 1 does assume that a float exists (i.e., $f \geq 1$), but we do in fact allow the choice of $f = 0$. In the $f = 0$ case, the diagram would change by way of having no float, and repaired machines would automatically be put to use.

A defining feature of polling models is the chosen service policy which dictates the server's behavior. In our model, we allow our mechanic be dynamic, whose decision to start a switch-in (i.e., the probability of deciding to switch) may depend on both queue lengths as well as what type of event is causing the server to make a decision, namely after a service completion (when the other queue has a positive length), or after observing an arrival to the opposite queue during a switch-in or a service. As these decision probabilities are state-dependent, we must first define the state space of the Markov chain describing this system before constructing the decision probability matrices.

3 Model Construction and Analysis

3.1 State Space and State-Dependent Decision Probabilities

In order to model this maintenance system as a continuous-time Markov chain (CTMC) without restricting the server's behavior, we must track six variables within our state space, $(X_1, X_2, L, Y, Y_1, Y_2)$. Here, $X_1 \in \{0, 1, \ldots, C + f\}$ is the length of the class-1 queue and is treated as the level of the process. Next, $X_2 \in \{0, 1, \ldots, C + f - X_1\}$ is the length of the class-2 queue. $L \in \{0, 1, 2, 3, 4, 5\}$ denotes the location of the server (0: idle at class 0; 1: switching into class 1; 2: serving class 1; 3: switching into class 2; 4: serving class 2; 5: switching into class 0). Y denotes the phase of a switch-in time or takes the value of 0 when the mechanic is either idle or repairing a machine, i.e.,

$$Y \in \Omega_Y(L) = \begin{cases} \{0\} & , \text{ if } L = 0, \\ \{1, 2, \ldots, s_1\} & , \text{ if } L = 1, \\ \{0\} & , \text{ if } L = 2, \\ \{1, 2, \ldots, s_2\} & , \text{ if } L = 3, \\ \{0\} & , \text{ if } L = 4, \\ \{1, 2, \ldots, s_0\} & , \text{ if } L = 5. \end{cases}$$

Lastly, Y_1 and Y_2 are the current phases of service of the class-1 and class-2 machines leading their respective queues. Y_i takes on a value of zero if the i^{th} queue is empty, so that

$$Y_i \in \Omega_{Y_i}(X_i) = \begin{cases} \{0\} & , \text{ if } X_i = 0, \\ \{1, 2, \ldots, b_i\} & , \text{ if } X_i \geq 1. \end{cases}$$

Note that this variable is initialized as soon as either X_1 or X_2 changes from 0 to 1 (after observing a class-1 or class-2 failure), which is in general not the same time when its service actually begins.

With the above notation in place, we can now define the decision probability matrices. As mentioned previously, we categorize decision epochs into one of three types, with the first type occurring after a service completion. Define $\mathcal{P}^{1S}_{i,j}$ as the probability of initiating a class-1 switch-in (from class 2) immediately after a class-2 service completion that reduces X_2 from $j+1$ to j, when $X_1 = i$. For ease of presentation (and storage), we let

$$\mathcal{P}^{1S} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ \vdots \\ C+f-3 \\ C+f-2 \end{array} \begin{array}{c} \begin{array}{ccccccc} 1 & 2 & 3 & \cdots & C+f-3 & C+f-2 \end{array} \\ \begin{pmatrix} \mathcal{P}^{1S}_{1,1} & \mathcal{P}^{1S}_{1,2} & \mathcal{P}^{1S}_{1,3} & \cdots & \mathcal{P}^{1S}_{1,C+f-3} & \mathcal{P}^{1S}_{1,C+f-2} \\ \mathcal{P}^{1S}_{2,1} & \mathcal{P}^{1S}_{2,2} & \mathcal{P}^{1S}_{2,3} & \cdots & \mathcal{P}^{1S}_{2,C+f-3} & 0 \\ \mathcal{P}^{1S}_{3,1} & \mathcal{P}^{1S}_{3,2} & \mathcal{P}^{1S}_{3,3} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ \mathcal{P}^{1S}_{C+f-3,1} & \mathcal{P}^{1S}_{C+f-3,2} & 0 & \cdots & 0 & 0 \\ \mathcal{P}^{1S}_{C+f-2,1} & 0 & 0 & \cdots & 0 & 0 \end{pmatrix} \end{array}.$$

Note that we do not need to define probabilities where $X_1 + X_2 = i + j = C + f$, since there must be at least one functional machine after a service completion, and we do not consider probabilities for $i = 0$ or $j = 0$, as we make the assumption that the mechanic will always choose to serve the class having a non-zero queue length should the other queue be empty. A corresponding matrix $\mathcal{P}^{2S}$ is also constructed in the same way, such that $\mathcal{P}^{2S}_{i,j}$ is the probability of switching to serve class 2 after a class-1 service completion which reduces X_1 from $i+1$ to i, when $X_2 = j$.

Next, we define $\mathcal{P}^{1P}_{i,j}$ ($\mathcal{P}^{2P}_{i,j}$) and $\mathcal{P}^{1N}_{i,j}$ ($\mathcal{P}^{2N}_{i,j}$) to be the probabilities of the server initiating a class-1 (class-2) switch-in after observing a class-1 (class-2) failure that results in $(X_1, X_2) = (i, j)$ after said failure when $L = 4$ ($L = 2$) or $L = 3$ ($L = 1$) immediately prior to the failure epoch, respectively. We distinguish these probabilities with a P or N to denote the fact that they represent switch-ins that are either *preemptive* or *non-preemptive* in nature, with respect to service times of the opposite class. We now let

$$\mathcal{P}^{1P} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ \vdots \\ C+f-3 \\ C+f-2 \\ C+f-1 \end{array} \begin{array}{c} \begin{array}{ccccccc} 1 & 2 & 3 & \cdots & C+f-3 & C+f-2 & C+f-1 \end{array} \\ \begin{pmatrix} \mathcal{P}^{1P}_{1,1} & \mathcal{P}^{1P}_{1,2} & \mathcal{P}^{1P}_{1,3} & \cdots & \mathcal{P}^{1P}_{1,C+f-3} & \mathcal{P}^{1P}_{1,C+f-2} & \mathcal{P}^{1P}_{1,C+f-1} \\ \mathcal{P}^{1P}_{2,1} & \mathcal{P}^{1P}_{2,2} & \mathcal{P}^{1P}_{2,3} & \cdots & \mathcal{P}^{1P}_{2,C+f-3} & \mathcal{P}^{1P}_{2,C+f-2} & 0 \\ \mathcal{P}^{1P}_{3,1} & \mathcal{P}^{1P}_{3,2} & \mathcal{P}^{1P}_{3,3} & \cdots & \mathcal{P}^{1P}_{3,C+f-3} & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ \mathcal{P}^{1P}_{C+f-3,1} & \mathcal{P}^{1P}_{C+f-3,2} & \mathcal{P}^{1P}_{C+f-3,3} & \cdots & 0 & 0 & 0 \\ \mathcal{P}^{1P}_{C+f-2,1} & \mathcal{P}^{1P}_{C+f-2,2} & 0 & \cdots & 0 & 0 & 0 \\ \mathcal{P}^{1P}_{C+f-1,1} & 0 & 0 & \cdots & 0 & 0 & 0 \end{pmatrix} \end{array},$$

and similarly define $\mathcal{P}^{1N}$, $\mathcal{P}^{2P}$, and $\mathcal{P}^{2N}$, containing the same ranges of indexed probabilities. In contrast to $\mathcal{P}^{1S}$ and $\mathcal{P}^{2S}$, we now must consider cases with $i+j = C+f$, since it is possible for there to be no functional machines after a failure. We still do not need to consider cases with either $i = 0$ or $j = 0$, since the event of observing an arrival to one queue (in the form of a machine failure) while switching into or serving at the other implies that both queue lengths are positive after the failure.

Finally, we define the class-1 adjusted decision probabilities

$$d_k^{[i,j]}(m,n) = \mathcal{P}^{kS}_{m+(C+f)-(i+j),n},\ k=1,2, \tag{1}$$

$$a_{k,p}^{[i,j]}(m,n) = \mathcal{P}^{kP}_{m+(C+f)-(i+j),n},\ k=1,2, \tag{2}$$

and

$$a_k^{[i,j]}(m,n) = \mathcal{P}^{kN}_{m+(C+f)-(i+j),n},\ k=1,2, \tag{3}$$

such that, for example, $d_2^{[C-l,0]}(m,n) = \mathcal{P}^{2S}_{m+l+f,n}$ and $a_1^{[C,f]}(m,n) = \mathcal{P}^{1N}_{m,n}$. Note that the inclusion of "$(C+f)$" in the subscripts above is treated as a constant (i.e., independent of the superscript of generator blocks to be defined in Section 3.2), allowing us to accurately determine the length of the class-1 queue as we reduce the effective number of machines from $[C,f]$ to $[i,j]$ in the system as part of the sojourn time analysis in Section 3.3.

3.1.1 Select Service Policies and Their Decision Probability Matrices

Within the numerical examples in Sections 4 and 5, we examine several service policies of interest which we are able to construct from specific combinations of decision probabilities. Before specifying these cases, we define $\mathcal{A}$ as the matrix $\mathcal{P}^{1P}$ if we let $\mathcal{P}^{1P}_{i,j} = 1$, $i = 1,2,\ldots,C+f-1$, $j = 1,2,\ldots,C+f-i$. That is, $\mathcal{A}$ has the same dimension and structure as the four failure instant decision probability matrices, but with each probability set equal to 1. Similarly, define $\mathcal{D}$ as the matrix $\mathcal{P}^{1S}$ with $\mathcal{P}^{1S}_{i,j} = 1$, $i = 1,2,\ldots,C+f-2$, $j = 1,2,\ldots,C+f-i-1$. Finally, for $j \in \mathbb{Z}^+$ and $i = 1,2,\ldots,j+1$, let

$$\mathcal{T}_i^{[j]} = \begin{array}{c} \\ 1 \\ 2 \\ \vdots \\ i-1 \\ i \\ i+1 \\ \vdots \\ j-1 \\ j \end{array} \begin{array}{c} \begin{array}{cccccccccc} 1 & 2 & \cdots & j-i & j+1-i & j+2-i & \cdots & j-1 & j \end{array} \\ \begin{pmatrix} 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 1 & \cdots & 1 & 1 & 0 & \cdots & 0 & 0 \\ 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots \\ 1 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix} \end{array},$$

such that in its boundary cases,

$$\mathcal{T}_i^{[j]} = \begin{cases} \mathcal{A} & , \text{ if } i=1, j=C+f-1, \\ \mathcal{D} & , \text{ if } i=1, j=C+f-2, \\ \mathbf{0} & , \text{ if } i>j, \end{cases} \tag{4}$$

where $\mathbf{0}$ denotes an appropriately-dimensioned matrix of zeroes, which in this case has dimension $j \times j$.

We now discuss our service policies of interest, whose switch-in decision probability matrices are specified in Table 1. The first service policy we consider is the classic *exhaustive* service policy, where the server remains at a particular queue until it empties, at which time a switch to the other queue is made, or for our model specifically, to class 0 if $X_1 = X_2 = 0$ at this time. Since the server will never leave a queue while it has a positive length, all decision probabilities must be zero.

Next, we have a pair of priority policies wherein the mechanic prefers to serve one class of failures before the other. We present the class-1 priority policies here, while the class-2 priority policies may be obtained by simply interchanging the class-1 and class-2 decision probabilities. For class-1 *non-preemptive priority*, the server will always immediately begin a class-1 switch-in upon observing a class-1 failure to an empty queue (note that the server is only allowed to leave queue 1 once $X_1 = 0$) so long as they do not have to interrupt, or preempt, a class-2 service time. Thus, the decision probabilities when conducting a class-2 switch-in, or after completing a class-2 service, are all one, whereas the decision probabilities during a class-2 service time are zero. In contrast, the *preemptive resume priority* policy gives the server permission to interrupt a service time, which will later be resumed with no work lost, so the decision probabilities during a class-2 service time are also set equal to one. We remark that we chose to let $\mathcal{P}^{1S} = \mathcal{D}$ in the preemptive case, even though it is not possible to observe a class-2 service completion when $X_1 > 0$ (and hence these probabilities will never be checked in practice by the CTMC).

A threshold policy is a modification of standard priority policies, in that a class' higher priority is conditional on it having a queue length equaling or exceeding a particular class-dependent threshold. As we are already considering priority policies that are both preemptive and non-preemptive in nature, we elect to use a variant of the threshold policy which can assign non-preemptive priority to a class after reaching a threshold (a), and then preemptive resume priority to a class after reaching another threshold (b) that is equal to or greater than the non-preemptive threshold. That is, if $X_1 < a$, then the server acts as if under an exhaustive policy, if $a \leq X_1 < b$, the server acts as if under a class-1 non-preemptive policy, and if $b \leq X_1$, the server acts as if under a class-1 preemptive resume priority policy. We refer to this variant as an (a, b) *threshold* policy.

In order to handle the activation of priority, we make use of the above $\mathcal{T}_i^{[j]}$ matrices which change from having decision probabilities of zero to probabilities of one once $X_1 \geq i$ (i.e., for row i and below). If we instead wanted to use a class-2 threshold policy, we would use transposes of these matrices, $\left(\mathcal{T}_i^{[j]}\right)'$, so that the policy would adjust after observing $X_2 \geq i$ (i.e., for column i and to the right). As implied by Equation (4) and Table 1, an (a, b) threshold policy can recover the exhaustive or priority policies. In fact, it can also represent a non-preemptive threshold policy if we let $b = C + f$, or a preemptive resume threshold policy if we let $b = a$.

Lastly, we consider a modification of the Bernoulli service discipline introduced by Keilson and Servi (1986). In the original discipline, after every service completion, the server would either switch away or go on vacation (in the case of a single queue system) depending on the

result of an independent Bernoulli trial having a fixed class-dependent probability. In our model, we are assuming that the mechanic has full information concerning queue lengths, and as such would not be inclined to switch away from a queue without emptying it if the opposite queue has no machines waiting to be serviced. Therefore, we implement a modified policy that we refer to as (p_1^{SB}, p_2^{SB}) *smart Bernoulli*, or simply smart Bernoulli, which only conducts a class-dependent Bernoulli trial having probability p_i^{SB} of starting another class-i service, $i = 1, 2$, rather than switching away to the opposite queue, if the opposite queue has a positive length. Hence, under this policy, the only decisions the server has to make are at service completions, and these decisions always have the same class-dependent probability for each combination of queue lengths. Finally, we remark that if we let $p_1^{SB} = p_2^{SB} = 1$, then the server never leaves a queue until it is empty and we recover the exhaustive service policy.

Table 1: Switch-in decision probability matrices for select service policies.

Service Policy	$\mathcal{P}^{1S}$	$\mathcal{P}^{1P}$	$\mathcal{P}^{1N}$	$\mathcal{P}^{2S}$	$\mathcal{P}^{2P}$	$\mathcal{P}^{2N}$
Exhaustive	$\mathbf{0}$	$\mathbf{0}$	$\mathbf{0}$	$\mathbf{0}$	$\mathbf{0}$	$\mathbf{0}$
Class-1 Non-preemptive Priority	$\mathcal{D}$	$\mathbf{0}$	$\mathcal{A}$	$\mathbf{0}$	$\mathbf{0}$	$\mathbf{0}$
Class-1 Preemptive Resume Priority	$\mathcal{D}$	$\mathcal{A}$	$\mathcal{A}$	$\mathbf{0}$	$\mathbf{0}$	$\mathbf{0}$
Class-1 (a, b) Threshold	$\mathcal{T}_a^{[C+f-2]}$	$\mathcal{T}_b^{[C+f-1]}$	$\mathcal{T}_a^{[C+f-1]}$	$\mathbf{0}$	$\mathbf{0}$	$\mathbf{0}$
(p_1^{SB}, p_2^{SB}) Smart Bernoulli	$(1-p_2^{SB})\mathcal{D}$	$\mathbf{0}$	$\mathbf{0}$	$(1-p_1^{SB})\mathcal{D}$	$\mathbf{0}$	$\mathbf{0}$

3.2 Steady-state Probabilities

We are able to solve for the steady-state probabilities by representing the system as a level-dependent quasi-birth-and-death (QBD) process, taking the length of the class-1 queue, X_1, as the level of the process. First of all, let π_{m,n,l,y,y_1,y_2} be the steady-state probability that $X_1 = m$, $X_2 = n$, $L = l$, $Y = y$, $Y_1 = y_1$, and $Y_2 = y_2$, where the variables take on values from their supports defined in Section 3.1. Next, we order the steady-state probabilities into the row vector

$$\underline{\pi} = (\underline{\pi}_0, \underline{\pi}_1, \ldots, \underline{\pi}_{C+f}), \tag{5}$$

where

$$\underline{\pi}_m = (\underline{\pi}_{m,0}, \underline{\pi}_{m,1}, \ldots, \underline{\pi}_{m,C+f-m})$$

contains the ordered steady-state probabilities for level m, $m = 0, 1, \ldots, C+f$. For level 0,

$$\underline{\pi}_{0,0} = (\pi_{0,0,0,0,0,0}, \pi_{0,0,5,1,0,0}, \ldots, \pi_{0,0,5,s_0,0,0})$$

has length $1 + s_0$, and

$$\begin{aligned}\underline{\pi}_{0,n} = (&\pi_{0,n,3,1,0,1}, \ldots, \pi_{0,n,3,1,0,b_2}, \pi_{0,n,3,2,0,1}, \ldots, \pi_{0,n,3,s_2,0,b_2},\\ &\pi_{0,n,4,0,0,1}, \ldots, \pi_{0,n,4,0,0,b_2})\end{aligned}$$

has length $(s_2+1)b_2$ for $n = 1, 2, \ldots, C+f$, resulting in $1 + s_0 + (C+f)(s_2+1)b_2$ total states. For level $m = 1, 2, \ldots, C+f$,

$$\begin{aligned}\underline{\pi}_{m,0} = (&\pi_{m,0,1,1,1,0}, \ldots, \pi_{m,0,1,1,b_1,0}, \pi_{m,0,1,2,1,0}, \ldots, \pi_{m,0,1,s_1,b_1,0},\\ &\pi_{m,0,2,0,1,0}, \ldots, \pi_{m,0,2,0,b_1,0})\end{aligned}$$

has length $(s_1+1)b_1$, and for $m = 1, 2, \ldots, C+f-1$ and $n = 1, 2, \ldots, C+f-m$,

$$\begin{aligned}\underline{\pi}_{m,n} = (&\pi_{m,n,1,1,1,1}, \ldots, \pi_{m,n,1,1,1,b_2}, \pi_{m,n,1,1,2,1}, \ldots, \pi_{m,n,1,1,b_1,b_2}, \pi_{m,n,1,2,1,1}, \ldots,\\ &\pi_{m,n,1,s_1,b_1,b_2}, \pi_{m,n,2,0,1,1}, \ldots, \pi_{m,n,2,0,1,b_2}, \pi_{m,n,2,0,2,1}, \ldots, \pi_{m,n,2,0,b_1,b_2},\\ &\pi_{m,n,3,1,1,1}, \ldots, \pi_{m,n,3,1,1,b_2}, \pi_{m,n,3,1,2,1}, \ldots, \pi_{m,0,3,1,b_1,b_2}, \pi_{m,n,3,2,1,1}, \ldots,\\ &\pi_{m,n,3,s_2,b_1,b_2}, \pi_{m,n,4,0,1,1}, \ldots, \pi_{m,n,4,0,1,b_2}, \pi_{m,n,4,0,2,1}, \ldots, \pi_{m,n,4,0,b_1,b_2})\end{aligned}$$

has length $(s_1+s_2+2)b_1b_2$, resulting in $(s_1+1)b_1 + (C+f-m)(s_1+s_2+2)b_1b_2$ total states.

The corresponding infinitesimal generator $Q^{[C,f]}$ for this QBD process takes on the form

$$Q^{[C,f]} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C+f-2 \\ C+f-1 \\ C+f \end{array} \begin{array}{c} \begin{array}{ccccccc} 0 & 1 & 2 & \cdots & C+f-2 & C+f-1 & C+f \end{array} \\ \begin{pmatrix} Q_{0,0}^{[C,f]} & Q_{0,1}^{[C,f]} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ Q_{1,0}^{[C,f]} & Q_{1,1}^{[C,f]} & Q_{1,2}^{[C,f]} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & Q_{2,1}^{[C,f]} & Q_{2,2}^{[C,f]} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{C+f-2,C+f-2}^{[C,f]} & Q_{C+f-2,C+f-1}^{[C,f]} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{C+f-1,C+f-2}^{[C,f]} & Q_{C+f-1,C+f-1}^{[C,f]} & Q_{C+f-1,C+f}^{[C,f]} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q_{C+f,C+f-1}^{[C,f]} & Q_{C+f,C+f}^{[C,f]} \end{pmatrix} \end{array}, \quad (6)$$

where each submatrix (or block) $Q_{i,j}^{[C,f]}$ contains all rates corresponding to state transitions where the level changes from i to j. Here, we use superscript "$[C,f]$" to denote the sizes of the system's capacity (C) and maintenance float (f). We will make use of this superscript notation in the sojourn time analysis of Section 3.3.

We solve for $\underline{\pi}$ from the equation $\underline{0} = \underline{\pi} Q^{[C,f]}$, where $\underline{0}$ is a row vector of zeroes having the appropriate length. From Equations (5) and (6), we obtain the block equilibrium equations

$$\underline{0} = \underline{\pi}_0 Q_{0,0}^{[C,f]} + \underline{\pi}_1 Q_{1,0}^{[C,f]}, \quad (7)$$

$$\underline{0} = \underline{\pi}_{m-1} Q_{m-1,m}^{[C,f]} + \underline{\pi}_m Q_{m,m}^{[C,f]} + \underline{\pi}_{m+1} Q_{m+1,m}^{[C,f]}, \ m = 1, 2, \ldots, C+f-1, \quad (8)$$

$$\underline{0} = \underline{\pi}_{C+f-1} Q_{C+f-1,C+f}^{[C,f]} + \underline{\pi}_{C+f} Q_{C+f,C+f}^{[C,f]}. \quad (9)$$

We can express each level's steady-state probability row vector in terms of $\underline{\pi}_0$ and apply an algorithm based on the procedure for birth-and-death models with finite state spaces proposed by Gaver et al. (1984). Specifically, applying Equations (8) and (9), it can be shown that

$$\underline{\pi}_m = \underline{\pi}_0 \prod_{j=1}^{m} U_j, \ m = 1, 2, \ldots, C+f, \quad (10)$$

where

$$U_{C+f} = -Q^{[C,f]}_{C+f-1,C+f}\left(Q^{[C,f]}_{C+f,C+f}\right)^{-1},$$

and U_j, $j = 1, 2, \ldots, C+f-1$, are obtained from the recursive relationship

$$U_j = -Q^{[C,f]}_{j-1,j}\left(Q^{[C,f]}_{j,j} + U_{j+1}Q^{[C,f]}_{j+1,j}\right)^{-1}.$$

Defining $U_0 = Q^{[C,f]}_{0,0} + U_1 Q^{[C,f]}_{1,0}$, it immediately follows from Equation (7) that $\underline{\pi}_0$ satisfies

$$\underline{\pi}_0 U_0 = \underline{0}. \tag{11}$$

Since the sum of all steady-state probabilities must equal 1, we also have

$$1 = \underline{\pi}\,\underline{e}' = \sum_{m=0}^{C+f}\underline{\pi}_m\underline{e}' = \sum_{m=0}^{C+f}\underline{\pi}_0\prod_{j=1}^{m}U_j\underline{e}' = \underline{\pi}_0\left(\sum_{m=0}^{C+f}\prod_{j=1}^{m}U_j\underline{e}'\right), \tag{12}$$

using the convention that $\prod_{j=1}^{0}U_j\underline{e}' = \underline{e}'$. We may now calculate $\underline{\pi}_0$ from the system of linear equations resulting from Equations (11) and (12), which can then be used in Equation (10) to solve for $\underline{\pi}_m$, $m = 1, 2, \ldots, C+f$.

We conclude this subsection by specifying the constructed blocks of $Q^{[C,f]}$. To this end, we require the following notation. Let $\otimes$ represent the standard Kronecker product operator, let I_i be an $i \times i$ identity matrix, and let $\underline{e}_i$ be a row vector of ones having length i. In addition, we define $\underline{B}'_{0,i} = -B_i\underline{e}'_{b_i}$ and $\underline{S}'_{0,i} = -S_i\underline{e}'_{s_i}$ as the column vectors of absorption rates for the PH$(\underline{\beta}_i, B_i)$ distributed class-i service times and PH$(\underline{\gamma}_{ji}, S_i)$ distributed class-i switch-in times, respectively, and let $\underline{\gamma}^{[+0]}_{ji} = (\underline{\gamma}_{ji}, \gamma^{[0]}_{ji})$ be the concatenated probability vector joining the initial distribution of a class j to class i switch-in with the probability of the switch-in being zero in duration. Finally, let $\Delta^{[C,f]}_{m,n} = \min\{C, C+f-m-n\}$ denote the number of working machines when $X_1 = m$ and $X_2 = n$.

For levels $m = 0, 1, \ldots, C+f$, the main diagonal blocks of $Q^{[C,f]}$ are given by

$$Q^{[C,f]}_{m,m} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C+f-m-1 \\ C+f-m \end{array}\!\!\begin{array}{c} \begin{array}{ccccccc} 0 & 1 & 2 & \cdots & C+f-m-1 & C+f-m \end{array} \\ \begin{pmatrix} Q^{[C,f]}_{m,m,0} & (UD)^{[C,f]}_{m,0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ (LD)^{[C,f]}_{m,1} & Q^{[C,f]}_{m,m,1} & (UD)^{[C,f]}_{m,1} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (LD)^{[C,f]}_{m,2} & Q^{[C,f]}_{m,m,2} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[C,f]}_{m,m,C+f-m-1} & (UD)^{[C,f]}_{m,C+f-m-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & (LD)^{[C,f]}_{m,C+f-m} & Q^{[C,f]}_{m,m,C+f-m} \end{pmatrix} \end{array},$$

where for $m = 0$

$$Q^{[C,f]}_{0,0,0} = -C\alpha I_{1+s_0} + \begin{bmatrix} 0 & \underline{0}_{s_0} \\ \underline{S}'_{0,0} & S_0 \end{bmatrix},$$

$$Q_{0,0,n}^{[C,f]} = -\Delta_{0,n}^{[C,f]}\alpha I_{(s_2+1)b_2} + \begin{bmatrix} S_2 \otimes I_{b_2} & \underline{S}'_{0,2} \otimes I_{b_2} \\ \mathbf{0} & B_2 \end{bmatrix},\ n = 1, 2, \ldots, C+f,$$

$$(UD)_{0,0}^{[C,f]} = C\alpha_2 \underline{e}'_{1+s_0} \underline{\gamma}_{02}^{[+0]} \otimes \underline{\beta}_{-2},$$

$$(UD)_{0,n}^{[C,f]} = \Delta_{0,n}^{[C,f]}\alpha_2 I_{(s_2+1)b_2},\ n = 1, 2, \ldots, C+f-1,$$

$$(LD)_{0,1}^{[C,f]} = \begin{bmatrix} \underline{0}'_{s_2 b_2} & \mathbf{0} \\ \underline{\gamma}_{20}^{[0]} \underline{B}'_{0,2} & \underline{B}'_{0,2} \underline{\gamma}_{-20} \end{bmatrix},$$

and

$$(LD)_{0,n}^{[C,f]} = \begin{bmatrix} \underline{0}'_{s_2 b_2} \underline{0}_{s_2 b_2} & \mathbf{0} \\ \mathbf{0} & \underline{B}'_{0,2} \underline{\beta}_{-2} \end{bmatrix},\ n = 2, 3, \ldots, C+f,$$

while for $m = 1, 2, \ldots, C+f$,

$$Q_{m,m,0}^{[C,f]} = -\Delta_{m,0}^{[C,f]}\alpha I_{(s_1+1)b_1} + \begin{bmatrix} S_1 \otimes I_{b_1} & \underline{S}'_{0,1} \otimes I_{b_1} \\ \mathbf{0} & B_1 \end{bmatrix},$$

and

$$Q_{m,m,n}^{[C,f]} = -\Delta_{m,n}^{[C,f]}\alpha I_{(s_1+s_2+2)b_1 b_2} + \begin{bmatrix} S_1 \otimes I_{b_1 b_2} & \underline{S}'_{0,1} \otimes I_{b_1 b_2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & B_1 \otimes I_{b_2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & S_2 \otimes I_{b_1 b_2} & \underline{S}'_{0,2} \otimes I_{b_1 b_2} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & I_{b_1} \otimes B_2 \end{bmatrix}$$

for $n = 1, 2, \ldots, C+f-m$,

$$(UD)_{m,0}^{[C,f]} = \Delta_{m,0}^{[C,f]}\alpha_2 \begin{bmatrix} (1-a_2^{[C,f]}(m,1))I_{s_1 b_1} \otimes \underline{\beta}_{-2} & \mathbf{0} & a_2^{[C,f]}(m,1)\underline{e}'_{s_1} \underline{\gamma}_{12}^{[+0]} \otimes I_{b_1} \otimes \underline{\beta}_{-2} \\ \mathbf{0} & (1-a_{2,p}^{[C,f]}(m,1))I_{b_1} \otimes \underline{\beta}_{-2} & a_{2,p}^{[C,f]}(m,1)\underline{\gamma}_{12}^{[+0]} \otimes I_{b_1} \otimes \underline{\beta}_{-2} \end{bmatrix},$$

and

$$(UD)_{m,n}^{[C,f]} = \Delta_{m,n}^{[C,f]}\alpha_2 \begin{bmatrix} (1-a_2^{[C,f]}(m,n+1))I_{s_1 b_1 b_2} & \mathbf{0} & a_2^{[C,f]}(m,n+1)\underline{e}'_{s_1} \underline{\gamma}_{12}^{[+0]} \otimes I_{b_1 b_2} \\ \mathbf{0} & (1-a_{2,p}^{[C,f]}(m,n+1))I_{b_1 b_2} & a_{2,p}^{[C,f]}(m,n+1)\underline{\gamma}_{12}^{[+0]} \otimes I_{b_1 b_2} \\ \mathbf{0} & \mathbf{0} & I_{(s_2+1)b_1 b_2} \end{bmatrix}$$

for $n = 1, 2, \ldots, C+f-m-1$, and

$$(LD)_{m,1}^{[C,f]} = \begin{bmatrix} \underline{0}'_{(s_1+s_2+1)b_1 b_2} \underline{0}_{(s_1+1)b_1} \\ \underline{\gamma}_{-21}^{[+0]} \otimes I_{b_1} \otimes \underline{B}'_{0,2} \end{bmatrix},$$

and

$$(LD)_{m,n}^{[C,f]} = \begin{bmatrix} \underline{0}'_{(s_1+s_2+1)b_1 b_2} \underline{0}_{(s_1+1)b_1 b_2} & \underline{0}'_{(s_1+s_2+1)b_1 b_2} \underline{0}_{s_2 b_1 b_2} & \underline{0}'_{(s_1+s_2+1)b_1 b_2} \underline{0}_{b_1 b_2} \\ d_1^{[C,f]}(m,n-1)\underline{\gamma}_{-21}^{[+0]} \otimes I_{b_1} \otimes \underline{B}'_{0,2} \underline{\beta}_{-2} & \mathbf{0} & (1-d_1^{[C,f]}(m,n-1))I_{b_1} \otimes \underline{B}'_{0,2} \underline{\beta}_{-2} \end{bmatrix}$$

for $n = 2, 3, \ldots, C+f-m$.

Next, for levels $m = 0, 1, \ldots, C+f-1$, the upper diagonal blocks of $Q^{[C,f]}$ have the form

$$Q^{[C,f]}_{m,m+1} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C+f-m-1 \\ C+f-m \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & \cdots & C+f-m-1 \end{array} \\ \begin{pmatrix} Q^{[C,f]}_{m,m+1,0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & Q^{[C,f]}_{m,m+1,1} & \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & Q^{[C,f]}_{m,m+1,2} & \ddots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[C,f]}_{m,m+1,C+f-m-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{pmatrix} \end{array},$$

where for $m = 0$,

$$Q^{[C,f]}_{0,1,0} = C\alpha_1 \underline{e}'_{1+s_0} \underline{\gamma}^{[+0]}_{01} \otimes \underline{\beta}_{-1},$$

and

$$Q^{[C,f]}_{0,1,n} = \Delta^{[C,f]}_{0,n} \alpha_1 \begin{bmatrix} a^{[C,f]}_1(1,n) \underline{e}'_{s_2} \underline{\gamma}^{[+0]}_{21} \otimes \underline{\beta}_{-1} \otimes I_{b_2} & (1 - a^{[C,f]}_1(1,n)) I_{s_2} \otimes \underline{\beta}_{-1} \otimes I_{b_2} & \mathbf{0} \\ a^{[C,f]}_{1,p}(1,n) \underline{\gamma}^{[+0]}_{21} \otimes \underline{\beta}_{-1} \otimes I_{b_2} & \mathbf{0} & (1 - a^{[C,f]}_{1,p}(1,n)) \underline{\beta}_{-1} \otimes I_{b_2} \end{bmatrix}$$

for $n = 1, 2, \ldots, C+f-1$, while for $m = 1, 2, \ldots, C+f-1$,

$$Q^{[C,f]}_{m,m+1,0} = \Delta^{[C,f]}_{m,0} \alpha_1 I_{(s_1+1)b_1},$$

and

$$Q^{[C,f]}_{m,m+1,n} = \Delta^{[C,f]}_{m,n} \alpha_1 \begin{bmatrix} I_{(s_1+1)b_1 b_2} & \mathbf{0} & \mathbf{0} \\ a^{[C,f]}_1(m+1,n) \underline{e}'_{s_2} \underline{\gamma}^{[+0]}_{21} \otimes I_{b_1 b_2} & (1 - a^{[C,f]}_1(m+1,n)) I_{s_2 b_1 b_2} & \mathbf{0} \\ a^{[C,f]}_{1,p}(m+1,n) \underline{\gamma}^{[+0]}_{21} \otimes I_{b_1 b_2} & \mathbf{0} & (1 - a^{[C,f]}_{1,p}(m+1,n)) I_{b_1 b_2} \end{bmatrix}$$

for $n = 1, 2, \ldots, C+f-m-1$.

Lastly, for levels $m = 1, 2, \ldots, C+f$, the lower diagonal blocks of $Q^{[C,f]}$ are given by

$$Q^{[C,f]}_{m,m-1} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C+f-m-1 \\ C+f-m \end{array} \begin{array}{c} \begin{array}{ccccccc} 0 & 1 & 2 & \cdots & C+f-m-1 & C+f-m & C+f-m+1 \end{array} \\ \begin{pmatrix} Q^{[C,f]}_{m,m-1,0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & Q^{[C,f]}_{m,m-1,1} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & Q^{[C,f]}_{m,m-1,2} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[C,f]}_{m,m-1,C+f-m-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q^{[C,f]}_{m,m-1,C+f-m} & \mathbf{0} \end{pmatrix} \end{array},$$

where for $m = 0$

$$Q^{[C,f]}_{1,0,0} = \begin{bmatrix} \underline{0}'_{s_1 b_1} & \mathbf{0} \\ \gamma^{[0]}_{10} \underline{B}'_{0,1} & \underline{B}'_{0,1} \underline{\gamma}_{10} \end{bmatrix},$$

and

$$Q^{[C,f]}_{1,0,n} = \begin{bmatrix} \underline{0}'_{s_1 b_1 b_2} \underline{0}_{(s_2+1)b_2} \\ \underline{\gamma}^{[+0]}_{12} \otimes \underline{B}'_{0,1} \otimes I_{b_2} \\ \underline{0}'_{(s_2+1)b_1 b_2} \underline{0}_{(s_2+1)b_2} \end{bmatrix}, \ n = 1, 2, \ldots, C+f-1,$$

while for $m = 2, 3, \ldots, C + f$,

$$Q^{[C,f]}_{m,m-1,0} = \begin{bmatrix} \underline{0}'_{s_1 b_1} \underline{0}_{s_1 b_1} & \mathbf{0} \\ \mathbf{0} & \underline{B}'_{0,1} \underline{\beta}_1 \end{bmatrix},$$

and

$$Q^{[C,f]}_{m,m-1,n} = \begin{bmatrix} \underline{0}'_{s_1 b_1 b_2} \underline{0}_{s_1 b_1 b_2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (1 - d_2^{[C,f]}(m-1,n)) \underline{B}'_{0,1} \underline{\beta}_1 \otimes I_{b_2} & d_2^{[C,f]}(m-1,n) \underline{\gamma}^{[+0]}_{12} \otimes \underline{B}'_{0,1} \underline{\beta}_1 \otimes I_{b_2} \\ \underline{0}'_{(s_2+1) b_1 b_2} \underline{0}_{s_1 b_1 b_2} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

for $n = 1, 2, \ldots, C + f - m$.

3.3 Sojourn Time Distribution

In this subsection, we derive the continuous phase-type representation for the sojourn time (i.e., the time between a machine's failure and when its repairs are complete) distribution of a target machine that suffers a class-1 failure, $\mathcal{S}_1$. Our analysis considers the system at steady state, and hence we require the steady-state probabilities of the maintenance system immediately prior to a class-1 failure. Letting $C_{1,h}$ denote the event of observing only a single class-1 failure within the next h time units and S_{m,n,l,y,y_1,y_2} denote the event that $(X_1, X_2, L, Y, Y_1, Y_2) = (m, n, l, y, y_1, y_2)$ at steady state (such that $P(S_{m,n,l,y,y_1,y_2}) = \pi_{m,n,l,y,y_1,y_2}$), it follows that (e.g., Lakatos et al. 2012, Chapter 9)

$$\begin{aligned}
&q_{m,n,l,y,y_1,y_2} \\
&= P((X_1, X_2, L, Y, Y_1, Y_2) = (m, n, l, y, y_1, y_2) \text{ immediately prior to a class-1 failure}) \\
&= \lim_{h \to 0} P(S_{m,n,l,y,y_1,y_2} | C_{1,h}) \\
&= \lim_{h \to 0} \frac{P(C_{1,h} | S_{m,n,l,y,y_1,y_2}) P(S_{m,n,l,y,y_1,y_2})}{\sum_{x_1,x_2,w,z,z_1,z_2} P(C_{1,h} | S_{x_1,x_2,w,z,z_1,z_2}) P(S_{x_1,x_2,w,z,z_1,z_2})} \\
&= \lim_{h \to 0} \frac{(\alpha_1 \min\{C, C + f - m - n\} h + o(h)) \pi_{m,n,l,y,y_1,y_2}}{\sum_{x_1,x_2,w,z,z_1,z_2} (\alpha_1 \min\{C, C + f - x_1 - x_2\} h + o(h)) \pi_{x_1,x_2,w,z,z_1,z_2}} \\
&= \lim_{h \to 0} \frac{\alpha_1 \min\{C, C + f - m - n\} \pi_{m,n,l,y,y_1,y_2} + o(h)/h}{\sum_{x_1,x_2,w,z,z_1,z_2} \alpha_1 \min\{C, C + f - x_1 - x_2\} \pi_{x_1,x_2,w,z,z_1,z_2} + o(h)/h} \\
&= \frac{\min\{C, C + f - m - n\} \pi_{m,n,l,y,y_1,y_2}}{\sum_{x_1,x_2,w,z,z_1,z_2} \min\{C, C + f - x_1 - x_2\} \pi_{x_1,x_2,w,z,z_1,z_2}}. \qquad (13)
\end{aligned}$$

That is, the probability that the system was in state (m, n, l, y, y_1, y_2) immediately prior to a class-1 failure is the ratio of the steady-state class-1 failure rate from state (m, n, l, y, y_1, y_2) and the total steady-state class-1 failure rate over all states.

Now that we have the distribution of the system before the failure, we must consider how the failure causes the state of the system to change. If the mechanic was previously conducting a switch-in and this failure causes a class-1 switch-in to begin, then the switch-in

phase occupied prior to the failure has no bearing on the future development of the system since we track interrupted service times, but not interrupted switch-in times. Thus, let

$$q_{m,n,3,\bullet,y_1,y_2} = \sum_{y=1}^{s_2} q_{m,n,3,y,y_1,y_2}$$

be the total probability that the server was conducting a class-2 switch-in, and define

$$\underline{q}_{0,n,3,\bullet} = (q_{0,n,3,\bullet,0,1}, q_{0,n,3,\bullet,0,2}, \ldots, q_{0,n,3,\bullet,0,b_2})$$

and

$$\underline{q}_{m,n,3,\bullet} = (q_{m,n,3,\bullet,1,1}, q_{m,n,3,\bullet,1,2}, \ldots, q_{m,n,3,\bullet,1,b_2}, q_{m,n,3,\bullet,2,1}, \ldots, q_{m,n,3,\bullet,b_1,b_2}).$$

Similarly, for the case of the queue being empty prior to the failure, let

$$q_{0,0,\bullet,\bullet,\bullet,\bullet} = q_{0,0,0,0,0,0} + \sum_{y=1}^{s_0} q_{0,0,5,y,0,0}.$$

We otherwise group the pre-failure probabilities into the following row vectors. For level $m = 1, 2, \ldots, C + f - 1$, let

$$\begin{aligned}\underline{q}_{m,0} = (&q_{m,0,1,1,1,0}, \ldots, q_{m,0,1,1,b_1,0}, q_{m,0,1,2,1,0}, \ldots, q_{m,0,1,s_1,b_1,0},\\ &q_{m,0,2,0,1,0}, \ldots, q_{m,0,2,0,b_1,0}),\end{aligned}$$

and for $n = 1, 2, \ldots, C + f - m$,

$$\begin{aligned}\underline{q}_{m,n,1} = (&q_{m,n,1,1,1,1}, \ldots, q_{m,n,1,1,1,b_2}, q_{m,n,1,1,2,1}, \ldots, q_{m,n,1,1,b_1,b_2},\\ &q_{m,n,1,2,1,1}, \ldots, q_{m,n,1,s_1,b_1,b_2}),\\ \underline{q}_{m,n,2} = (&q_{m,n,2,0,1,1}, \ldots, q_{m,n,2,0,1,b_2}, q_{m,n,2,0,2,1}, \ldots, q_{m,n,2,0,b_1,b_2}),\\ \underline{q}_{m,n,3} = (&q_{m,n,3,1,1,1}, \ldots, q_{m,n,3,1,1,b_2}, q_{m,n,3,1,2,1}, \ldots, q_{m,0,3,1,b_1,b_2},\\ &q_{m,n,3,2,1,1}, \ldots, q_{m,n,3,s_2,b_1,b_2}),\\ \underline{q}_{m,n,4} = (&q_{m,n,4,0,1,1}, \ldots, q_{m,n,4,0,1,b_2}, q_{m,n,4,0,2,1}, \ldots, q_{m,n,4,0,b_1,b_2}).\end{aligned}$$

Also, for level 0 and $n = 1, 2, \ldots, C + f - 1$, let

$$\underline{q}_{0,n,3,y} = (q_{0,n,3,y,0,1}, \ldots, q_{0,n,3,y,0,b_2}),\ y = 1, 2, \ldots, s_2,$$

and

$$\underline{q}_{0,n,4} = (q_{0,n,4,0,0,1}, \ldots, q_{0,n,4,0,0,b_2}).$$

Now, for level m, $m = 0, 1, \ldots, C + f - 1$, define the probability row vector

$$\underline{p}_{m+1} = (\underline{p}_{m+1,0}, \underline{p}_{m+1,1}, \ldots, \underline{p}_{m+1,C+f-m-1}).$$

For $m > 0$, $\underline{p}_{m+1,0} = \underline{q}_{m,0}$ and

$$\underline{p}_{m+1,n} = (\underline{p}_{m+1,n,1}, \underline{p}_{m+1,n,2}, \underline{p}_{m+1,n,3}, \underline{p}_{m+1,n,4}),\ n = 1, 2, \ldots, C + f - m - 1,$$

where

$$\begin{aligned}
\underline{p}_{m+1,n,1} &= \underline{q}_{m,n,1} + a_1^{[C,f]}(m+1,n)\underline{\gamma}_{21} \otimes \underline{q}_{m,n,3,\bullet} + a_{1,p}^{[C,f]}(m+1,n)\underline{\gamma}_{21} \otimes \underline{q}_{m,n,4},\\
\underline{p}_{m+1,n,2} &= \underline{q}_{m,n,2} + a_1^{[C,f]}(m+1,n)\gamma_{21}^{[0]}\underline{q}_{m,n,3,\bullet} + a_{1,p}^{[C,f]}(m+1,n)\gamma_{21}^{[0]}\underline{q}_{m,n,4},\\
\underline{p}_{m+1,n,3} &= \left(1 - a_1^{[C,f]}(m+1,n)\right)\underline{q}_{m,n,3},\\
\underline{p}_{m+1,n,4} &= \left(1 - a_{1,p}^{[C,f]}(m+1,n)\right)\underline{q}_{m,n,4}.
\end{aligned}$$

Here, we observe that the initial "level" of the sojourn time distribution will be increased by the new class-1 machine's presence, which is why the first index of the $\underline{p}$'s are one larger than their component $\underline{q}$'s. Additionally, if the mechanic was already at queue 1 or conducting a class-1 switch-in, then the new failure will not require them to make a decision. However, if a class-2 switch-in or service time was underway, then the failure would cause the mechanic to begin a class-1 switch-in with probability $a_1^{[C,f]}(m+1,n)$ or $a_{1,p}^{[C,f]}(m+1,n)$, respectively (and this switch-in will have a duration of zero with probability $\gamma_{21}^{[0]}$). Note as well that since there was already at least one class-1 machine at queue 1, the service phase of the lead class-1 machine was already determined.

For $m = 0$, in addition to the probability of the failure inducing server movements, we need to initialize the lead class-1 machine's service phase according to the probability vector $\underline{\beta}_1$ since there was an empty queue previous to this failure. Hence, we have

$$\underline{p}_{1,0} = (q_{0,0,\bullet,\bullet,\bullet,\bullet}\underline{\gamma}_{01} \otimes \underline{\beta}_1, q_{0,0,\bullet,\bullet,\bullet,\bullet}\gamma_{01}^{[0]}\underline{\beta}_1)$$

and

$$\underline{p}_{1,n} = (\underline{p}_{1,n,1}, \underline{p}_{1,n,2}, \underline{p}_{1,n,3}, \underline{p}_{1,n,4}),\ n = 1, 2, \ldots, C + f - 1,$$

where

$$\begin{aligned}
\underline{p}_{1,n,1} &= a_1^{[C,f]}(1,n)\underline{\gamma}_{21} \otimes \underline{\beta}_1 \otimes \underline{q}_{0,n,3,\bullet} + a_{1,p}^{[C,f]}(1,n)\underline{\gamma}_{21} \otimes \underline{\beta}_1 \otimes \underline{q}_{0,n,4},\\
\underline{p}_{1,n,2} &= a_1^{[C,f]}(1,n)\gamma_{21}^{[0]}\underline{\beta}_1 \otimes \underline{q}_{0,n,3,\bullet} + a_{1,p}^{[C,f]}(1,n)\gamma_{21}^{[0]}\underline{\beta}_1 \otimes \underline{q}_{0,n,4},\\
\underline{p}_{1,n,3} &= \left(1 - a_1^{[C,f]}(1,n)\right)(\underline{\beta}_1 \otimes \underline{q}_{0,n,3,1}, \underline{\beta}_1 \otimes \underline{q}_{0,n,3,2}, \ldots, \underline{\beta}_1 \otimes \underline{q}_{0,n,3,s_2}),\\
\underline{p}_{1,n,4} &= \left(1 - a_{1,p}^{[C,f]}(1,n)\right)\underline{\beta}_1 \otimes \underline{q}_{0,n,4}.
\end{aligned}$$

We can now construct the complete steady-state probability row vector of length

$$(C+f)\left((s_1+1)b_1 + (s_1+s_2+2)b_1b_2\frac{C+f-1}{2}\right)$$

describing the state of the system immediately after a class-1 failure, namely

$$\underline{p} = (\underline{p}_{C+f}, \underline{p}_{C+f-1}, \ldots, \underline{p}_1), \tag{14}$$

which satisfies $\underline{p}\,\underline{e}' = 1$. Before constructing the rate matrix for the machine's sojourn time distribution, we make the following observation. Since we are assuming a first-come-first-served order within each queue, no matter the service policy, a target class-1 machine will never have to wait for the service time of any machines that suffer class-1 failures after their own. However, subsequent class-1 failures may still have an impact on the target machine's sojourn time. The reason for this is twofold. A machine that fails after the target and enters behind them in their queue is a machine that cannot be at risk of entering the opposite queue and potentially receiving service before the target. Also, further class-1 machine failures behind the target may yet influence the mechanic, as the switch-in decision probabilities can be unique for every combination of both (positive) queue lengths.

It then follows that to model the sojourn time, we must track both the position of the target class-1 machine within their queue, as well as the total length of their queue. We achieve this by effectively reducing the number of machines that the system needs to track after every class-1 failure following that of the target, such that the number of reductions is the excess queue length behind the target. This is where we make use of the previous QBD block superscripts, $[C, f]$, as it allows us to construct our generator blocks as functions of C and f, which otherwise would have simply been treated as constants. Note that by reducing the number of considered machines, we are not necessarily reducing the maximum that may be in use at a given time. Therefore, it is important to reduce f to zero before reducing C. Combined with this use of notation, the application of Equations (1)-(3) ensure that the true queue lengths are used when referencing the switch-in decision probabilities.

The sojourn time's rate matrix can thus be constructed as follows:

$$\mathcal{R}_1 = \begin{array}{c} \\ [C,f] \\ [C,f-1] \\ [C,f-2] \\ \vdots \\ [C,1] \\ [C,0] \\ [C-1,0] \\ \vdots \\ [2,0] \\ [1,0] \end{array} \begin{array}{c} \begin{array}{cccccccccc} [C,f] & [C,f-1] & [C,f-2] & \cdots & [C,1] & [C,0] & [C-1,0] & \cdots & [2,0] & [1,0] \end{array} \\ \begin{pmatrix} \tilde{Q}^{[C,f]} & \tilde{Q}_{-}^{[C,f]} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \tilde{Q}^{[C,f-1]} & \tilde{Q}_{-}^{[C,f-1]} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \tilde{Q}^{[C,f-2]} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \tilde{Q}^{[C,1]} & \tilde{Q}_{-}^{[C,1]} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \tilde{Q}^{[C,0]} & \tilde{Q}_{-}^{[C,0]} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \tilde{Q}^{[C-1,0]} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \tilde{Q}^{[2,0]} & \tilde{Q}_{-}^{[2,0]} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \tilde{Q}^{[1,0]} \end{pmatrix} \end{array}, \tag{}$$

where

$$\tilde{Q}^{[i,j]} = \begin{array}{c} \\ i+j \\ i+j-1 \\ i+j-2 \\ \vdots \\ 2 \\ 1 \end{array} \begin{array}{c} \begin{array}{ccccccc} i+j & i+j-1 & i+j-2 & \cdots & 2 & 1 \end{array} \\ \begin{pmatrix} Q^{[i,j]}_{i+j,i+j} & Q^{[i,j]}_{i+j,i+j-1} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & Q^{[i,j]}_{i+j-1,i+j-1} & Q^{[i,j]}_{i+j-1,i+j-2} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & Q^{[i,j]}_{i+j-2,i+j-2} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[i,j]}_{2,2} & Q^{[i,j]}_{2,1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q^{[i,j]}_{1,1} \end{pmatrix} \end{array}$$

and

$$\tilde{Q}^{[i,j]}_{-} = \begin{array}{c} \\ i+j \\ i+j-1 \\ i+j-2 \\ \vdots \\ 2 \\ 1 \end{array} \begin{array}{c} \begin{array}{ccccc} i+j-1 & i+j-2 & \cdots & 2 & 1 \end{array} \\ \begin{pmatrix} \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ Q^{[i,j]}_{i+j-1,i+j} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & Q^{[i,j]}_{i+j-2,i+j-1} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & Q^{[i,j]}_{2,3} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q^{[i,j]}_{1,2} \end{pmatrix} \end{array},$$

such that $\mathcal{R}_1$ is a square matrix of dimension

$$\frac{(C+f)(C+f+1)}{2}\left((s_1+1)b_1 + (s_1+s_2+2)b_1b_2\frac{C+f-1}{3}\right).$$

If $f = 0$, then $\mathcal{R}_1$ is the bottom right quadrant starting with level $[C, 0]$ and top-left block $\tilde{Q}^{[C,0]}$. From the above, the absorption rates are contributed from $Q^{[i,j]}_{1,0}$ subblocks, corresponding to possible transitions which would result in the lead machine (in this case, the target) receiving service and exiting the queue.

With the rate matrix in hand, we return to the initial probability row vector, $\underline{p}$. This vector contains probabilities for the system immediately after the target machine's failure, which considers all $C + f$ machines, as only one machine can fail at a time. This of course implies that at the time instant when the target machine enters its queue, there cannot be any other machines queued behind it. Thus, the initial probability vector corresponding to the phase-type distribution having rate matrix $\mathcal{R}$ is $\underline{\Phi}_1 = (\underline{p}, \underline{0}, \underline{0}, \ldots, \underline{0})$, and so it holds that $\mathcal{S}_1 \sim \mathrm{PH}(\underline{\Phi}_1, \mathcal{R}_1)$.

We conclude this subsection with the following comment. We have so far considered only the sojourn time distribution of a class-1 machine. If we want the distribution of $\mathcal{S}_2$ for a machine that suffers a class-2 failure, the distribution can be obtained via the interchange of exponential failure rates, service and switch-in time distributions, and transposes of switch-in decision probability matrices (e.g., replace $\mathcal{P}^{1S}$ by $\left(\mathcal{P}^{2S}\right)'$ and $\mathcal{P}^{2S}$ by $\left(\mathcal{P}^{1S}\right)'$). Following

this, the class-2 sojourn time distribution can be obtained by simply repeating the analysis contained within this section, treating it as the new class 1 (and hence class 1 as the new class 2), and calculating the equivalent $\underline{\Phi}_2$ and $\mathcal{R}_2$.

4 Results Concerning the Expected Number of Working Machines

4.1 Limit Theorems

In this section, we investigate some behaviors of the expected number of working machines at steady state, defined as

$$\begin{aligned}\mathrm{E}[N_W] &= \mathrm{E}[\min\{C, C+f-X_1-X_2\}] \\ &= \sum_m \sum_n \sum_l \sum_y \sum_{y_1} \sum_{y_2} \min\{C, C+f-m-n\}\pi_{m,n,l,y,y_1,y_2}. \end{aligned} \tag{15}$$

Specifically, we are interested in the impact of C and f on $\mathrm{E}[N_W]$, so for the sake of clarity within the theorems of this section, we adjust our notation slightly so that $N_W^{[C,f]} = \min\{C, C+f-X_1^{[C,f]}-X_2^{[C,f]}\}$ and $\pi_{m,n,l,y,y_1,y_2}^{[C,f]}$ denote the number of working machines and steady-state probabilities, respectively, as functions of C and f.

Our first theorem demonstrates the effect of reducing the maximum number of working machines by one to begin a maintenance float.

Theorem 1 *For a system at steady state with $k = 2, 3, \ldots$ total machines,* $\mathrm{E}[N_W^{[k,0]}] > \mathrm{E}[N_W^{[k-1,1]}]$.

Proof Refer to the Appendix.

□

Remark 1 At the end of the proof of Theorem 1, we show that $\mathrm{E}[N_W^{[k,0]}] = c_k \mathrm{E}[N_W^{[k-1,1]}]$ where $1 < c_k < \frac{k}{k-1}$, $k = 2, 3, \ldots$, so it follows that the negative impact of reducing the maximum number of working machines to begin a maintenance float goes to 0 as $k \to \infty$. Therefore, we observe that

$$\lim_{k\to\infty} \mathrm{E}[N_W^{[k,0]}] = \lim_{k\to\infty} \mathrm{E}[N_W^{[k-1,1]}] = \lim_{k\to\infty} \mathrm{E}[N_W^{[k,1]}],$$

implying that the act of including a maintenance float of size $f = 1$ does not impact the limit of the expected number of working machines in comparison to not using a maintenance float.

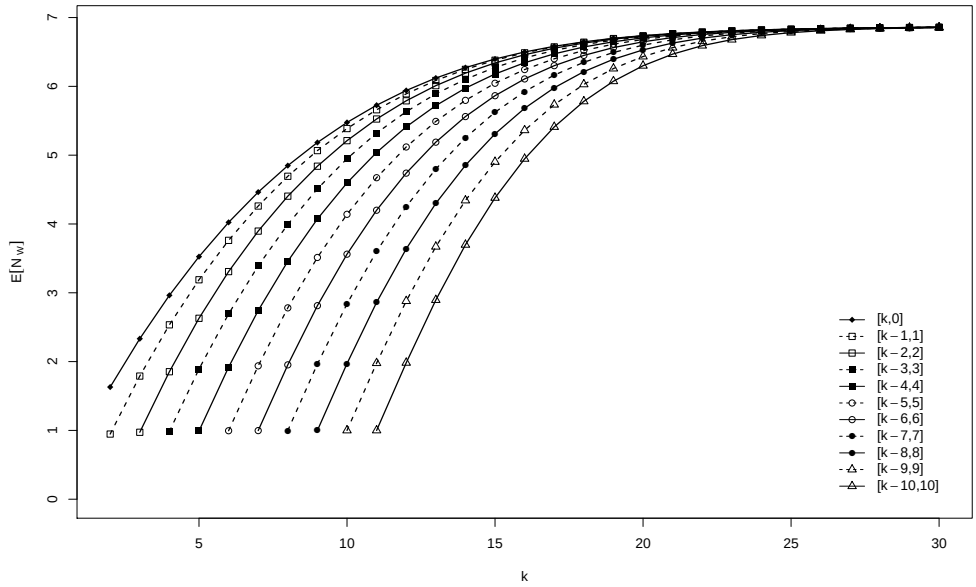

Figure 2: Plot of the expected number of working machines $\mathrm{E}[N_W]$ against the total number of machines k for maintenance floats $f = 0, 1, \ldots, 10$, under an exhaustive service policy.

To get an idea if this also holds true for larger maintenance floats, we have plotted in Figure 2 $\mathrm{E}[N_W]$ against the total number of machines k (minimum 2), for the cases $[k, k-f]$, $f = 0, 1, \ldots, 10$. In this plot, we have used an exhaustive service policy with exponentially distributed service times having means 1 and 20 for classes 1 and 2, respectively. Switch-in times between classes are also exponentially distributed with means 1, 0.5, and 1 for classes 0, 1, and 2, respectively. The total failure rate was $\alpha = 0.05$, with $\alpha_1 = 0.9\alpha$ and $\alpha_2 = 0.1\alpha$, so that most jobs were "small". This 90:10 split will be used throughout this paper unless otherwise specified.

From Figure 2, we observe that independent of how many machines we divert to the float, as the total number of machines k is increased, all of the $\mathrm{E}[N_W^{[k,k-f]}]$ values converge to a single limit as the distance between vertically adjacent points goes to 0. Additionally, we can see that increasing f for a fixed C increases $\mathrm{E}[N_W]$, but of course cannot increase it past the value of C based on the definition in Equation (15). This limit result is not a coincidence, nor unique to the exhaustive service policy, as we state in our next theorem.

Theorem 2 *For any service policy and fixed maintenance float size of $f = 0, 1, 2, \ldots$ machines, the limit of the number of working machines satisfies*

$$\mathrm{E}[N_W^{[\infty]}] = \lim_{C\to\infty} \mathrm{E}[N_W^{[C,f]}] \leq \frac{-1}{\alpha_1 \underline{\beta}_1 B_1^{-1} \underline{e}'_{b_1} + \alpha_2 \underline{\beta}_2 B_2^{-1} \underline{e}'_{b_2}}. \tag{16}$$

*Additionally, if switch-in times between the class-*1 *and class-*2 *queues are identically zero*

(i.e., $\gamma_{ji}^{[0]} = 1 \ \forall \ i, j \in \{1, 2\}$), then the upper bound will surely be reached, i.e.,

$$\mathrm{E}[N_W^{[\infty]}] = \frac{-1}{\alpha_1 \underline{\beta}_1 B_1^{-1} \underline{e}'_{b_1} + \alpha_2 \underline{\beta}_2 B_2^{-1} \underline{e}'_{b_2}}. \tag{17}$$

Proof Refer to the Appendix.

□

Remark 2 We can re-express Equation (17) as

$$\alpha \mathrm{E}[N_W^{[\infty]}] = \mathrm{E}[Z^m]^{-1},$$

where $\alpha \mathrm{E}[N_W^{[\infty]}]$ is the average rate of machine failures as $C \to \infty$ and $\mathrm{E}[Z^m]^{-1}$ is the average rate of machine repairs when the fraction of time that the mechanic is servicing machines goes to 1. Therefore, we can interpret $\mathrm{E}[N_W^{[\infty]}]$ as the expected queue length that reaches an equilibrium which balances the rate of failures with the server's fastest possible rate of repairs. If there are no switch-in times, then any policy can reach this repair rate. However, if switch-ins are possibly incurred when transiting between the class-1 and class-2 queues, then the quantity of these switch-ins (dependent on the service policy) will cause the server to spend a larger fraction of their time idle, lowering their peak repair rate and hence lowering the value of $\mathrm{E}[N_W^{[\infty]}]$ that a policy can reach.

Remark 3 For a given service policy, if the expected time servicing machines between renewals as defined in the proof of Theorem 2, $\mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]$, increases faster than the expected time switching between queues, $\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}]$, then by Equation (48), the aggregate rate of machine repairs, $\lambda_r^{[C,f]}$, and hence the expected number of working machines, $\mathrm{E}[N_W^{[C,f]}]$, are monotonically increasing in C for a given f.

From our numerical analysis, this appears to be normal behavior, but we were able to replicate a non-monotonic or monotonic decreasing relationship between $\mathrm{E}[N_W]$ and C. For example, we observed this in some cases using an unreasonable service policy that sets every decision epoch probability to 1 for both queues (i.e., the mechanic would always switch after observing any arrival to the opposite queue, and after service completions if the opposite queue had a positive length), with the aim of maximizing the number of switches. In Figure 3, under this policy, we plot $\mathrm{E}[N_W]$ against C with $f = 0$, symmetric classes having failure rates $\alpha_1 = \alpha_2 = 0.05$ and exponentially distributed service times with mean 2, and exponentially distributed switch-in times for the three classes having equal means of $M_{\mathrm{S}} \in \{0.5, 1, 2, 3, 4, 5, 10, 15, 20, 25, 50\}$. It is clear that a slight non-monotonic relationship is visible in the $M_{\mathrm{S}} = 0.5$ case which becomes more pronounced as M_{S} increases, eventually turning into a monotonic decreasing relationship in C. Omitted from these plots, we also considered the impact of f, which had no bearing on the limiting value of $\mathrm{E}[N_W^{[\infty]}]$.

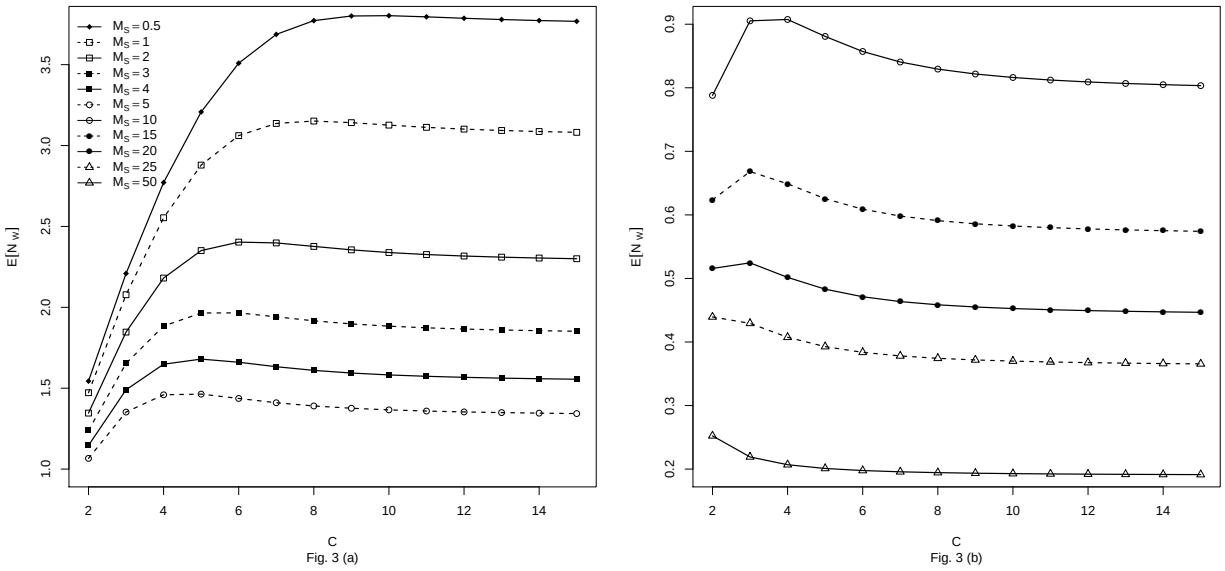

Figure 3: Plots of the expected number of working machines $\mathrm{E}[N_W]$ against the capacity C for $f = 0$, $\alpha_1 = \alpha_2 = 0.05$, and exponentially distributed services and switch-in times having means 2 or M_{S}, respectively, under a service policy that maximizes the number of switches.

To accompany Theorem 2, Table 2 presents $\mathrm{E}[N_W^{[C,f]}]$ obtained using the methods in Section 3, $\lambda_r^{[C,f]} = \mathrm{E}[N_W^{[C,f]}]/\mathrm{E}[W^{[C,f]}] = \alpha \mathrm{E}[N_W^{[C,f]}]$, simulated values of $\tilde{\mathrm{E}}[BP_{\mathrm{ser}}^{[C,f]}]$ and $\tilde{\mathrm{E}}[BP_{\mathrm{swi}}^{[C,f]}]$ obtained from 500,000 simulated renewal cycles as defined in the proof, as well as the corresponding simulated value

$$\tilde{\lambda}_r^{[C,f]} = \frac{\tilde{\mathrm{E}}[BP_{\mathrm{ser}}^{[C,f]}]/\mathrm{E}[Z^m]}{\frac{1}{C\alpha} + \tilde{\mathrm{E}}[BP_{\mathrm{ser}}^{[C,f]}] + \tilde{\mathrm{E}}[BP_{\mathrm{swi}}^{[C,f]}]}.$$

Select values of C and f are considered, along with several service policies (exhaustive, preemptive resume priority (P), non-preemptive priority (NP), smart Bernoulli (SB), and class-1 (a, b) threshold priority (Thr)). Note that we suppress the superscripts for space considerations. In all cases, the total failure rate was set to $\alpha = 0.05$, and the service times followed hyperexponential-2 (henceforth referred to as H_2 service) distributions with initial probability vectors

$$\underline{\beta}_1 = \underline{\beta}_2 = (0.9, 0.1) \tag{18}$$

and rate matrices

$$B_1 = 2 \begin{pmatrix} -1 & 0 \\ 0 & -\frac{1}{11} \end{pmatrix}, \; B_2 = \frac{1}{10 M_{\mathrm{B}}} \begin{pmatrix} -1 & 0 \\ 0 & -\frac{1}{11} \end{pmatrix}, \tag{19}$$

resulting in means of 1 and $20M_{\mathrm{B}}$ for classes 1 and 2, respectively, such that M_{B} can be used as a scaling factor to adjust the expected size of class-2 jobs, with M_{B} set to 1 by default.

For the switch-in time distributions, we used initial probability vectors

$$\underline{\gamma}_{10} = (p_{>0}, 0),\ \underline{\gamma}_{20} = (0, p_{>0}),\ \underline{\gamma}_{0i} = (0, p_{>0}, 0),\ i = 1, 2, \tag{20}$$

and

$$\underline{\gamma}_{ji} = (p_{>0}, 0, 0),\ i, j \in \{1, 2\},\ i \neq j, \tag{21}$$

where $p_{>0} = 1 - \gamma_{ji}^{[0]}$ is the probability of a switch-in time being positive in duration, and rate matrices

$$S_1 = \frac{1}{M_{\mathrm{S}}} \begin{pmatrix} -1 & 1 & 0 \\ 0 & -2 & 2 \\ 0 & 0 & -2 \end{pmatrix},\ S_2 = \frac{1}{M_{\mathrm{S}}} \begin{pmatrix} -2 & 2 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & -1 \end{pmatrix}, \tag{22}$$

and

$$S_0 = \frac{1}{M_{\mathrm{S}}} \begin{pmatrix} -2 & 0 \\ 0 & -1 \end{pmatrix}, \tag{23}$$

where M_{S} is a scaling factor for all mean switch-in times that is set to 1 by default. These rate matrices imply class-dependent Erlang-2 (denoted by E_2) distributed setup times before beginning service, and exponentially distributed takedown times before leaving either class 1 or 2. If the opposite queue is empty and the mechanic would switch to class 0, then they will complete the takedown and only be required to perform a setup after the next failure. As class 1 is being used to denote the smaller jobs, we let these times for class 1 be faster than those for class 2.

For these cases, note that $\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}] = 0$ when $p_{>0} = 0$, so we omit the corresponding column. In all cases, based on the results of Theorem 2, it follows that $\mathrm{E}[N_W^{[\infty]}] \leq 6.896552$ and $\lambda_r^{[\infty]} \leq 0.3448276$. Comparing the $p_{>0} = 0$ and $p_{>0} = 1$ cases, it is clear that the presence of switch-in times reduces the rate at which machines are repaired, as is evident in the values of $\lambda_r^{[C,f]}$.

In the absence of switch-in times, the class-1 preemptive resume priority policy outperforms the others as it prioritizes increasing the expected number of working machines (at the cost of longer class-2 sojourn times) by always choosing to repair small class-1 failures as they occur, to get those machines up and working again as soon as possible. As repaired machines that are put to work are again at risk of failure, a machine that would have otherwise had to wait for a class-2 machine service time to complete could be repaired multiple times (if it suffers another class-1 failure) during this time span, effectively increasing the aggregate rate of machine failures.

However, when switch-ins are present, every time a class-1 failure causes the mechanic to leave the class-2 queue, an extra idle period is incurred which reduces the mechanic's efficiency at a noticeable cost to $\lambda_r^{[C,f]}$. In contrast to all other policies, the ratio of $\tilde{\mathrm{E}}[BP_{\mathrm{swi}}^{[C,f]}]$ to $\tilde{\mathrm{E}}[BP_{\mathrm{ser}}^{[C,f]}]$ is by far the highest. The magnitudes of these values for class-1 preemptive priority is due to the fact that the preemptive nature with switch-ins requires a long period of time to actually empty the class-2 queue. With switch-ins, we see the $(5, 5)$ threshold,

$(6,7)$ threshold, and $(9,9)$ threshold policies maximize $\mathrm{E}[N_W^{[C,f]}]$ for the $[8,2]$, $[8,6]$, and $[14,0]$ cases, respectively. In fact, these are the optimal choices of a and b for (a,b) threshold policies in these positive switch-in cases, as we will demonstrate in Section 4.2 for the $[8,6]$ and $[14,0]$ cases.

Table 2: $\mathrm{E}[N_W^{[C,f]}]$, $\lambda_r^{[C,f]}$, and simulated values of $\mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]$ and $\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}]$ for select C, f, and $p_{>0}$ and various service policies, with $\alpha = 0.05$, H_2 service, and $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$. $\mathrm{E}[N_W^{[\infty]}] \leq 6.896552$ and $\lambda_r^{[\infty]} \leq 0.3448276$.

	$p_{>0}$								
$[C,f] = [8,2]$	0				1				
Service Policy	$\mathrm{E}[N_W]$	λ_r	$\tilde{\mathrm{E}}[BP_{\mathrm{ser}}]$	$\tilde{\lambda}_r$	$\mathrm{E}[N_W]$	λ_r	$\tilde{\mathrm{E}}[BP_{\mathrm{ser}}]$	$\tilde{\mathrm{E}}[BP_{\mathrm{swi}}]$	$\tilde{\lambda}_r$
Exhaustive	5.0419	0.2521	6.8018	0.2521	4.8525	0.2426	10.2967	1.8395	0.2426
Class-1 P	5.8966	0.2948	14.6109	0.2944	4.0018	0.2001	206.5702	147.1180	0.2000
Class-1 NP	5.1829	0.2591	7.5561	0.2591	4.9006	0.2450	17.6155	4.7238	0.2445
Class-2 P	4.9782	0.2489	6.4578	0.2486	4.7242	0.2362	9.4712	1.8473	0.2363
Class-2 NP	4.9927	0.2496	6.5196	0.2492	4.7503	0.2375	9.5967	1.8281	0.2376
(1,0.2) SB	5.1544	0.2577	7.3341	0.2572	4.9016	0.2451	12.2305	2.4778	0.2451
(1,0.8) SB	5.0689	0.2534	6.9446	0.2536	4.8657	0.2433	10.6010	1.9495	0.2429
(5,5) Thr	5.5130	0.2756	9.9895	0.2758	5.0858	0.2543	17.6209	3.7948	0.2541
(6,7) Thr	5.3217	0.2661	8.4661	0.2662	5.0288	0.2514	13.5382	2.5496	0.2512
(9,9) Thr	5.1035	0.2552	7.1373	0.2554	4.8971	0.2449	10.9218	1.9408	0.2451
$[C,f] = [8,6]$									
Exhaustive	5.3341	0.2667	8.4668	0.2662	5.1970	0.2599	13.7285	1.9851	0.2599
Class-1 P	6.3007	0.3150	26.0904	0.3147	4.0234	0.2012	5186.4519	3701.8065	0.2012
Class-1 NP	5.5949	0.2797	10.7111	0.2796	5.3015	0.2651	39.8718	9.5118	0.2650
Class-2 P	5.2502	0.2625	8.0464	0.2631	5.0338	0.2517	12.1819	2.0277	0.2514
Class-2 NP	5.2616	0.2631	8.0887	0.2634	5.0591	0.2530	12.3493	1.9994	0.2527
(1,0.2) SB	5.5457	0.2773	10.2455	0.2772	5.3380	0.2669	20.2300	3.4138	0.2668
(1,0.8) SB	5.3846	0.2692	8.8813	0.2691	5.2331	0.2617	14.7609	2.1864	0.2617
(5,5) Thr	6.1168	0.3058	19.3409	0.3054	5.5402	0.2770	51.5775	10.1664	0.2768
(6,7) Thr	5.9823	0.2991	16.5053	0.2995	5.5911	0.2796	33.4503	5.3725	0.2791
(9,9) Thr	5.7850	0.2892	13.0782	0.2895	5.5203	0.2760	23.1265	3.3062	0.2756
$[C,f] = [14,0]$									
Exhaustive	6.1840	0.3092	12.3103	0.3090	5.8612	0.2931	22.7827	2.5702	0.2933
Class-1 P	6.7814	0.3391	83.5556	0.3390	4.0222	0.2011	71885.4765	51371.5087	0.2011
Class-1 NP	6.3934	0.3197	18.0328	0.3195	5.4919	0.2746	216.4525	53.9657	0.2746
Class-2 P	6.0973	0.3049	10.9145	0.3049	5.6571	0.2829	18.4622	2.6176	0.2828
Class-2 NP	6.1098	0.3055	11.0118	0.3052	5.6904	0.2845	18.7949	2.5705	0.2843
(1,0.2) SB	6.3526	0.3176	16.6906	0.3176	5.8373	0.2919	41.9483	6.1635	0.2920
(1,0.8) SB	6.2231	0.3112	13.2559	0.3113	5.8672	0.2934	25.2003	2.9786	0.2935
(5,5) Thr	6.6466	0.3323	38.3886	0.3325	5.7470	0.2874	142.2960	27.0132	0.2874
(6,7) Thr	6.5632	0.3282	28.2071	0.3282	5.9142	0.2957	73.7993	10.8060	0.2958
(9,9) Thr	6.4484	0.3224	20.5086	0.3224	5.9709	0.2985	43.3606	5.3009	0.2985

In Figure 2, we observed in the case of the exhaustive service policy that $\mathrm{E}[N_W]$ converged to a limit as we increased the number of machines in the system, a result supported by Theorem 2. We now aim to expand on this by plotting in Figures 4 and 5 $\mathrm{E}[N_W^{[C,f]}]$ for exhaustive, class-i preemptive resume and non-preemptive priority, $i = 1, 2$, as well as $(1, 0.2)$ and $(1, 0.8)$ smart Bernoulli service policies against C for $f = 0$ or $f = 4$. We used the same phase-type service and switch-in distributions used for Table 2. As Theorem 2 states, the presence of switch-ins will affect the limit of $\mathrm{E}[N_W^{[C,f]}]$, so we consider $p_{>0} = 0$ in Figure 4 and $p_{>0} = 1$ in Figure 5. In Figure 4, we focus on the $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$ and $\alpha = 0.05$ case, while in Figure 5 we also allow $\alpha = 0.1$ and $M_{\mathrm{B}} = 0.5$. Note that due to space constraints, the legend provided in Figure 4 is representative of itself as well as Figure 5. In all plots, the horizontal grey line is the corresponding limit or upper bound from Equations (16) and (17).

In Figure 4, we confirm that in the absence of switch-in times, this range of service policies all eventually reach the same limiting expected number of working machines, with or without a maintenance float. Unlike Figure 2, we are specifically plotting against C, and hence the systems plotted in Figure 4 (b) have 4 more total machines for a given C. An increase in $\mathrm{E}[N_W]$ is observed from the presence of a maintenance float, which increases the speed at which each policy approaches $\mathrm{E}[N_W^{[\infty]}]$. Consistent with Table 2, with $p_{>0} = 0$ the preemptive resume priority policy converges to $\mathrm{E}[N_W^{[\infty]}]$ at the highest rate, followed by the other policies in an order depending on their preference to serve class-1 machines (the small jobs) over class-2 machines (the large jobs), with class-2 priority policies performing the worst. Not surprisingly, $(1, 0.2)$ smart Bernoulli is comparable to class-1 non-preemptive priority (which is equivalent to a $(1, 0)$ smart Bernoulli policy), and $(1, 0.8)$ smart Bernoulli is comparable to exhaustive (which is equivalent to a $(1, 1)$ smart Bernoulli policy). There appears to be very little difference between class-2 preemptive resume and non-preemptive priorities, resulting from a combination of low class-2 failure rates relative to class 1 as well as small class-1 service times.

In Figure 5, we overlay both the $f = 0$ and $f = 4$ cases on the same plots. In every plot, we observe the same order of service policies in terms of magnitude of $\mathrm{E}[N_W^{[\infty]}]$, with exhaustive having the highest limit (as it incurs the fewest switch-ins) followed by the other policies in reverse order depending on their relative fraction of times spent in a switch-in during a busy period as defined in the proof of Theorem 2, i.e., relative to each policy's value of

$$\lim_{C\to\infty} \frac{\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}]}{\mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]}.$$

Comparing Figure 5 (a) and (c) to (b) and (d), doubling α approximately halves $\mathrm{E}[N_W^{[\infty]}]$. Comparing the Figure 5 (a) and (b) to (c) and (d), decreasing M_{B} and hence reducing the size of large jobs increases $\mathrm{E}[Z^m]^{-1}$, increasing the mechanic's peak rate of repair and hence $\mathrm{E}[N_W^{[\infty]}]$. It is also intuitive to observe that the number of machines required to converge to a policy's limiting expected number of working machines depends on the magnitude of $\mathrm{E}[N_W^{[\infty]}]$, and by including a maintenance float without reducing C, this limit is reached at a

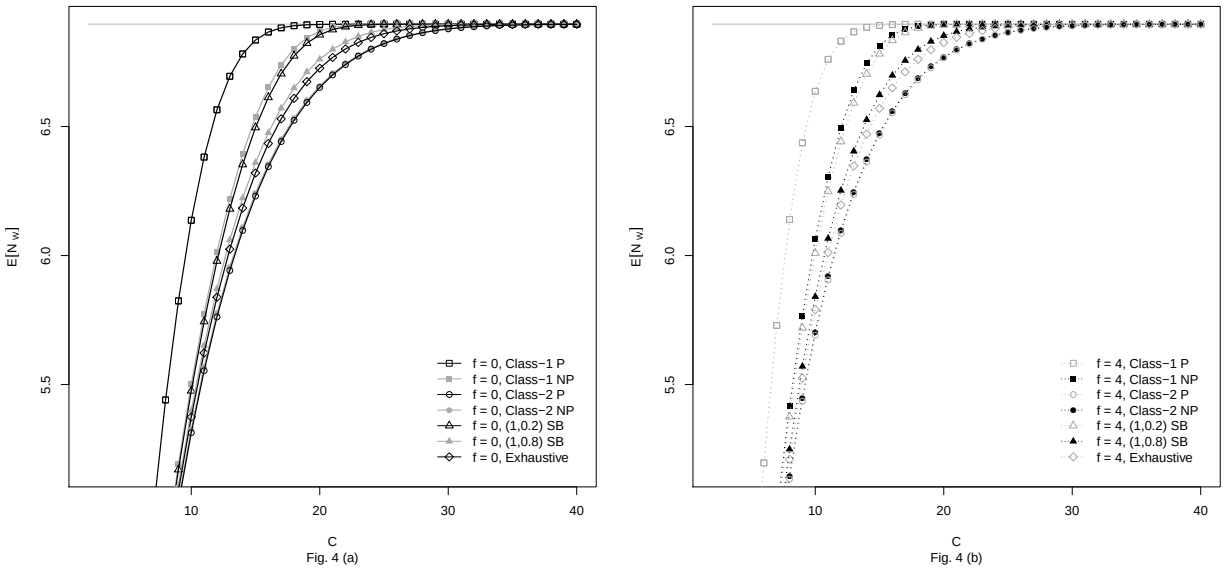

Figure 4: Plots of $\mathrm{E}[N_W^{[C,f]}]$ against C for $f = 0, 4$ and select service policies with H_2 service, $p_{>0} = 0$, $M_\mathrm{B} = 1$, $M_\mathrm{S} = 1$, and $\alpha = 0.05$, where $\mathrm{E}[N_W^{[\infty]}] = 6.896552$.

lower value of C. For all the plots in Figures 4 and 5, the convergence to a policy's $\mathrm{E}[N_W^{[\infty]}]$ is monotonic, demonstrating that they satisfy the condition described in Remark 3.

4.2 Connection to Mean Sojourn Times

In Section 3.3, we showed that the amount of time between a class-1 machine failure and when it is repaired (i.e., its sojourn time) has a $\mathrm{PH}(\underline{\Phi}_1, \mathcal{R}_1)$ distribution, and noted that an equivalent $\mathrm{PH}(\underline{\Phi}_2, \mathcal{R}_2)$ distribution can be derived for class-2 machines by using the same method after interchanging the class-1 and class-2 failure rates, service and switch-in distributions, and transposes of switch-in decision probability matrices. It then follows that the sojourn time for an arbitrary failed machine, $\mathcal{S}$, is the mixture of $\mathcal{S}_1$ and $\mathcal{S}_2$ having mixing weights equal to the probability of a given failure being of either class, resulting in the probability density function (pdf)

$$f_\mathcal{S}(t) = \frac{\alpha_1}{\alpha}\underline{\Phi}_1 \exp\{\mathcal{R}_1 t\}\underline{\mathcal{R}}'_{0,1} + \frac{\alpha_2}{\alpha}\underline{\Phi}_2 \exp\{\mathcal{R}_2 t\}\underline{\mathcal{R}}'_{0,2},$$

where $\underline{\mathcal{R}}'_{0,i} = -\mathcal{R}_i \underline{e}'$ is the column vector of absorption rates for the class-i sojourn time distribution. It is straightforward to confirm that the r^{th} moment for $\mathcal{S}$ has formula

$$\mathrm{E}[\mathcal{S}^r] = (-1)^r r! \left(\frac{\alpha_1}{\alpha}\underline{\Phi}_1 \mathcal{R}_1^{-r} \underline{e}' + \frac{\alpha_2}{\alpha}\underline{\Phi}_2 \mathcal{R}_2^{-r} \underline{e}'\right). \tag{24}$$

If we are specifically interested in the first moment of the sojourn time, then we can obtain an alternate formula using Little's Law (Little 1961) in terms of the mean queue lengths

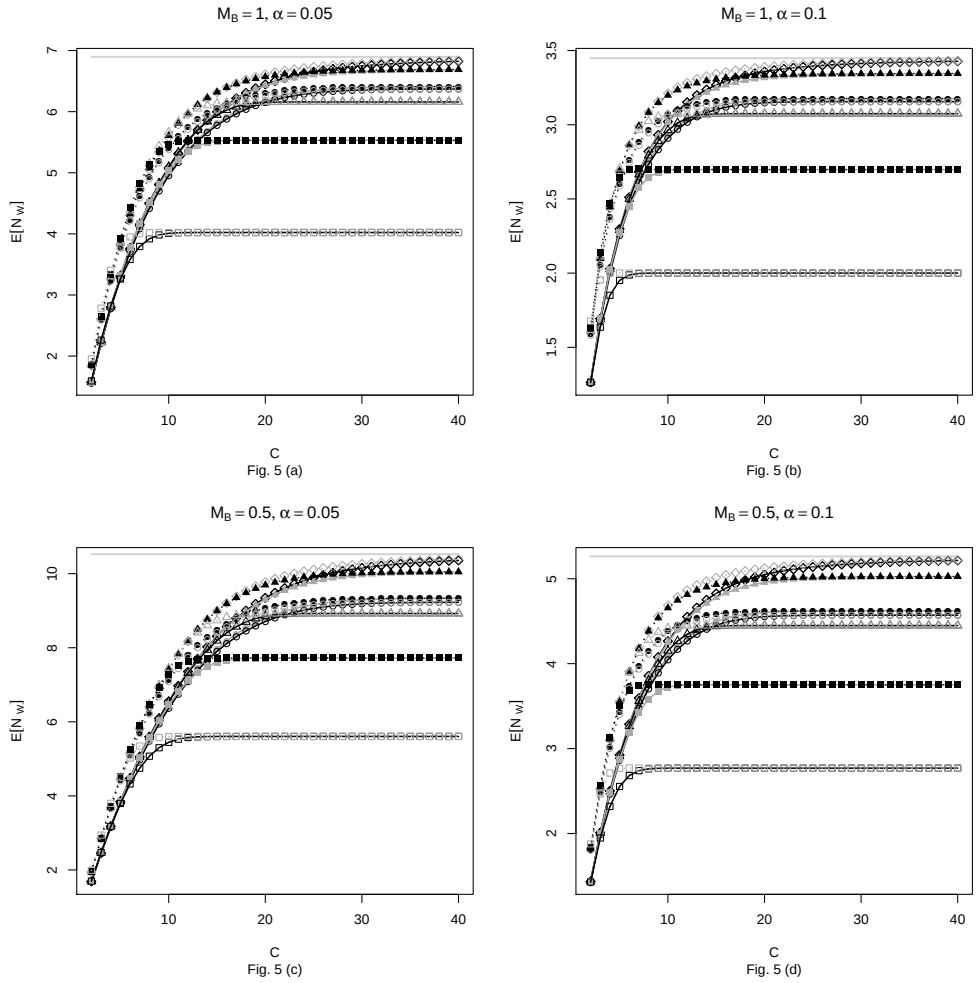

Figure 5: Plots of $\mathrm{E}[N_W^{[C,f]}]$ against C for $f = 0, 4$ and select service policies with H_2 service, $p_{>0} = 1$, $M_{\mathrm{B}} = 0.5, 1$, $M_{\mathrm{S}} = 1$, and $\alpha = 0.05, 0.1$.

and expected number of working machines. Treating the length of the class-i queue as a subsystem, the mean arrival rate to that subsystem is the mean class-i failure rate, $\bar{\alpha}_i = \alpha_i \mathrm{E}[N_W]$, and the time spent in the subsystem by a target machine is of course distributed as a class-i sojourn time. Thus, applying Little's Law, we obtain

$$\mathrm{E}[X_i] = \alpha_i \mathrm{E}[N_W]\mathrm{E}[\mathcal{S}_i], \ i = 1, 2. \tag{25}$$

As $\mathcal{S}$ is a mixture of $\mathcal{S}_1$ and $\mathcal{S}_2$, it holds that $\mathrm{E}[\mathcal{S}] = \frac{\alpha_1}{\alpha}\mathrm{E}[\mathcal{S}_1] + \frac{\alpha_2}{\alpha}\mathrm{E}[\mathcal{S}_2]$. Summing

Equation (25) for $i = 1, 2$, we observe that

$$\begin{aligned} \mathrm{E}[X_1] + \mathrm{E}[X_2] &= \alpha_1 \mathrm{E}[N_W]\mathrm{E}[\mathcal{S}_1] + \alpha_2 \mathrm{E}[N_W]\mathrm{E}[\mathcal{S}_2] \\ &= \alpha \mathrm{E}[N_W] \left(\frac{\alpha_1}{\alpha} \mathrm{E}[\mathcal{S}_1] + \frac{\alpha_2}{\alpha} \mathrm{E}[\mathcal{S}_2] \right) \\ &= \alpha \mathrm{E}[N_W]\mathrm{E}[\mathcal{S}], \end{aligned} \tag{26}$$

which can also be obtained through Little's Law by treating the total collection of failed machines as a single subsystem. The advantage of these formulas is that it produces a quicker way to calculate the mean sojourn times, as the mean queue lengths and $\mathrm{E}[N_W]$ only require calculation of the steady-state probabilities and avoids inverting the large rate matrices $\mathcal{R}_1$ and $\mathcal{R}_2$ in Equation (24). Equation (26) leads us to our third theorem.

Theorem 3 *For a maintenance system with $[C, f]$ machines and a given failure rate α, $\mathrm{E}[N_W]$ will simultaneously be maximized while $\mathrm{E}[\mathcal{S}]$ is minimized if $f = 0$.*

Proof: Recall Equation (15), which when $f = 0$ simplifies to

$$\mathrm{E}[N_W] = \mathrm{E}[C - X_1 - X_2] = C - \mathrm{E}[X_1] - \mathrm{E}[X_2]. \tag{27}$$

Subtracting both sides of Equation (26) from C, applying Equation (27), and isolating for $\mathrm{E}[N_W]$, we find

$$\mathrm{E}[N_W] = \frac{C}{1 + \alpha \mathrm{E}[\mathcal{S}]}. \tag{28}$$

While Equation (28) is not a linear relationship, it is clear that the selection of a service policy that maximizes $\mathrm{E}[N_W]$ for a given C and α must simultaneously minimize $\mathrm{E}[\mathcal{S}]$.

□

Remark 4 Equations (28) and (42) provide an alternate formula for the aggregate rate at which machines fail and are repaired when $f = 0$, namely

$$\lambda_r^{[C,0]} = \frac{C}{\frac{1}{\alpha} + \mathrm{E}[\mathcal{S}^{[C,0]}]}. \tag{29}$$

The denominator is equal to the sum of the mean time it takes a working machine to fail and the expected time until it is working again after suffering an arbitrary failure (in a $[C, 0]$ system), and hence is equivalent to the expected time between repairs for a given machine (since there is no maintenance float, a repaired machine is put back to work immediately after it is repaired). The inverse of the time between repairs is the rate of repairs for a single machine, which when multiplied by C, results in the aggregate rate of repairs for the entire system.

Remark 5 If $f \geq 1$, we can obtain an alternative relationship than Equation (28) between $\mathrm{E}[N_W]$ and $\mathrm{E}[\mathcal{S}]$. Subtracting Equation (26) from $2C + f$ and applying the fact that for any two random variables X and Y, $\mathrm{E}[\min\{X, Y\}] + \mathrm{E}[\max\{X, Y\}] = \mathrm{E}[X] + \mathrm{E}[Y]$, we obtain

$$\begin{aligned} 2C + f - \alpha \mathrm{E}[N_W]\mathrm{E}[\mathcal{S}] &= 2C + f - \mathrm{E}[X_1] - \mathrm{E}[X_2] \\ &= \mathrm{E}[N_W] + \mathrm{E}[\max\{C, C + f - X_1 - X_2\}], \end{aligned}$$

which if we rearrange for $\mathrm{E}[N_W]$,

$$\begin{aligned} \mathrm{E}[N_W] &= \frac{2C + f - \mathrm{E}[\max\{C, C + f - X_1 - X_2\}]}{1 + \alpha \mathrm{E}[\mathcal{S}]} \\ &= \frac{C + f - \mathrm{E}[\max\{0, f - X_1 - X_2\}]}{1 + \alpha \mathrm{E}[\mathcal{S}]}, \end{aligned} \tag{30}$$

where $\mathrm{E}[\max\{0, f - X_1 - X_2\}]$ is the expected number of functional machines in the maintenance float. Unlike in Equation (28) where $\mathrm{E}[N_W]$ and $\mathrm{E}[\mathcal{S}]$ were the only "variable" components such that one must be maximized when the other is minimized, $\mathrm{E}[\max\{0, f - X_1 - X_2\}]$ will also change if we adjust model parameters or service policies and as such, the simultaneous optimization of $\mathrm{E}[N_W]$ and $\mathrm{E}[\mathcal{S}]$ is not guaranteed.

Remark 6 From Equations (30) and (42), we find an alternate equation for the aggregate rate at which machines fail and are repaired to be

$$\lambda_r^{[C,f]} = \frac{C + f - \mathrm{E}[\max\{0, f - X_1^{[C,f]} - X_2^{[C,f]}\}]}{\frac{1}{\alpha} + \mathrm{E}[\mathcal{S}^{[C,f]}]}, \tag{31}$$

where the numerator is the expected number of machines in the maintenance system that are in the process of failing (i.e., in use) or the process of being repaired (i.e., are receiving service or are waiting in a queue), and the denominator is the expected amount of time for a machine to fail and then be repaired, agreeing with the intended interpretation of $\lambda_r^{[C,f]}$. Unsurprisingly, Equation (31) reduces to Equation (29) if we let $f = 0$.

We demonstrate the simultaneous and non-simultaneous optimizations of $\mathrm{E}[N_W]$ and $\mathrm{E}[\mathcal{S}]$ by plotting them over all possible (a, b) threshold policies for the $[8, 6]$ and $[14, 0]$ systems (with switch-ins) considered in Table 2. For both figures, grey dashed vertical lines are presented to visually separate the (a, b) threshold policies according to values of a. All (a, b) threshold policies are plotted as grey dots by default, while we reuse the symbols from Figures 4 and 5 for the cases that replicate exhaustive (i.e., $(14, 14)$ threshold) or standard class-1 priority policies (i.e., $(1, 1)$ and $(1, 14)$ threshold). Additionally, the (a, b) threshold policies that maximizes $\mathrm{E}[N_W]$ and/or minimize $\mathrm{E}[\mathcal{S}]$ are plotted as black dots. Finally, black dashed lines are provided on the optimal policies on their corresponding optimal plots for even further visual contrast and to point to their policy on the horizontal axis.

In Figure 6, we examine the case of $[C, f] = [8, 6]$, and begin by also plotting the class-1 and class-2 mean sojourn times, $\mathrm{E}[\mathcal{S}_1]$ and $\mathrm{E}[\mathcal{S}_2]$. We observe that the two class-i expected

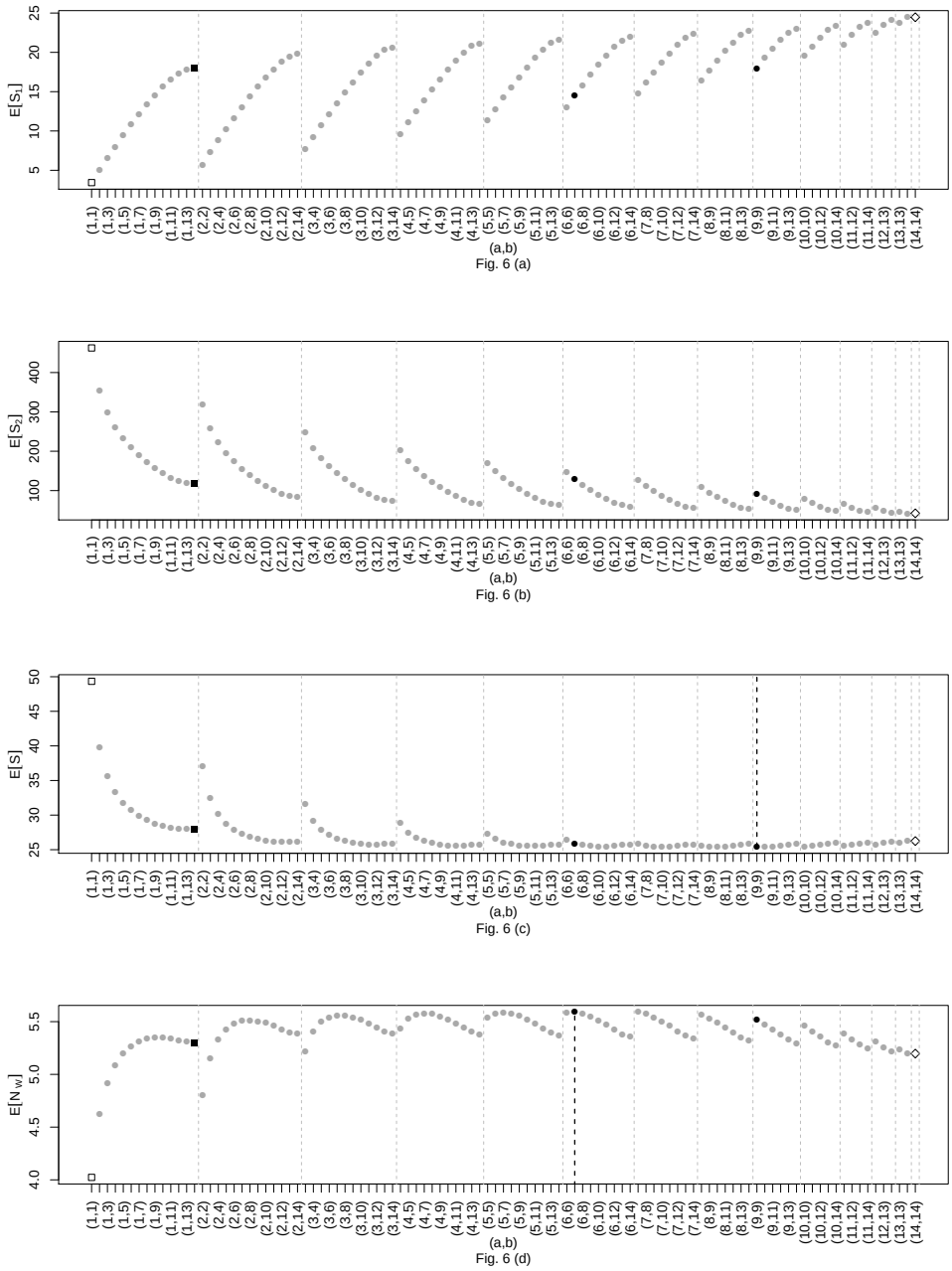

Figure 6: Plots of $\mathrm{E}[\mathcal{S}_1]$, $\mathrm{E}[\mathcal{S}_2]$, $\mathrm{E}[\mathcal{S}]$, and $\mathrm{E}[N_W]$ for all possible class-1 (a, b) threshold policies with $[C, f] = [8, 6]$, $\alpha = 0.05$, H_2 service, $p_{>0} = 1$, and $M_\mathrm{B} = M_\mathrm{S} = 1$.

sojourn times have opposite relationships with a and b. By increasing the value of a and/or b, the strength of the server's preference to serve class 1 before class 2 decreases, as the threshold priorities need larger class-1 queue lengths to activate. Therefore, it follows that

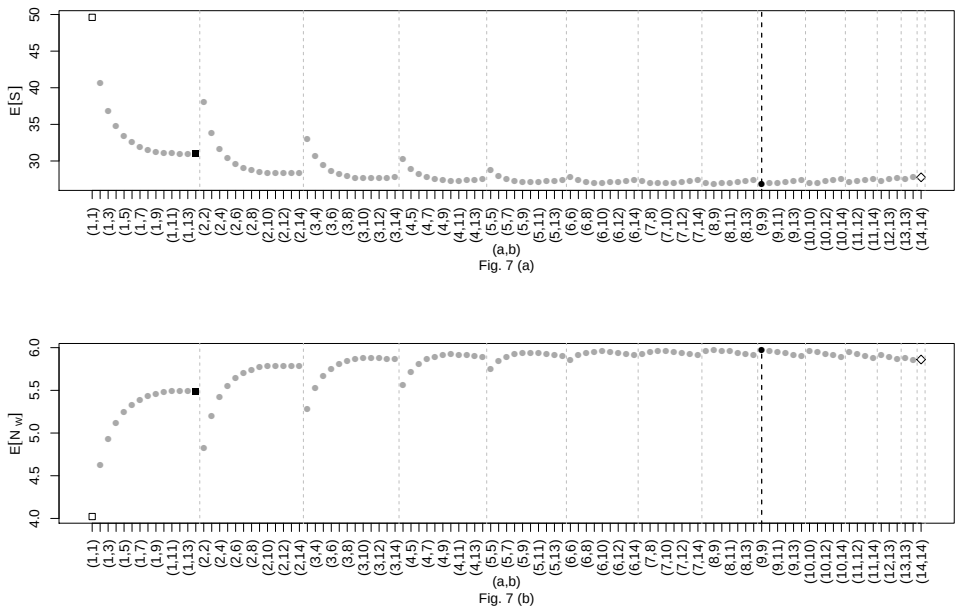

Figure 7: Plots of $\mathrm{E}[\mathcal{S}]$ and $\mathrm{E}[N_W]$ for all possible class-1 (a, b) threshold policies with $[C, f] = [14, 0]$, $\alpha = 0.05$, H_2 service, $p_{>0} = 1$, and $M_\mathrm{B} = M_\mathrm{S} = 1$.

increasing a and/or b increases (decreases) $\mathrm{E}[\mathcal{S}_1]$ ($\mathrm{E}[\mathcal{S}_2]$).

As the class-2 expected sojourn times are much larger than the class-1 expected sojourn times, it is not surprising to see in both figures that the overall mean sojourn times are largely decreasing with a and b, despite the low 10% mixing weight for class-2 failures, although it begins to increase as a function of a and b for large values of a as the benefit to class 2 for further increasing the thresholds diminishes. In some cases not presented here, it is also possible to see a pronounced concave relationship between $\mathrm{E}[\mathcal{S}]$ and b for small a when there are fewer total machines in the system, but the relation "flattens" for the higher b values as the number of machines (and hence the expected number of working machines) are increased.

The relationship between $\mathrm{E}[N_W]$ and the threshold boundaries is also clearly non-monotonic, as we observe a convex function of b in Figure 6 for low values of a before becoming a decreasing function of b for larger a's. This convex relation is "flattened" for high b in Figure 7 (similar to the expected sojourn times as mentioned above) as we increase C at the cost of f which results in a net increase in $\mathrm{E}[N_W]$. As these are cases with switch-ins, $\mathrm{E}[N_W^{[\infty]}]$ is increasing in a and b since increasing the thresholds reduces extra switch-ins. Therefore, the decreasing relationship between $\mathrm{E}[N_W]$ and the thresholds for certain ranges of a and b is much less prominent in Figure 7 than Figure 6, as purely having working machines with no float results in the $[14, 0]$ cases being closer to their limit. In fact, this relationship should approach monotonic increasing in a and b as $C \to \infty$, and the exhaustive policy becomes

optimal having the fewest possible switch-ins and hence the highest $\mathrm{E}[N_W^{[\infty]}]$.

Finally, agreeing with the result of Theorem 3, we observe simultaneous optimization in the $[14, 0]$ case at $(a, b) = (9, 9)$, resulting in $\mathrm{E}[N_W] = 5.9709$ and $\mathrm{E}[\mathcal{S}] = 26.8941$. Also, the $[8, 6]$ case demonstrates Remark 5, where $\mathrm{E}[N_W]$ is maximized at $(a, b) = (6, 7)$ resulting in $\mathrm{E}[N_W] = 5.5911$ and $\mathrm{E}[\mathcal{S}] = 25.9373$, and $\mathrm{E}[\mathcal{S}]$ is minimized at $(9, 9)$ where $\mathrm{E}[N_W] = 5.5203$ and $\mathrm{E}[\mathcal{S}] = 25.4357$.

5 Numerical Examples

5.1 (a, b) Threshold Optimization

We now imagine a factory setting where an array of identical machines represent an important component of their production process. To avoid creating a production bottleneck at this step, it is of interest to maximize the average rate at which work is processed by maximizing the expected number of working machines. From Theorem 2, we know that there exists a limit $\mathrm{E}[N_W^{[\infty]}]$ dependent on the failure rates and mean service times, which can only be reached if there are no switch-in times. If cost was no object, then this limit could be reached using any service policy given an arbitrarily large C if there were no switch-in times (in fact, it would be advantageous to use class-1 preemptive resume priority which we have seen will reach $\mathrm{E}[N_W^{[\infty]}]$ at the smallest value of C). If there are switch-in times corresponding to set-up times for one or both classes, then the exhaustive service policy will have the highest peak service rate and hence the maximum $\mathrm{E}[N_W^{[\infty]}]$.

Unfortunately, increasing your machines would have a real cost related to initial investment (e.g., purchase price), recurring costs (e.g., fuel, replacement parts, operational staff), space constraints (e.g., storage space for spares, space on the factory floor for operational machines), and so on. Due to these costs, it may be optimal to invest in a C and f which do not reach the highest possible rate of output. If this is the case, a different policy than exhaustive may be optimal as they have different rates of convergence to the server's peak repair rate and hence could have a higher $\mathrm{E}[N_W^{[C,f]}]$ at a given C and f as seen in Figures 6 and 7.

With this motivation in mind, we introduce a basic cost function $\mathrm{E}[N_W^{[C,f]}] - r_C C - r_f f$, where r_C is the cost to purchase a machine and to increase the maximum capacity of working machines by one and r_f is the cost per additional machine purchased as a spare and the corresponding cost of storage. Here, we assume that r_C and r_f are normalized with respect to the profit per unit time that a working machine produces, so that maximizing the cost function maximizes the average profit per unit time. We aim to optimize with respect to C, f, and all possible class-1 (a, b) threshold policies for a given number of machines (i.e., $1 \leq a \leq b \leq C + f$).

For the purposes of our example, we consider a factory with space for a total of $C + f = 14$ machines. We allow $\alpha \in \{0.05, 0.075, 0.10\}$, $M_{\mathrm{B}} \in \{0.5, 1\}$, $M_{\mathrm{S}} \in \{1, 2\}$, and $p_{>0} \in \{0, 0.5, 1\}$, while we assume that the switch-in distributions are of the kind defined in Equations (20)-(23) in Section 4.1. Along with the H_2 service time distributions outlined in

Equations (18) and (19), we also consider Erlang-3 distributions (henceforth referred to as E_3 service) having initial probability row vectors

$$\underline{\beta}_1 = \underline{\beta}_2 = (1, 0, 0), \tag{32}$$

and rate matrices

$$B_1 = \begin{pmatrix} -3 & 3 & 0 \\ 0 & -3 & 3 \\ 0 & 0 & -3 \end{pmatrix}, \; B_2 = \frac{1}{20M_{\mathrm{B}}} \begin{pmatrix} -3 & 3 & 0 \\ 0 & -3 & 3 \\ 0 & 0 & -3 \end{pmatrix}, \tag{33}$$

resulting in the same means as the H_2 service time distributions and maintaining the same interpretation of M_{B}. The E_3 distributions act as good examples of distributions which may be preempted and have a residual service time after the server's return that is less than if they had to restart their work.

We begin by considering optimization when $r_C = r_f$ (i.e., when every machine costs the same whether it will increase the system's capacity or act as a spare in the maintenance float). Table 3 contains the optimal $[C, f]$, (a, b), and $\mathrm{E}[N_W^{[C,f]}]$ (with suppressed superscripts) at $r_C = r_f = 0.10$ over the combinations of parameters and service time distributions outlined above. The first observation that stands out is that in no cases is it optimal to have $f \geq 1$, even when $C < 14$ allows for more additional machines before hitting the cap. Recalling Theorem 1, we have that $\mathrm{E}[N_W^{[k,0]}] > \mathrm{E}[N_W^{[k-1,1]}]$. We have also seen that for the exhaustive service policy example in Figure 2, $\mathrm{E}[N_W^{[k-f,f]}]$ is a decreasing function of f (although $\mathrm{E}[N_W^{[C,f]}]$ is an increasing function of f). Therefore, it makes sense in this case to never invest in a maintenance float if putting all machines towards C maximizes $\mathrm{E}[N_W^{[C,f]}]$ for a given $C + f$, when $r_C = r_f$ reduces the cost function to $\mathrm{E}[N_W^{[C,f]}] - r_C(C + f)$. Thus, it is only ever of financial interest to invest in a maintenance float if it is cheaper to add a spare machine to the system than it is to increase C (i.e., $r_C > r_f$). This brings us to our next result.

Theorem 4 *Under cost function* $\mathrm{E}[N_W] - r_C C - r_f f$*, if* $r_C > r_f$*, then for a system with* k *total machines,* $k = 2, 3, \ldots$*, it will be suboptimal to not use a maintenance float if*

$$\mathrm{E}[N_W^{[k,0]}] < k(r_C - r_f). \tag{34}$$

Proof Recall from the proof of Theorem 1, $\mathrm{E}[N_W^{[k,0]}] = c_k \mathrm{E}[N_W^{[k-1,1]}]$, where $1 < c_k < \frac{k}{k-1}$, $k = 2, 3, \ldots$. It will be suboptimal to select $f = 0$ in a system having k total machines if

$$\mathrm{E}[N_W^{[k-1,1]}] - (k-1)r_C - r_f > \mathrm{E}[N_W^{[k,0]}] - kr_C,$$

or equivalently,

$$\mathrm{E}[N_W^{[k,0]}] - \mathrm{E}[N_W^{[k-1,1]}] < r_C - r_f.$$

We observe that

$$\mathrm{E}[N_W^{[k,0]}] - \mathrm{E}[N_W^{[k-1,1]}] = \mathrm{E}[N_W^{[k,0]}](1 - c_k^{-1}) < \mathrm{E}[N_W^{[k,0]}] \left(1 - \frac{k-1}{k}\right) = \frac{1}{k}\mathrm{E}[N_W^{[k,0]}].$$

Table 3: Optimal C, f, a, and b, under H_2 and E_3 service for equal machine costs, $\underline{r} = (r_C, r_f) = (0.10, 0.10)$.

H_2 service			$p_{>0}$ = 0			$p_{>0}$ = 0.5			$p_{>0}$ = 1		
M_B	M_S	α	$[C, f]$	(a, b)	$E[N_W]$	$[C, f]$	(a, b)	$E[N_W]$	$[C, f]$	(a, b)	$E[N_W]$
1	1	0.05	$[13, 0]$	$(1, 1)$	6.6948	$[14, 0]$	$(7, 7)$	6.2251	$[14, 0]$	$(9, 9)$	5.9709
		0.075	$[10, 0]$	$(1, 1)$	4.4827	$[11, 0]$	$(6, 7)$	4.1487	$[11, 0]$	$(8, 8)$	3.9681
		0.10	$[8, 0]$	$(1, 1)$	3.3439	$[8, 0]$	$(5, 5)$	2.9854	$[9, 0]$	$(8, 8)$	2.9386
	2	0.05	$[13, 0]$	$(1, 1)$	6.6948	$[14, 0]$	$(9, 9)$	5.9549	$[14, 0]$	$(11, 11)$	5.5953
		0.075	$[10, 0]$	$(1, 1)$	4.4827	$[11, 0]$	$(8, 8)$	3.9548	$[12, 0]$	$(11, 12)$	3.8182
		0.10	$[8, 0]$	$(1, 1)$	3.3439	$[9, 0]$	$(7, 7)$	2.9251	$[9, 0]$	$(8, 9)$	2.7411
0.5	1	0.05	$[14, 0]$	$(1, 1)$	9.4562	$[14, 0]$	$(5, 6)$	8.5465	$[14, 0]$	$(8, 8)$	8.1463
		0.075	$[13, 0]$	$(1, 1)$	6.8193	$[14, 0]$	$(9, 9)$	6.3171	$[14, 0]$	$(11, 11)$	6.0460
		0.10	$[11, 0]$	$(1, 1)$	5.1484	$[12, 0]$	$(9, 9)$	4.7611	$[13, 0]$	$(12, 12)$	4.6616
	2	0.05	$[14, 0]$	$(1, 1)$	9.4562	$[14, 0]$	$(8, 8)$	8.0974	$[14, 0]$	$(11, 11)$	7.5325
		0.075	$[13, 0]$	$(1, 1)$	6.8193	$[14, 0]$	$(11, 11)$	6.0146	$[14, 0]$	$(13, 14)$	5.6166
		0.10	$[11, 0]$	$(1, 1)$	5.1484	$[13, 0]$	$(12, 12)$	4.6391	$[14, 0]$	$(13, 14)$	4.4440
E_3 service											
1	1	0.05	$[11, 0]$	$(1, 1)$	6.7993	$[13, 0]$	$(7, 8)$	6.3602	$[14, 0]$	$(10, 11)$	6.1970
		0.075	$[8, 0]$	$(1, 1)$	4.5144	$[10, 0]$	$(7, 7)$	4.2129	$[11, 0]$	$(9, 10)$	4.1143
		0.10	$[6, 0]$	$(1, 1)$	3.3179	$[8, 0]$	$(5, 6)$	3.1112	$[9, 0]$	$(8, 9)$	3.0543
	2	0.05	$[11, 0]$	$(1, 1)$	6.7993	$[14, 0]$	$(10, 10)$	6.1884	$[14, 0]$	$(12, 14)$	5.8122
		0.075	$[8, 0]$	$(1, 1)$	4.5144	$[11, 0]$	$(9, 10)$	4.1065	$[12, 0]$	$(11, 12)$	3.9632
		0.10	$[6, 0]$	$(1, 1)$	3.3179	$[9, 0]$	$(8, 9)$	3.0466	$[9, 0]$	$(8, 9)$	2.8523
0.5	1	0.05	$[14, 0]$	$(1, 1)$	10.2076	$[14, 0]$	$(5, 7)$	9.0983	$[14, 0]$	$(8, 10)$	8.6314
		0.075	$[11, 0]$	$(1, 1)$	6.9260	$[14, 0]$	$(11, 14)$	6.6001	$[14, 0]$	$(13, 14)$	6.3492
		0.10	$[9, 0]$	$(1, 1)$	5.2021	$[11, 0]$	$(9, 11)$	4.8821	$[12, 0]$	$(11, 12)$	4.7810
	2	0.05	$[14, 0]$	$(1, 1)$	10.2076	$[14, 0]$	$(7, 11)$	8.5985	$[14, 0]$	$(12, 14)$	7.9602
		0.075	$[11, 0]$	$(1, 1)$	6.9260	$[14, 0]$	$(13, 14)$	6.3362	$[14, 0]$	$(13, 14)$	5.9164
		0.10	$[9, 0]$	$(1, 1)$	5.2021	$[12, 0]$	$(11, 12)$	4.7730	$[14, 0]$	$(13, 14)$	4.6623

Thus, if Equation (34) holds, then

$$r_C - r_f > \frac{1}{k}\mathrm{E}[N_W^{[k,0]}] > \mathrm{E}[N_W^{[k,0]}] - \mathrm{E}[N_W^{[k-1,1]}],$$

and so it would be suboptimal to select $f = 0$ under the given cost function.

□

Note that it may be optimal to use a float when the inequality of Theorem 4 does not hold (i.e., for small k), so long as $r_C > r_f$ is still true. Theorem 4 simply provides an inequality that, if it holds, guarantees that $f = 0$ is not optimal at k total machines. By Theorem 2, we know that $\lim_{C\to\infty} \mathrm{E}[N_W^{[C,0]}]$ has a finite upper bound, which implies that there must exist a $k \in \mathbb{Z}^+$ such that Equation (34) is satisfied if $r_C > r_f$. Additionally, this implies that if we increase r_C for a given $k \geq 2$ and r_f, we will eventually reach a point for sure where it becomes optimal to use a maintenance float.

Other conclusions also follow from Table 3. In particular, when $p_{>0} = 0$, it is optimal to use class-1 preemptive resume priority, as it reaches its $\mathrm{E}[N_W^{[\infty]}]$ at the smallest value of C out of the considered policies, and this limit is not penalized by its additional switches due to identically zero switch-in times. We observe that some optimal C are less than 14, corresponding to situations where there is no (a, b) which would result in $\mathrm{E}[N_W^{[C+1,0]}]$ that is at least $r_C = 0.10$ greater than the optimal $\mathrm{E}[N_W^{[C,0]}]$. In fact, the E_3 distribution in contrast to H_2 often results in a smaller optimal C while simultaneously allowing a larger $\mathrm{E}[N_W^{[C,0]}]$, an advantage of having a lower service time variance and a type of partitioned sequential work which benefits more from the nature of the preemptive resume priorities (whereas H_2 simply remembers which exponential distribution from the mixture the job belonged to).

Decreasing M_{B} results in faster class-2 services, thereby increasing $\mathrm{E}[Z^m]^{-1}$ (and hence $\mathrm{E}[N_W^{[\infty]}]$). The higher limit requires more total machines to reach it, and hence increments in optimal $\mathrm{E}[N_W^{[C,0]}]$ (i.e., at optimal (a, b) threshold cases for given C) will outweigh the cost of an additional machine until larger values of C. This results in larger optimal C and $\mathrm{E}[N_W^{[C,0]}]$. Increasing α has the opposite effect on $\mathrm{E}[N_W^{[\infty]}]$, resulting in a lower limit and hence lower optimal C values. Increasing M_{S} for a given $p_{>0} > 0$ or increasing $p_{>0}$ for a given M_{S} (or increasing both) causes switch-ins to be more costly, penalizing a priority policy inversely proportional to a and b, while lowering $\mathrm{E}[N_W^{[\infty]}]$. This has the effect of increasing optimal threshold limits while lowering the optimal $\mathrm{E}[N_W^{[C,0]}]$.

Next, we allow increasing float machines to cost half as much as increasing the system capacity (i.e., $r_f = r_C/2$) and consider a range of costs $r_C \in \{0.05, 0.10, 0.25\}$ for H_2 service in Table 4 and E_3 service in Table 5. Comparing the $r_C = 0.10$ cases from these two tables to Table 3, it is possible (for the cases that did not already select $C = 14$) to increase the maintenance float by 2 machines for the cost of 1 capacity slot, which allows those cases to "afford" to increase the total number of machines. While a single increase in C is worth more than a single increase in f, the benefit of multiple spares can outweigh that of a single

capacity machine when C is already a few machines over $\mathrm{E}[N_W^{[\infty]}]$. This follows since the fraction of time that the system spends near capacity decreases as C becomes large and so the marginal benefit of increasing C over f will reduce, and increasing f multiple times can outweigh a unit increase of C when it is larger than $\mathrm{E}[N_W^{[\infty]}]$ (recall that increasing f cannot result in N_W surpassing C, and so the benefit of increasing C over f is much larger for $C < \mathrm{E}[N_W^{[\infty]}]$). We also observe some cases where $C = 14$ in Table 3 but some capacity is diverted to the float, slightly decreasing $\mathrm{E}[N_W^{[C,f]}]$ but saving much more in costs.

Our earlier observations concerning parameters M_{B}, M_{S}, α, and $p_{>0}$ clearly still hold true in Tables 4 and 5. Also, we still observe E_3 service achieving higher $\mathrm{E}[N_W^{[C,f]}]$ while typically selecting optimal C that are no larger than those selected for H_2 service, with the exception of the $r_C = 0.25$, $M_{\mathrm{B}} = 0.5$, $M_{\mathrm{S}} = 2$, $\alpha = 0.10$ case where the faster rate at which $\mathrm{E}[N_W^{[C,f]}]$ approaches its limit for E_3 allows it to "afford" to increase C longer than H_2 service. Finally, we observe that by increasing the cost per machine, the system will want to optimize at fewer machines as the incremental costs will begin to outpace the increases in $\mathrm{E}[N_W^{[C,f]}]$ at fewer machines. When optimizing the (a, b) threshold at fewer machines, the decreases in peak repair rate caused by extra incurred switch-ins are smaller due to fewer observed failures, and so the optimal a and b are non-increasing in r_C (and r_f).

5.2 Class-1 Sojourn Time Densities For (a, a) Threshold Policies

In Section 3.3, we derived the distribution for a class-1 machine's sojourn time, $\mathcal{S}_1$, to be $\mathrm{PH}(\underline{\Phi}_1, \mathcal{R}_1)$, resulting in the pdf

$$f_{\mathcal{S}_1}(t) = \underline{\Phi}_1 \exp\{\mathcal{R}_1 t\} \underline{R}'_{0,1}. \tag{35}$$

As an illustration, we plot some of these densities for a family of service policies. For the sake of brevity, we constrain ourselves to the set of (a, a) threshold policies (i.e., preemptive resume threshold policies), which exhibited notable sensitivities to the selection of threshold parameter a, much more so than the $(a, C + f)$ threshold policies (i.e., non-preemptive threshold policies) in the numerical cases we considered. As computing the matrix exponential function in Equation (35) can be quite time consuming for systems with large state spaces, we consider only exponential service time distributions within this example (having the typical means of 1 and $20M_{\mathrm{B}}$ for class 1 and class 2, respectively), along with the modest number of machines $C = 8$ and $f = 2$. We still elect to use the switch-in time distributions defined in Equations (20)-(23), as the size of the state space is less sensitive to the number of switching phases, as they are not always tracked like service phases are by Y_1 and Y_2.

In Figure 8, we plot $f_{\mathcal{S}_1}(t)$ for $t \in [0, 15]$ and $a = 1, 2, \ldots, 10$, letting $\alpha = 0.05$ and $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$. We consider both $p_{>0} = 0$ and $p_{>0} = 1$ to visualize the impact of switch-in times. Upon first inspection, it is clearly evident that the densities differ greatly for low values of a, when class 1's relative priority to class 2 is at its highest, while its shape is more consistent at higher values of a, requiring larger queue lengths (which are rarer to observe) and hence reducing the threshold's impact. Unsurprisingly, as we are considering class-1 sojourn times, the lower threshold policies result in more density towards small sojourn times

Table 4: Optimal C, f, a, and b, under H_2 service and cheaper reserve machines ($r_f = r_C/2$).

$\underline{r} = (0.05, 0.025)$			$p_{>0}$ = 0			$p_{>0}$ = 0.5			$p_{>0}$ = 1		
M_B	M_S	α	$[C, f]$	(a, b)	$E[N_W]$	$[C, f]$	(a, b)	$E[N_W]$	$[C, f]$	(a, b)	$E[N_W]$
1	1	0.05	$[12, 2]$	$(1, 1)$	6.7511	$[13, 1]$	$(7, 7)$	6.2110	$[13, 1]$	$(9, 9)$	5.9579
		0.075	$[9, 3]$	$(1, 1)$	4.5526	$[9, 5]$	$(8, 8)$	4.2851	$[10, 4]$	$(11, 11)$	4.1606
		0.10	$[7, 3]$	$(1, 1)$	3.4093	$[7, 7]$	$(9, 9)$	3.2572	$[8, 6]$	$(11, 12)$	3.1780
	2	0.05	$[12, 2]$	$(1, 1)$	6.7511	$[13, 1]$	$(9, 9)$	5.9411	$[13, 1]$	$(11, 11)$	5.5850
		0.075	$[9, 3]$	$(1, 1)$	4.5526	$[10, 4]$	$(10, 11)$	4.1517	$[11, 3]$	$(13, 14)$	3.9622
		0.10	$[7, 3]$	$(1, 1)$	3.4093	$[8, 6]$	$(11, 11)$	3.1728	$[9, 5]$	$(13, 14)$	3.0523
0.5	1	0.05	$[14, 0]$	$(1, 1)$	9.4562	$[14, 0]$	$(5, 6)$	8.5465	$[14, 0]$	$(8, 8)$	8.1463
		0.075	$[12, 2]$	$(1, 1)$	6.8780	$[13, 1]$	$(9, 9)$	6.3040	$[12, 2]$	$(11, 11)$	6.0114
		0.10	$[9, 4]$	$(1, 1)$	5.1964	$[11, 3]$	$(11, 11)$	4.8967	$[11, 3]$	$(13, 13)$	4.7217
	2	0.05	$[14, 0]$	$(1, 1)$	9.4562	$[14, 0]$	$(8, 8)$	8.0974	$[14, 0]$	$(11, 11)$	7.5325
		0.075	$[12, 2]$	$(1, 1)$	6.8780	$[13, 1]$	$(11, 11)$	6.0035	$[12, 2]$	$(13, 14)$	5.5904
		0.10	$[9, 4]$	$(1, 1)$	5.1964	$[11, 3]$	$(13, 13)$	4.7008	$[11, 3]$	$(13, 14)$	4.4221
$\underline{r} = (0.10, 0.05)$											
1	1	0.05	$[11, 3]$	$(1, 1)$	6.7098	$[11, 3]$	$(6, 6)$	6.1403	$[11, 3]$	$(8, 8)$	5.8850
		0.075	$[8, 3]$	$(1, 1)$	4.4901	$[8, 5]$	$(7, 7)$	4.2039	$[8, 6]$	$(9, 9)$	4.0882
		0.10	$[6, 3]$	$(1, 1)$	3.3414	$[6, 5]$	$(6, 6)$	3.1177	$[6, 5]$	$(8, 8)$	2.9792
	2	0.05	$[11, 3]$	$(1, 1)$	6.7098	$[11, 3]$	$(8, 8)$	5.8708	$[11, 3]$	$(11, 11)$	5.5148
		0.075	$[8, 3]$	$(1, 1)$	4.4901	$[8, 6]$	$(9, 9)$	4.0810	$[9, 5]$	$(12, 12)$	3.8996
		0.10	$[6, 3]$	$(1, 1)$	3.3414	$[6, 5]$	$(7, 8)$	2.9720	$[7, 5]$	$(11, 11)$	2.8933
0.5	1	0.05	$[14, 0]$	$(1, 1)$	9.4562	$[14, 0]$	$(5, 6)$	8.5465	$[13, 1]$	$(8, 8)$	8.1062
		0.075	$[11, 3]$	$(1, 1)$	6.8373	$[11, 3]$	$(8, 8)$	6.2307	$[11, 3]$	$(11, 11)$	5.9649
		0.10	$[9, 3]$	$(1, 1)$	5.1613	$[9, 5]$	$(9, 9)$	4.8288	$[9, 5]$	$(12, 12)$	4.6434
	2	0.05	$[14, 0]$	$(1, 1)$	9.4562	$[13, 1]$	$(7, 8)$	8.0555	$[13, 1]$	$(11, 11)$	7.5060
		0.075	$[11, 3]$	$(1, 1)$	6.8373	$[11, 3]$	$(10, 10)$	5.9331	$[11, 3]$	$(13, 13)$	5.5533
		0.10	$[9, 3]$	$(1, 1)$	5.1613	$[9, 5]$	$(11, 11)$	4.6228	$[9, 5]$	$(13, 14)$	4.3509
$\underline{r} = (0.25, 0.125)$											
1	1	0.05	$[9, 2]$	$(1, 1)$	6.2249	$[9, 3]$	$(3, 5)$	5.7651	$[9, 3]$	$(6, 6)$	5.4992
		0.075	$[6, 2]$	$(1, 1)$	4.0503	$[6, 2]$	$(1, 4)$	3.6371	$[6, 2]$	$(5, 5)$	3.4244
		0.10	$[5, 1]$	$(1, 1)$	2.9977	$[4, 2]$	$(1, 3)$	2.5299	$[4, 2]$	$(3, 4)$	2.3718
	2	0.05	$[9, 2]$	$(1, 1)$	6.2249	$[8, 4]$	$(4, 6)$	5.3665	$[8, 4]$	$(7, 8)$	5.0121
		0.075	$[6, 2]$	$(1, 1)$	4.0503	$[6, 2]$	$(1, 5)$	3.4294	$[5, 3]$	$(5, 6)$	3.0341
		0.10	$[5, 1]$	$(1, 1)$	2.9977	$[4, 2]$	$(1, 4)$	2.3846	$[4, 1]$	$(4, 5)$	2.0562
0.5	1	0.05	$[13, 1]$	$(1, 1)$	9.3829	$[12, 2]$	$(3, 6)$	8.3904	$[12, 2]$	$(8, 8)$	8.0157
		0.075	$[10, 2]$	$(1, 1)$	6.5878	$[9, 3]$	$(4, 6)$	5.8294	$[9, 4]$	$(8, 9)$	5.6796
		0.10	$[7, 2]$	$(1, 1)$	4.7393	$[7, 2]$	$(2, 5)$	4.2149	$[7, 3]$	$(7, 7)$	4.1154
	2	0.05	$[13, 1]$	$(1, 1)$	9.3829	$[12, 2]$	$(5, 8)$	7.9720	$[11, 3]$	$(10, 10)$	7.3296
		0.075	$[10, 2]$	$(1, 1)$	6.5878	$[9, 4]$	$(6, 9)$	5.6500	$[9, 4]$	$(11, 11)$	5.2731
		0.10	$[7, 2]$	$(1, 1)$	4.7393	$[6, 3]$	$(3, 6)$	3.8452	$[6, 3]$	$(8, 8)$	3.5430

Table 5: Optimal C, f, a, and b, under E_3 service and cheaper reserve machines ($r_f = r_C/2$).

$\underline{r} = (0.05, 0.025)$			$p_{>0}$ = 0			$p_{>0}$ = 0.5			$p_{>0}$ = 1		
M_B	M_S	α	$[C, f]$	(a, b)	$\mathrm{E}[N_W]$	$[C, f]$	(a, b)	$\mathrm{E}[N_W]$	$[C, f]$	(a, b)	$\mathrm{E}[N_W]$
1	1	0.05	$[9, 4]$	$(1, 1)$	6.8564	$[11, 3]$	$(8, 8)$	6.4220	$[12, 2]$	$(10, 10)$	6.1761
		0.075	$[7, 2]$	$(1, 1)$	4.5630	$[8, 6]$	$(10, 10)$	4.3862	$[9, 5]$	$(11, 12)$	4.2653
		0.10	$[5, 3]$	$(1, 1)$	3.4224	$[6, 6]$	$(9, 9)$	3.2713	$[7, 7]$	$(12, 13)$	3.2401
	2	0.05	$[9, 4]$	$(1, 1)$	6.8564	$[12, 2]$	$(10, 10)$	6.1678	$[12, 2]$	$(11, 13)$	5.7829
		0.075	$[7, 2]$	$(1, 1)$	4.5630	$[9, 5]$	$(11, 12)$	4.2624	$[10, 4]$	$(13, 14)$	4.0810
		0.10	$[5, 3]$	$(1, 1)$	3.4224	$[8, 6]$	$(12, 13)$	3.2635	$[9, 5]$	$(13, 14)$	3.1579
0.5	1	0.05	$[13, 1]$	$(1, 1)$	10.1838	$[14, 0]$	$(5, 7)$	9.0983	$[14, 0]$	$(8, 10)$	8.6314
		0.075	$[9, 4]$	$(1, 1)$	6.9852	$[12, 2]$	$(10, 13)$	6.5812	$[12, 2]$	$(13, 14)$	6.3296
		0.10	$[7, 4]$	$(1, 1)$	5.2438	$[10, 4]$	$(13, 14)$	5.0609	$[10, 4]$	$(13, 14)$	4.9010
	2	0.05	$[13, 1]$	$(1, 1)$	10.1838	$[14, 0]$	$(7, 11)$	8.5985	$[13, 1]$	$(12, 14)$	7.9388
		0.075	$[9, 4]$	$(1, 1)$	6.9852	$[12, 2]$	$(13, 14)$	6.3152	$[12, 2]$	$(13, 14)$	5.9028
		0.10	$[7, 4]$	$(1, 1)$	5.2438	$[10, 4]$	$(13, 14)$	4.9054	$[10, 4]$	$(13, 14)$	4.6350
$\underline{r} = (0.10, 0.05)$											
1	1	0.05	$[9, 3]$	$(1, 1)$	6.8209	$[9, 5]$	$(7, 7)$	6.3457	$[10, 4]$	$(9, 9)$	6.1118
		0.075	$[6, 3]$	$(1, 1)$	4.5221	$[6, 6]$	$(7, 7)$	4.2330	$[7, 6]$	$(9, 10)$	4.1480
		0.10	$[5, 2]$	$(1, 1)$	3.3846	$[5, 5]$	$(6, 6)$	3.1620	$[6, 5]$	$(9, 9)$	3.1047
	2	0.05	$[9, 3]$	$(1, 1)$	6.8209	$[10, 4]$	$(9, 9)$	6.1043	$[11, 3]$	$(11, 12)$	5.7471
		0.075	$[6, 3]$	$(1, 1)$	4.5221	$[7, 6]$	$(9, 10)$	4.1440	$[8, 6]$	$(12, 13)$	3.9933
		0.10	$[5, 2]$	$(1, 1)$	3.3846	$[6, 5]$	$(8, 9)$	3.1012	$[7, 4]$	$(10, 11)$	2.9589
0.5	1	0.05	$[13, 1]$	$(1, 1)$	10.1838	$[13, 1]$	$(5, 7)$	9.0686	$[13, 1]$	$(7, 10)$	8.6039
		0.075	$[9, 3]$	$(1, 1)$	6.9509	$[10, 4]$	$(9, 10)$	6.5099	$[11, 3]$	$(12, 14)$	6.2959
		0.10	$[7, 3]$	$(1, 1)$	5.2185	$[8, 5]$	$(9, 11)$	4.9492	$[9, 5]$	$(13, 14)$	4.8775
	2	0.05	$[13, 1]$	$(1, 1)$	10.1838	$[13, 1]$	$(7, 10)$	8.5707	$[13, 1]$	$(12, 14)$	7.9388
		0.075	$[9, 3]$	$(1, 1)$	6.9509	$[11, 3]$	$(12, 14)$	6.2804	$[11, 3]$	$(13, 14)$	5.8745
		0.10	$[7, 3]$	$(1, 1)$	5.2185	$[9, 5]$	$(13, 14)$	4.8714	$[9, 5]$	$(13, 14)$	4.5977
$\underline{r} = (0.25, 0.125)$											
1	1	0.05	$[8, 3]$	$(1, 1)$	6.6525	$[8, 4]$	$(5, 5)$	6.0695	$[8, 5]$	$(7, 7)$	5.8679
		0.075	$[6, 2]$	$(1, 1)$	4.4527	$[5, 4]$	$(4, 4)$	3.8974	$[5, 4]$	$(5, 5)$	3.6552
		0.10	$[4, 2]$	$(1, 1)$	3.1785	$[4, 2]$	$(1, 3)$	2.7359	$[4, 2]$	$(4, 4)$	2.5211
	2	0.05	$[8, 3]$	$(1, 1)$	6.6525	$[8, 5]$	$(7, 7)$	5.8587	$[8, 5]$	$(9, 9)$	5.4209
		0.075	$[6, 2]$	$(1, 1)$	4.4527	$[5, 3]$	$(1, 5)$	3.5254	$[5, 4]$	$(6, 7)$	3.3174
		0.10	$[4, 2]$	$(1, 1)$	3.1785	$[4, 2]$	$(1, 4)$	2.5387	$[4, 2]$	$(5, 6)$	2.2780
0.5	1	0.05	$[12, 2]$	$(1, 1)$	10.1016	$[11, 3]$	$(4, 6)$	8.8776	$[11, 3]$	$(6, 8)$	8.4141
		0.075	$[8, 3]$	$(1, 1)$	6.7758	$[8, 4]$	$(6, 6)$	6.1325	$[8, 5]$	$(8, 9)$	5.9258
		0.10	$[6, 2]$	$(1, 1)$	4.9274	$[6, 3]$	$(4, 5)$	4.4154	$[7, 3]$	$(7, 10)$	4.3951
	2	0.05	$[12, 2]$	$(1, 1)$	10.1016	$[11, 3]$	$(5, 9)$	8.3872	$[11, 3]$	$(10, 14)$	7.7659
		0.075	$[8, 3]$	$(1, 1)$	6.7758	$[8, 5]$	$(7, 9)$	5.9073	$[9, 4]$	$(12, 13)$	5.5994
		0.10	$[6, 2]$	$(1, 1)$	4.9274	$[6, 4]$	$(5, 8)$	4.2553	$[7, 3]$	$(9, 10)$	4.0371

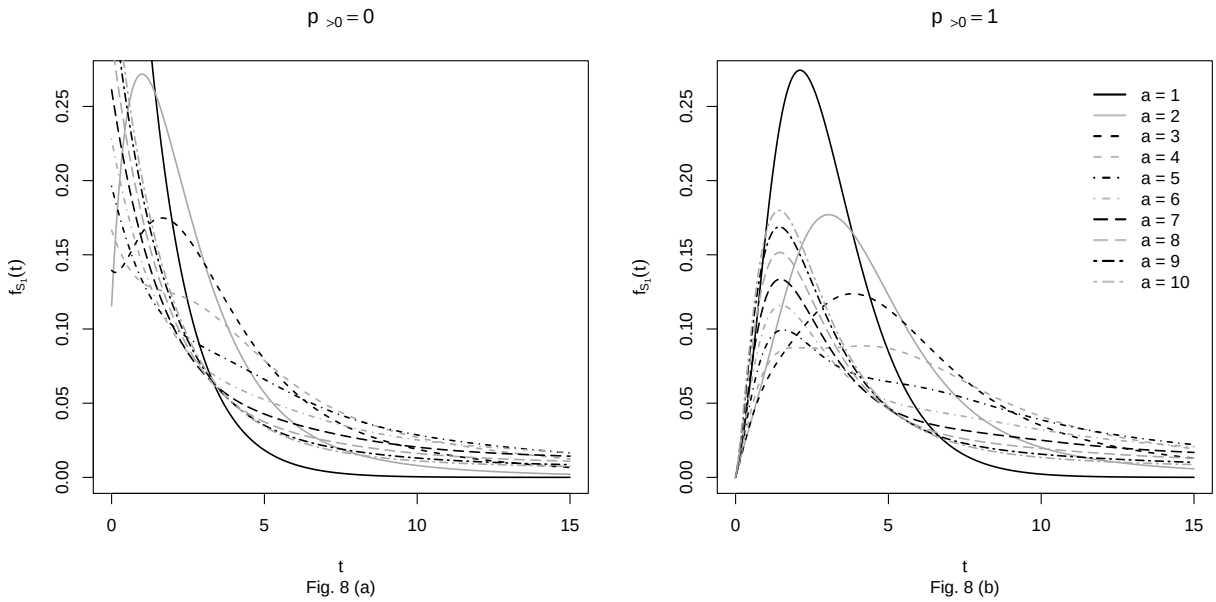

Figure 8: Plots of class-1 sojourn time densities for (a, a) threshold policies, $a = 1, 2, \ldots, 10$, with exponentially distributed service times, $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$, $p_{>0} = 0, 1$, $C = 8$, $f = 2$, and $\alpha = 0.05$.

and have lighter tails. Letting $p_{>0} = 0$, sojourn times for a class-1 machine will be shortened on average due to not having to potentially wait for the server to switch depending on the state of the system at the failure epoch, as well as having fewer class-1 machines queued ahead of it caused by the system's higher rate at which machines are repaired, $\lambda_r^{[C,f]}$, as a consequence of the server never being idle when there are still broken machines to repair (as observed in Table 2 for a range of policies).

Of these plots, the $(4, 4)$ threshold policy stands out as having a particularly interesting density, exhibiting a bimodal structure in the $p_{>0} = 1$ case as it has two local maxima. A sojourn time of a machine will depend greatly on the initial state of the system immediately after its failure epoch, particularly on the location of the server, so we decompose the density $f_{\mathcal{S}_1}(t)$ into components $f_{\mathcal{S}_1,L_I}(t)$ where $L_I \in \{1, 2, 3, 4\}$ are the possible server locations after observing the failure. We achieve this decomposition by considering each case separately, modifying Equation (14) (and hence $\underline{\Phi}_1$) by setting any element p_{m,n,l,y,y_1,y_2} of the probability vector with $l \neq L_I$ equal to zero. If we re-normalized the modified $\underline{\Phi}_1$'s, then this would alternatively result in the conditional distributions of a class-1 sojourn time given different initial server locations.

Due to the nature of our considered preemptive resume threshold policies, sojourn times when $L_I = 3$ (class-2 switch-in) or $L_I = 4$ (class-2 service) will be comparable in the majority of cases due to a class-2 switch-in time being small relative to a service time and the threshold being commonly triggered prior to the next class-2 service completion. Thus, we keep these

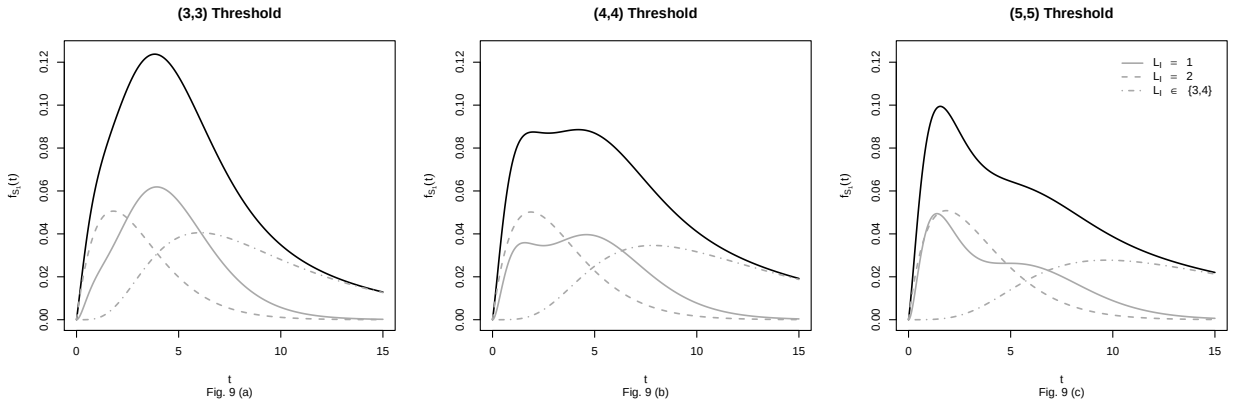

Figure 9: Plots of class-1 sojourn time densities and their component densities $f_{\mathcal{S}_1,L_I}$ for (a,a) threshold policies, $a = 3,4,5$, with exponentially distributed service times, $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$, $p_{>0} = 1$, $C = 8$, $f = 2$, and $\alpha = 0.05$.

two cases grouped together, leaving us with $L_I = 1$, $L_I = 2$, and $L_I \in \{3,4\}$, so that

$$f_{\mathcal{S}_1}(t) = f_{\mathcal{S}_1,1}(t) + f_{\mathcal{S}_1,2}(t) + f_{\mathcal{S}_1,\{3,4\}}(t).$$

In Figure 9, we plot the densities and their three components for $a = 3,4,5$. We observe that the components $f_{\mathcal{S}_1,2}(t)$ are very comparable, whereas $f_{\mathcal{S}_1,\{3,4\}}(t)$ has its density allocated to larger sojourn times as a increases, representing the requirement of more total class-1 machine failures to trigger the higher thresholds (which also increases the probability of needing to wait for one or more class-2 repairs to complete prior to receiving service). Note that $f_{\mathcal{S}_1,\{3,4\}}(t)$ appears to be solely responsible for the remaining tails of these distributions, as the machine will almost surely be repaired within 15 time units if the server is either already serving class 1 or is switching to class 1 after the target machine fails.

It is in $f_{\mathcal{S}_1,1}(t)$ that we observe great variability between the adjacent thresholds, including the bimodal structure observed in the $(4,4)$ threshold policy. We note that this second local maxima is near 5 time units. If the target machine triggered the threshold, then it would take on average 2 time units for the class-1 switch-in time and 4 time units to repair the target machine and the three machines queued ahead of it, for a total of 6 time units. In fact, plotting the density of the sojourn time in this specific case (which we omit here) results in a right-skewed density possessing a single maxima just after $t = 5$. We therefore suspect a large portion of initial states to be of this type, causing the observed second maxima.

Letting $p_{m,\bullet,1,\bullet,\bullet,\bullet} = \sum_{n,y,y_1,y_2} p_{m,n,1,y,y_1,y_2}$ denote the marginal probability of the server conducting a class-1 switch-in immediately after a class-1 machine failure fills the m^{th} slot in queue 1, we compare these probabilities for $m = 1,2,\ldots,8$ and $a = 3,4,5$ in Table 6. It would seem that in the $L_I = 1$ cases, there is indeed a large jump in initial probability for cases where the target machine is indeed the threshold trigger. The other most likely cases denote a failure to an empty system (i.e., $m = 1$) which is more likely the higher the threshold (as it lets the class 2 queue empty faster) and at $m = a+1$ indicating cases where

the target machine failed during a switch-in time in progress which was triggered by the preceding class-1 machine failure. Other choices of m have much less probability as they require there to be multiple failures during the short switch-in time.

Table 6: Marginal $L_I = 1$ class-1 sojourn time distribution initial probabilities, $p_{m,\bullet,1,\bullet,\bullet,\bullet}$, for the (a, a) threshold service policy with $a = 3, 4, 5$, exponentially distributed service times, $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$, $p_{>0} = 1$, $C = 8$, $f = 2$, and $\alpha = 0.05$.

	m							
(a, b)	1	2	3	4	5	6	7	8
$(3, 3)$	0.0395	0.0127	0.2181	0.0597	0.0106	0.0014	0.0001	< 0.0001
$(4, 4)$	0.0688	0.0213	0.0074	0.1512	0.0376	0.0058	0.0006	< 0.0001
$(5, 5)$	0.0992	0.0300	0.0095	0.0049	0.1076	0.0228	0.0028	0.0002

We therefore conclude that the jumps in $p_{m,\bullet,1,\bullet,\bullet,\bullet}$ near $m = 1$ and $m = a$ are responsible for the shapes of density components $f_{\mathcal{S}_1,1}(t)$ (as they of course represent mixtures of distributions), namely the bimodal structure of the $(4, 4)$ threshold policy as well as the flat region in the $(5, 5)$ threshold policy. For the $(3, 3)$ threshold policy, $p_{3,\bullet,1,\bullet,\bullet,\bullet}$ is much larger than $p_{1,\bullet,1,\bullet,\bullet,\bullet}$, which hides this obvious mixture appearance.

5.3 Smart Bernoulli Optimization

Among other service policies, we considered $(1, 0.2)$ and $(1, 0.8)$ smart Bernoulli in Table 2 and Figures 4 and 5. It was evident that due to $(1, 0.2)$ smart Bernoulli's higher preference for serving class-1 machines (causing additional switch-ins), it both converged to $\mathrm{E}[N_W^{[\infty]}]$ at fewer total machines when switch-in times were identically zero in duration, and to a lower limit when switch-in time durations had positive expected values, in comparison to $(1, 0.8)$ smart Bernoulli. In Table 2, $(1, 0.2)$ had a larger $\mathrm{E}[N_W^{[C,f]}]$ in every considered case except when $p_{>0} = 1$ and $[C, f] = [14, 0]$. In this subsection, we will investigate some new examples to observe the impact of switch-ins and the number of machines on the optimal selection of smart Bernoulli probabilities that maximizes $\mathrm{E}[N_W^{[C,f]}]$.

First of all, we justify the choice of $p_1^{SB} = 1$. The $c\mu$ rule (Meilijson and Yechiali 1977, Van Mieghem 1995) states that in a priority queue, if class-i customers have a holding cost of c_i per time unit and an expected service time of $1/\mu_i$, then the classes should be served in decreasing order of $c_i\mu_i$, independent of arrival rate. For finite-population systems, this is not necessarily true, as the presence of a broken machine waiting to be serviced reduces the number of machines that can fail of that type (i.e., despite a potentially fast service time, if each class of machines comes from its own independent population and the time to failure for class-i machines is small, then it may not be optimal to give them higher service priority). A modified $c\mu\lambda$ rule (Iravani and Kolfal 2005) was investigated for a fully exponential model of this type (an example of the machine-repairman problem). Based on certain assumptions and conditions, it was concluded that priority may be given to class j with positive queue

length if $\frac{c_j\mu_j}{\lambda_j} \geq \frac{c_k\mu_k}{\lambda_k}\ \forall\ k \neq j$, such that there is at least one class-k machine waiting to be repaired.

In our model, since both classes of failure come from the same pool of machines, we can effectively ignore the fact that we are using a finite-population system since no matter which machine type is repaired, the time until it fails again has an identical distribution. Thus, we can consider the standard $c\mu$ rule. For our model under the case of zero duration switch-in times, we would assign priority to the class with the highest value of $c_i\mu_i$, and as such always prefer to serve it over the other class. In our investigation, we simply want to maximize the expected number of working machines, so we would select equal holding costs (e.g., $c_1 = c_2 = 1$), as a broken machine of either type equally lowers the expected number of working machines. Therefore, by the $c\mu$ rule, the class with the highest μ_i (i.e., shortest expected service time) should have priority, corresponding to class 1 in our numerical examples.

Now, if in this zero switch-in case we would never want to switch away from class 1 (to go serve class 2), then in the cases with positive switch-in times, it follows that it would still never be optimal to switch away from class 1 since not only would the mechanic switch to serving the less efficient-to-serve class, they must incur a period of idleness during the switch-in which reduces their average rate of repair. Thus, similar to the arguments of Blanc and van der Mei (1995), we can conclude that in the smart Bernoulli framework, class 1 (having the smallest average repair times) should receive a probability of $p_1^{SB} = 1$ to continue repairs (and hence, not switching) after each service completion, should its queue not be empty.

It is not as clear for the lower priority class 2. If there were no switch-in times, then it would be optimal to switch after every service completion and have a probability of starting another service of $p_2^{SB} = 0$. However, as each positive duration switch-in time incurs idleness, in reality there may be an optimal p_2^{SB} that is positive. This probability is what we must find to optimize the use of smart Bernoulli in our model. To do this, we find the approximate $\hat{p}_2^{SB}$ that maximizes $\mathrm{E}[N_W]$ using the algorithm outlined in the Appendix. For all approximated optimal $\hat{p}_2^{SB}$ in this subsection, we set precision = 4 (i.e., we approximate to four decimal places).

We now investigate the impacts of reducing p_1^{SB} from 1 (considering $p_1^{SB} \in \{0.9, 0.95, 1\}$) and varying the expected switch-in time durations in Figure 10, where we plot the optimal p_2^{SB} against $p_{>0}$ (with $M_{\mathrm{S}} = 1$) or M_{S} (with $p_{>0} = 1$), so that the mean switch-in time durations are equal and hence comparable. The corresponding values of $\mathrm{E}[N_W]$ calculated using the optimal values of p_2^{SB} for the $\alpha = 0.10$ cases are plotted in Figure 11. We set $M_{\mathrm{B}} = 1$ and allow both class' service time distributions to be exponential, H_2, or E_3, to observe the effect of service variance, while letting $\alpha \in \{0.075, 0.10\}$, $C = 8$, and $f = 2$. Additionally, we approximate the impact of heavy-tailed service time distributions by applying the EM algorithm (Asmussen et al. 1996) to fit log-normal (LN) distributions to continuous phase-type distributions of order 5. A summary of the service time distributions' expectations and variances are provided in Table 7. Log-normal parameters were selected to match mean repair time values, while being slightly less variable than the H_2 distributions. While the approximations of these LN distributions provide very close fits for the expected values, the

difficulty of accurately fitting heavy tails is evident by their smaller variances.

Table 7: Expected values and variances for service time distributions Exp, H_2, and E_3, along with the LN distributions of interest and their EM algorithm fits using continuous phase-type distributions of order 5.

	Class 1		Class 2	
Service	Expectation	Variance	Expectation	Variance
H_2	1	5.5	20	2200
Exp	1	1	20	400
E_3	1	1/3	20	400/3
LN	1	4	20	2000
LN (fit)	0.99998	3.80365	19.97997	1227.85435

In Figure 10, we can see that for very small switch-in times it is optimal to maintain $p_2^{SB} = 0$ and act as a class-1 non-preemptive priority policy (or similar to one if $p_1^{SB} < 1$), but by increasing the mean switch-in times we make additional switches (relative to the exhaustive service policy) more costly and it becomes optimal for p_2^{SB} to become positive, eventually reaching $p_2^{SB} = 1$ in order to minimize the number of switch-ins (note that in the $p_1^{SB} = 1$ and $\alpha = 0.075$ case, if we continue to increase M_S beyond 1, then these curves will also hit $p_2^{SB} = 1$).

By decreasing p_1^{SB}, it becomes possible to switch away from class 1 before its queue empties and it is not hard to see that this will have the effect of increasing the fraction of time that the mechanic is idle. This has the effect of increasing the slopes of the curves in Figure 10, indicating that the behavior dictating how the mechanic treats class 2 is more sensitive to the expected switch-in time durations and will opt to treat class 2 in an exhaustive manner sooner, even if they are not allowed to do the same for class 1, in order to compensate for the additional class-2 switch-ins out of class 1.

By decreasing α, there are fewer failures resulting in shorter queue lengths and less opportunities for the smart Bernoulli policy to cause the server to leave before emptying a queue. Therefore, increasing p_2^{SB} has a smaller impact on reducing the number of extra switch-ins and the mean switch-in durations need to be larger before it becomes optimal to use a positive p_2^{SB}.

Comparing the four sets of service time distributions, they transition from $p_2^{SB} = 0$ to $p_2^{SB} = 1$ at comparable rates, but the more variable distributions require more incentive in the form of higher costs from switch-in times to increase p_2^{SB} from 0. This follows since the more class-2 services that are completed before returning to the class-1 queue, the more opportunities there are for the server to be stuck in a particularly long service time (e.g., the 10% case in the class-2 H_2 distribution having mean 110) which will have a large effect on the sojourn times of class-1 machines that are waiting to be serviced. As service variance is reduced, there is less uncertainty accepted from additional class-2 service times and the mechanic is willing to begin increasing p_2^{SB} at smaller mean switch-in times. We observe

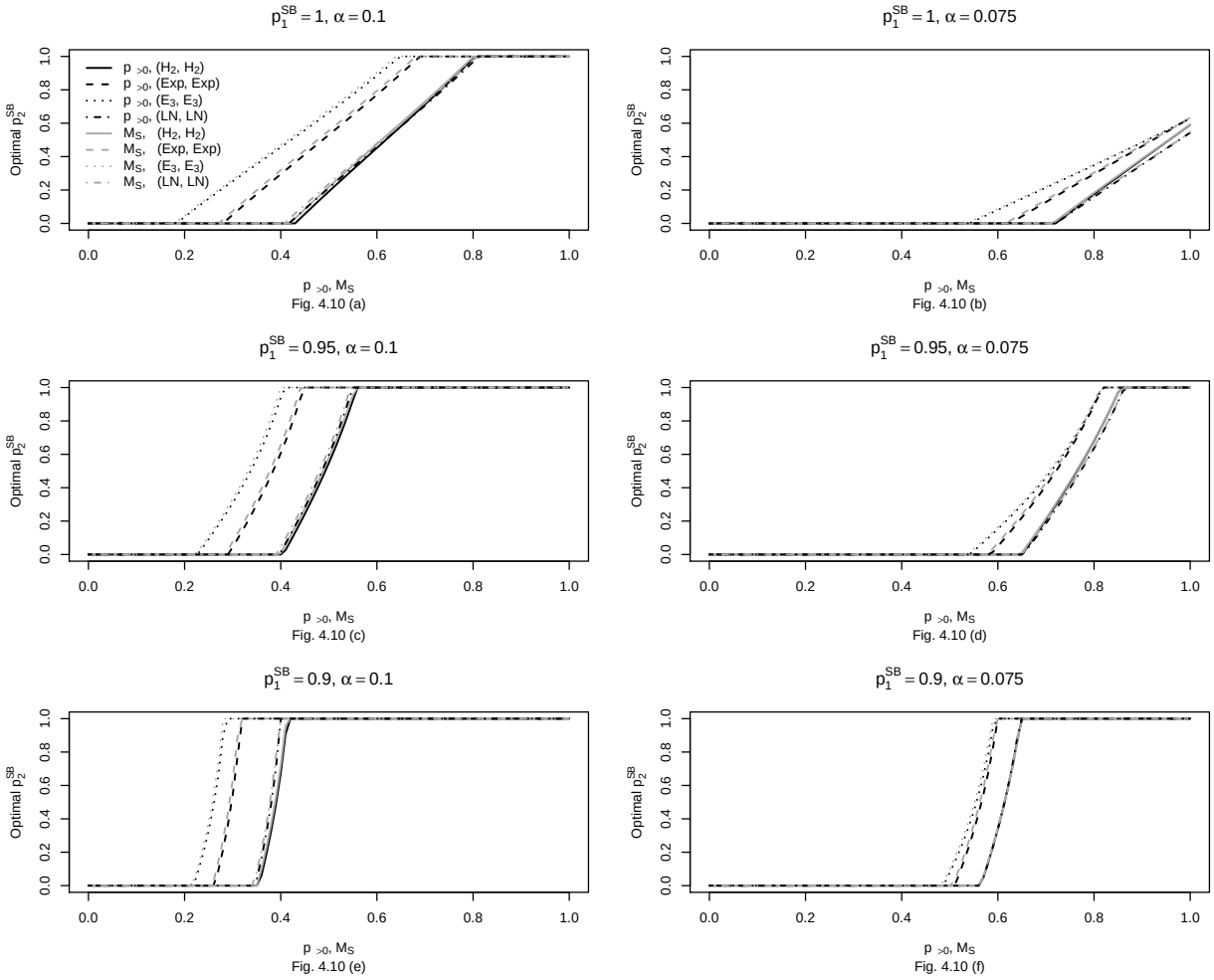

Figure 10: Plots of optimal class-2 smart Bernoulli probability p_2^{SB} against $p_{>0}$ (with $M_{\rm S} = 1$) or $M_{\rm S}$ (with $p_{>0} = 1$) for $\alpha = 0.075, 0.1$ and $p_1^{SB} = 0.9, 0.95, 1$.

that the H_2 and LN service time distributions result in very close optimal values of p_2^{SB}. As H_2 is more variable, this would indicate that optimality must be less sensitive to changes in variance when it is already large.

In Figure 11, we confirm that reducing p_1^{SB} lowers the maximum $\mathrm{E}[N_W]$ possible at the corresponding optimal p_2^{SB} probabilities. The differences between the plots may not be large, but note that these are not for fixed p_2^{SB}, but rather the optimal p_2^{SB}'s at each $p_{>0}$ or $M_{\rm S}$ given the different values of p_1^{SB}. Additionally, we observe that increasing service variance has a negative effect on the mean number of working machines (e.g., H_2 has lower values than LN, despite similar optimal p_2^{SB} probabilities), but the relationship between $\mathrm{E}[N_W]$ and switch-in times is primarily dependent on the first moments. Interestingly, these relationships are approximately linear between $\mathrm{E}[N_W]$ and the mean switch-in times in ranges where the optimal p_2^{SB} are unchanged, either at 0 or 1.

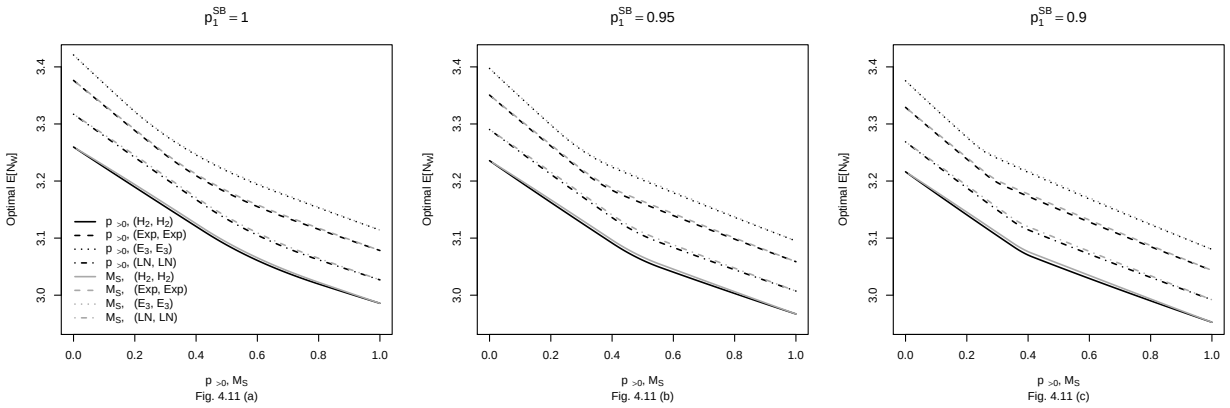

Figure 11: Plots of $\mathrm{E}[N_W]$ at optimal class-2 smart Bernoulli probabilities against $p_{>0}$ (with $M_{\mathrm{S}} = 1$) or M_{S} (with $p_{>0} = 1$) for $\alpha = 0.10$ and $p_1^{SB} = 0.9, 0.95, 1$.

In Figure 12, we plot optimal p_2^{SB} against $C = 3, 4, \ldots, 25$ for $f = 0, 2, 4$, $p_1^{SB} = 1$, $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$, $p_{>0} = 0.5$, and $\alpha \in \{0.05, 0.075, 0.10\}$. We observe that increasing C results in more failures, longer queue lengths, and more opportunities for a smart Bernoulli policy to cause a switch from a queue before it is emptied. Therefore, as C becomes large, with the exception of the fitted LN distributions, the server eventually increases p_2^{SB} until class 2 is treated in an exhaustive manner. Increasing f has a similar effect, and as the total number of machines are greater for a given C, the mechanic begins the transition from class-1 non-preemptive priority to an exhaustive policy at fewer C, acting largely as a horizontal shift with minimal effect on the rate of increase in p_2^{SB}. By increasing α, the sensitivity of the optimal p_2^{SB} on C is heightened as every increment of C has a larger impact on the average failure rate, causing p_2^{SB} to transition from 0 to 1 at fewer total machines and at a faster rate. Finally, in comparing the Exp, H_2, and E_3 service time distributions, we observe results consistent with those from Figure 10, in that it becomes optimal to increase p_2^{SB} earlier (i.e., for smaller C) for service time distributions having smaller variances.

While the fitted LN distributions acted very similarly to the H_2 distributions in Figure 10, they display a unique behavior in Figure 12. In Figure 12, rather than converging to an exhaustive discipline as C increases, the optimal p_2^{SB} peaks before decreasing to some positive limit. This peak seems highest for the cases with $f = 0$, falling just short of 1 in Figure 12 (d). In part (f), all three plots hit $p_2^{SB} = 1$ before moving away from the exhaustive policy at 20 total machines. As this behavior is not shared with the other pair of highly variable service time distributions, this seems to suggest that it must be due to the more general structure of the fitted continuous phase-type distributions. As these are intended to behave similarly to heavy-tailed distributions, it would be of great interest to revisit this problem using a like model within a semi-Markov framework. This would allow the usage of general distributions for service and switch-in times, and hence, enable us to investigate the true impact of heavy-tailed distributions.

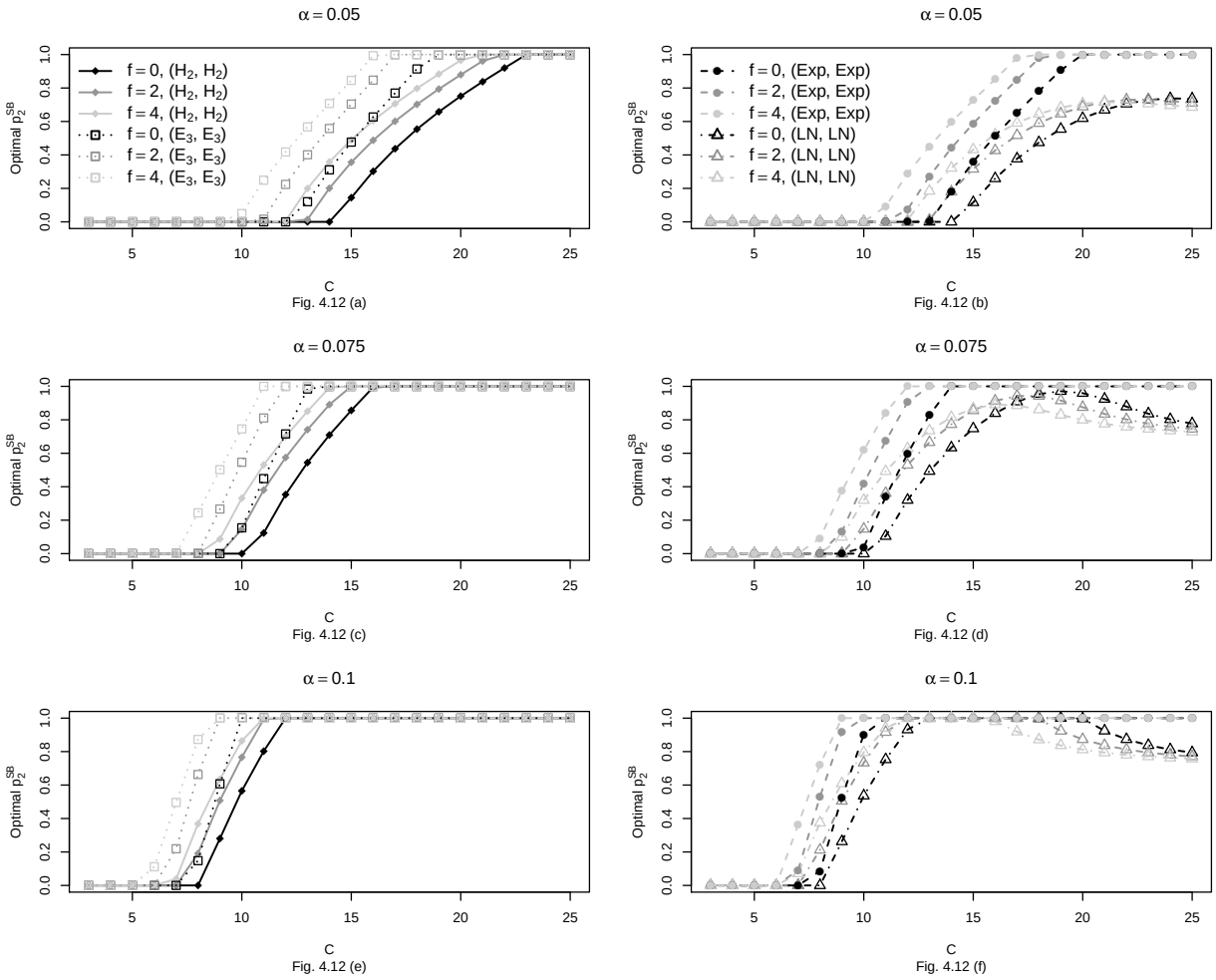

Figure 12: Plots of optimal class-2 smart Bernoulli probability p_2^{SB} against C for $f = 0, 2, 4$, $\alpha = 0.05, 0.075, 0.10$, $p_{>0} = 0.5$, $p_1^{SB} = 1$, and $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$.

6 Concluding Remarks

We have investigated a closed maintenance system having a capacity of C machines and an optional maintenance float of f spare machines. Working machines can suffer one of multiple types of failures which are assigned to one of two classes, whose queues are visited by a single mechanic according to a dynamic behavior which can replicate several classic service policies, as well as the proposed (a, b) threshold and (p_1^{SB}, p_2^{SB}) smart Bernoulli policies. The system was modelled as a level-dependent QBD process and matrix analytic methods were applied to obtain the steady-state probabilities as well as to derive the phase-type distribution of a broken machine's sojourn time. Limit results as well as connections to mean sojourn times were presented for the expected number of working machines, and three numerical examples were conducted.

In our future work, we anticipate extending the number of classes to three or more. For instance, in the 3-class system, allowing a class to hold medium jobs which were previously grouped with small or large jobs may lead to further gains in system optimization, depending on the presence and size of switch-in times. If we maintain the current assumption of phase-type distributed service and switch-in times, our server decision process lends itself to the framework of a Markov decision process. It is of interest to use MDPs to obtain the globally optimal policy and to see if it can noticeably outperform the optimized (a, b) threshold policies. Additionally, an alternative way in which we envision extending this work is to relax these phase-type distributional assumptions to general ones and to analyze the model as a semi-Markov process.

Acknowledgements

Steve Drekic and Kevin Granville acknowledge the financial support from the Natural Sciences and Engineering Research Council of Canada through its Discovery Grants program (RGPIN-2016-03685) and Postgraduate Scholarship-Doctoral program, respectively. The authors are also very thankful to the referee for their helpful comments and suggestions, which have improved the presentation of the paper as well as widened the scope of the discussion in one of our numerical examples.

Appendix

Proof of Theorem 1

We begin by remarking that the infinitesimal generator subblocks $Q_{0,1,0}^{[C,f]}$ and $(UD)_{0,0}^{[C,f]}$ are both comprised of $1+s_0$ identical rows equal to $C\alpha_1\underline{\gamma}_{01}^{[+0]}\otimes\underline{\beta}_1$ and $C\alpha_2\underline{\gamma}_{02}^{[+0]}\otimes\underline{\beta}_2$, respectively. This implies that given a machine failure has occurred, the CTMC transitions away from any of the empty queue states $\{(0,0,0,0,0,0)\}\cup\{(0,0,5,y,0,0), y=1,2,\ldots,s_0\}$ in an identical fashion. This observation immediately follows from our assumption that interrupting and switching away from a class-i switch-in is treated the same as the server switching away from class i itself.

We now consider the differences between a system containing k machines where $[C,f]=[k,0]$ or $[C,f]=[k-1,1]$. The two systems will act identically, in terms of infinitesimal generator construction, with the exception of the rows for states where all k machines are functional (the first system puts the k^{th} machine to use, while the second stores it in the maintenance float). In either case, the total time spent visiting any combination of the empty queue states between the previous service completion and the next observed failure will have an exponential distribution with rate $C\alpha$ (i.e., $\text{Exp}(C\alpha)$). Hence, we may adjust the CTMCs and consolidate the empty queue states into a single state $(0,0)$ with steady-state probability $\pi_{0,0}^{[C,f]}=\underline{\pi}_{0,0}^{[C,f]}\underline{e}'_{1+s_0}$, such that we do not track the potential phase-type class-0 switch-in time and the sojourn time in this state is simply the time until the next machine failure. Note that

this consolidation will not affect the other steady-state probabilities due to the identical rows of $Q_{0,1,0}^{[C,f]}$ and $(UD)_{0,0}^{[C,f]}$ which each are now just present once, corresponding to transitions out of state $(0,0)$. Thus, we have at steady state

$$\begin{aligned}
\mathrm{E}[N_W^{[C,f]}] &= \mathrm{E}[\min\{C, C+f-X_1^{[C,f]}-X_2^{[C,f]}\}] \\
&= \sum_{m,n,l,y,y_1,y_2} \min\{C, C+f-m-n\}\pi_{m,n,l,y,y_1,y_2}^{[C,f]} \\
&= C\left(\pi_{0,0,0,0,0,0}^{[C,f]} + \sum_{y=1}^{s_0} \pi_{0,0,5,y,0,0}^{[C,f]}\right) \\
&+ \sum_{m+n\neq 0} \sum_{l,y,y_1,y_2} \min\{C, C+f-m-n\}\pi_{m,n,l,y,y_1,y_2}^{[C,f]} \\
&= C\pi_{0,0}^{[C,f]} + \sum_{m+n\neq 0} \sum_{l,y,y_1,y_2} \min\{C, C+f-m-n\}\pi_{m,n,l,y,y_1,y_2}^{[C,f]}. \qquad (36)
\end{aligned}$$

That is, the expected number of working machines will be the same in the original CTMCs and the corresponding adjusted CTMCs with the consolidated empty queue state.

Let $\psi_{0,0}^{[C,f]}$ and $\psi_{m,n,l,y,y_1,y_2}^{[C,f]}$ denote the steady-state probabilities of the embedded discrete-time Markov chain, or *DTMC* (e.g., Syski 1992, p. 14), describing an adjusted CTMC with a given $[C,f]$. As the generators for $[k,0]$ and $[k-1,1]$ are now identical outside of the first rows for state $(0,0)$, which for $[k,0]$ is

$$\begin{bmatrix} -k\alpha & k\alpha_2\underline{\gamma}_{02}^{[+0]} \otimes \underline{\beta}_2 & \underline{0} & \cdots & \underline{0} & k\alpha_1\underline{\gamma}_{01}^{[+0]} \otimes \underline{\beta}_1 & \underline{0} & \cdots & \underline{0} \end{bmatrix}$$

and for $[k-1,1]$ is

$$\begin{bmatrix} -(k-1)\alpha & (k-1)\alpha_2\underline{\gamma}_{02}^{[+0]} \otimes \underline{\beta}_2 & \underline{0} & \cdots & \underline{0} & (k-1)\alpha_1\underline{\gamma}_{01}^{[+0]} \otimes \underline{\beta}_1 & \underline{0} & \cdots & \underline{0} \end{bmatrix},$$

it is clear that while the steady-state probabilities for the CTMCs differ, it holds that $\psi_{0,0}^{[k,0]} = \psi_{0,0}^{[k-1,1]}$ and $\psi_{m,n,l,y,y_1,y_2}^{[k,0]} = \psi_{m,n,l,y,y_1,y_2}^{[k-1,1]}$.

It is known from the theory of semi-Markov processes (e.g., Ross 2014, p. 445) that if the long-run proportion of transitions by a semi-Markov process into state i is π_i (i.e., the steady-state probability of the embedded DTMC being in state i) and the amount of time spent in state i before transitioning away has mean μ_i, then the long-run proportion of time that the semi-Markov process is in state i is

$$\frac{\pi_i\mu_i}{\sum_{j=1}^{N}\pi_j\mu_j}, \qquad (37)$$

where N is the total number of states. Since we are considering CTMCs, the time spent in a state is exponentially distributed with a mean equal to the negative inverse of that state's corresponding main diagonal element from the infinitesimal generator. Let $\mu_{m,n,l,y,y_1,y_2}^{[k,0]} = \mu_{m,n,l,y,y_1,y_2}^{[k-1,1]}$ denote the mean time spent in a visit to state (m,n,l,y,y_1,y_2), and $\mu_{0,0}^{[k,0]} = \frac{1}{k\alpha}$

and $\mu_{0,0}^{[k-1,1]} = \frac{1}{(k-1)\alpha}$ be the mean times spent in visits to the empty queue state in either adjusted CTMC. We then have

$$\pi_{m,n,l,y,y_1,y_2}^{[C,f]} = \frac{\psi_{m,n,l,y,y_1,y_2}^{[C,f]}\mu_{m,n,l,y,y_1,y_2}^{[C,f]}}{\psi_{0,0}^{[C,f]}\mu_{0,0}^{[C,f]} + \sum_{x_1+x_2\neq 0}\sum_{w,z,z_1,z_2}\psi_{x_1,x_2,w,z,z_1,z_2}^{[C,f]}\mu_{x_1,x_2,w,z,z_1,z_2}^{[C,f]}}$$

and

$$\pi_{0,0}^{[C,f]} = \frac{\psi_{0,0}^{[C,f]}\mu_{0,0}^{[C,f]}}{\psi_{0,0}^{[C,f]}\mu_{0,0}^{[C,f]} + \sum_{x_1+x_2\neq 0}\sum_{w,z,z_1,z_2}\psi_{x_1,x_2,w,z,z_1,z_2}^{[C,f]}\mu_{x_1,x_2,w,z,z_1,z_2}^{[C,f]}}.$$

Let

$$D^{[C,f]} = \sum_{x_1+x_2\neq 0}\sum_{w,z,z_1,z_2}\psi_{x_1,x_2,w,z,z_1,z_2}^{[C,f]}\mu_{x_1,x_2,w,z,z_1,z_2}^{[C,f]},$$

which we know satisfies $D^{[k,0]} = D^{[k-1,1]}$. It now follows that

$$\begin{aligned}
\pi_{m,n,l,y,y_1,y_2}^{[k,0]} &= \frac{\psi_{m,n,l,y,y_1,y_2}^{[k,0]}\mu_{m,n,l,y,y_1,y_2}^{[k,0]}}{\psi_{0,0}^{[k,0]}\mu_{0,0}^{[k,0]} + D^{[k,0]}} \\
&= \frac{\psi_{m,n,l,y,y_1,y_2}^{[k-1,1]}\mu_{m,n,l,y,y_1,y_2}^{[k-1,1]}}{\psi_{0,0}^{[k-1,1]}\mu_{0,0}^{[k-1,1]} + D^{[k-1,1]}} \times \frac{\psi_{0,0}^{[k-1,1]}\mu_{0,0}^{[k-1,1]} + D^{[k-1,1]}}{\psi_{0,0}^{[k,0]}\mu_{0,0}^{[k,0]} + D^{[k,0]}} \\
&= \pi_{m,n,l,y,y_1,y_2}^{[k-1,1]}c_k, \qquad (38)
\end{aligned}$$

where

$$c_k = \frac{\psi_{0,0}^{[k-1,0]}\mu_{0,0}^{[k-1,1]} + D^{[k-1,1]}}{\psi_{0,0}^{[k,0]}\mu_{0,0}^{[k,0]} + D^{[k,0]}} = \frac{\frac{1}{(k-1)\alpha}\psi_{0,0}^{[k,0]} + D^{[k,0]}}{\frac{1}{k\alpha}\psi_{0,0}^{[k,0]} + D^{[k,0]}} > 1. \qquad (39)$$

Similarly,

$$\begin{aligned}
\pi_{0,0}^{[k,0]} &= \frac{\psi_{0,0}^{[k,0]}\mu_{0,0}^{[k,0]}}{\psi_{0,0}^{[k,0]}\mu_{0,0}^{[k,0]} + D^{[k,0]}} \\
&= \frac{\psi_{0,0}^{[k-1,1]}\mu_{0,0}^{[k-1,1]}\left(\frac{k-1}{k}\right)}{\psi_{0,0}^{[k-1,1]}\mu_{0,0}^{[k-1,1]} + D^{[k-1,1]}} \times \frac{\psi_{0,0}^{[k-1,1]}\mu_{0,0}^{[k-1,1]} + D^{[k-1,1]}}{\psi_{0,0}^{[k,0]}\mu_{0,0}^{[k,0]} + D^{[k,0]}} \\
&= \pi_{0,0}^{[k-1,1]}\left(\frac{k-1}{k}\right)c_k. \qquad (40)
\end{aligned}$$

Note that we can find an upper bound on c_k. As the steady-state probabilities for both cases must respectively sum to 1, using Equations (38) and (40), it must simultaneously hold that

$$\begin{aligned}
1 &= \pi_{0,0}^{[k,0]} + \sum_{x_1+x_2\neq 0}\sum_{w,z,z_1,z_2}\pi_{x_1,x_2,w,z,z_1,z_2}^{[C,f]} \\
&= \pi_{0,0}^{[k-1,1]}\left(\frac{k-1}{k}\right)c_k + c_k\sum_{x_1+x_2\neq 0}\sum_{w,z,z_1,z_2}\pi_{m,n,l,y,y_1,y_2}^{[k-1,1]}
\end{aligned}$$

and

$$1 = \pi_{0,0}^{[k-1,1]} + \sum_{x_1+x_2\neq 0} \sum_{w,z,z_1,z_2} \pi_{m,n,l,y,y_1,y_2}^{[k-1,1]}.$$

Clearly, as every probability is non-negative, by Equation (39),

$$c_k \sum_{x_1+x_2\neq 0} \sum_{w,z,z_1,z_2} \pi_{m,n,l,y,y_1,y_2}^{[k-1,1]} > \sum_{x_1+x_2\neq 0} \sum_{w,z,z_1,z_2} \pi_{m,n,l,y,y_1,y_2}^{[k-1,1]}$$

implying that we must have

$$\pi_{0,0}^{[k-1,1]} \left(\frac{k-1}{k}\right) c_k < \pi_{0,0}^{[k-1,1]},$$

or equivalently,

$$1 < c_k < \frac{k}{k-1}.$$

Finally, using Equations (36) - (40),

$$\begin{aligned}
&\mathrm{E}[N_W^{[k,0]}] \\
&= k\pi_{0,0}^{[k,0]} + \sum_{m+n\neq 0} \sum_{l,y,y_1,y_2} \min\{k, k+0-m-n\} \pi_{m,n,l,y,y_1,y_2}^{[k,0]} \\
&= k\pi_{0,0}^{[k,0]} + \sum_{m+n\neq 0} \sum_{l,y,y_1,y_2} (k-m-n) \pi_{m,n,l,y,y_1,y_2}^{[k,0]} \\
&= k\pi_{0,0}^{[k-1,1]} \left(\frac{k-1}{k}\right) c_k + \sum_{m+n\neq 0} \sum_{l,y,y_1,y_2} (k-m-n) \pi_{m,n,l,y,y_1,y_2}^{[k-1,1]} c_k \\
&= c_k \left((k-1)\pi_{0,0}^{[k-1,1]} + \sum_{m+n\neq 0} \sum_{l,y,y_1,y_2} \min\{k-1, k-1+1-m-n\} \pi_{m,n,l,y,y_1,y_2}^{[k-1,1]} \right) \\
&= c_k \mathrm{E}[N_W^{[k-1,1]}] \\
&> \mathrm{E}[N_W^{[k-1,1]}].
\end{aligned}$$

□

Proof of Theorem 2

In order to consider the limit of the expected number of working machines, we need to first find an expression for $\mathrm{E}[N_W^{[C,f]}]$. Similar to Abboud (1996), we consider the number of working machines as a subsystem and apply the result of Little (1961). Recall that Little's Law states that the expected number of "customers" in a system ($\mathrm{E}[L]$) is equal to the product of their average arrival rate (λ) and the expected amount of time that a customer spends in the system ($\mathrm{E}[W]$).

As we are treating the number of *working* machines as the subsystem, not the number of *functional* machines, it is clear that W is simply the time until a working machine fails. Thus, we have $W \sim \text{Exp}(\alpha)$, and so

$$\text{E}[W] = \frac{1}{\alpha}, \tag{41}$$

which is independent of C, f, and the service policy. Next, we require the limiting aggregate rate that machines fail and are repaired, which we define as $\lambda_r^{[C,f]}$, which is the effective average "arrival rate" of repaired machines satisfying

$$\text{E}[N_W^{[C,f]}] = \lambda_r^{[C,f]}\text{E}[W] = \frac{\lambda_r^{[C,f]}}{\alpha}. \tag{42}$$

We cite a result from the theory of renewal reward processes (e.g., Ross 2014, p. 427), describing a system which earns a reward R_n after the n^{th} renewal of a renewal process $\{N(t), t \geq 0\}$ with interarrival times X_n, $n \in \mathbb{Z}^+$, where the R_n's are iid, but may depend on X_n. The total amount of rewards that have accumulated by time $t \geq 0$ is

$$R(t) = \sum_{n=1}^{N(t)} R_n,$$

and it is known that the long run rate at which rewards are earned is

$$\lim_{t\to\infty} \frac{R(t)}{t} = \frac{\text{E}[R]}{\text{E}[X]}. \tag{43}$$

We now define a renewal process based on our adjusted model from the proof of Theorem 1 with $[C, f]$ machines, such that a renewal occurs whenever the adjusted CTMC enters the empty queue state $(0, 0)$ (i.e., at time instants immediately after a repair which leaves all machines functional). At the end of each renewal, we receive a reward of 1 unit per observed service completion during that cycle. Applying Equation (43) to this renewal process will result in the aggregate rate at which machines are repaired. That is, if we let $\text{E}[BP^{[C,f]}]$ denote the mean duration of a busy period (i.e., the time between a failure to an empty system and when the system is empty again), then

$$\lambda_r^{[C,f]} = \frac{\text{E}[\text{Number of repairs in } BP^{[C,f]}]}{\text{E}[\text{Time until first failure at full capacity}] + \text{E}[BP^{[C,f]}]}. \tag{44}$$

Let $BP_{\text{ser}}^{[C,f]}$ and $BP_{\text{swi}}^{[C,f]}$ denote the time spent serving or switching during a busy period, respectively, such that $BP^{[C,f]} = BP_{\text{ser}}^{[C,f]} + BP_{\text{swi}}^{[C,f]}$. Note that regardless of order caused by a particular service policy, every machine that fails during (or initiating) the busy period must eventually be served. Since we assume that any preempted services are resumed when the server returns, no work is lost due to switch-ins. Therefore, if for example a class-2 repair

time has the potential to be interrupted until some number of class-1 repairs are completed, the total expected time to repair that class-2 machine is still $-\underline{\beta}_2 B_2^{-1} \underline{e}'_{b_2}$. Thus, if we let N_{BP} be the number of repairs in $BP^{[C,f]}$, then $BP^{[C,f]}_{\text{ser}}$ can be represented as the sum of all total service times observed during the busy period

$$BP^{[C,f]}_{\text{ser}} = \sum_{n=1}^{N_{BP}} Z_n^m,$$

where Z_n^m, $n = 1, 2, \ldots$, are iid random service times which are mixtures of PH$(\underline{\beta}_i, B_i)$ distributions, $i = 1, 2$, with weights α_1/α and α_2/α, having mean

$$\mathrm{E}[Z^m] = -\left(\frac{\alpha_1}{\alpha}\right) \underline{\beta}_1 B_1^{-1} \underline{e}'_{b_1} - \left(\frac{\alpha_2}{\alpha}\right) \underline{\beta}_2 B_2^{-1} \underline{e}'_{b_2}.$$

Therefore, it follows that

$$\mathrm{E}[BP^{[C,f]}_{\text{ser}}] = \mathrm{E}[\text{Number of repairs in } BP^{[C,f]}]\mathrm{E}[Z^m],$$

and Equation (44) becomes

$$\lambda_r^{[C,f]} = \frac{\mathrm{E}[BP^{[C,f]}_{\text{ser}}]/\mathrm{E}[Z^m]}{\frac{1}{C\alpha} + \mathrm{E}[BP^{[C,f]}_{\text{ser}}] + \mathrm{E}[BP^{[C,f]}_{\text{swi}}]}. \tag{45}$$

It should be noted that the distributions of N_{BP}, $BP^{[C,f]}_{\text{ser}}$, and $BP^{[C,f]}_{\text{swi}}$ (and hence $BP^{[C,f]}$) depend not only on C and f, but also on the switch-in decision probabilities. For example, a class-1 preemptive resume priority discipline will always choose to clear out the small jobs as they arrive, which will result in those machines being able to fail again sooner than if the class-2 queue had to be emptied first, hence making it more likely that the server will need to repair more total machines during that busy period in comparison to other policies. We note however that the sole act of serving more machines during a busy period, and hence between renewals, does not necessarily mean that its resulting $\lambda_f^{[C,f]}$ will be smaller or larger, as it very much also depends on whether these extra switches (relative to other disciplines) cause idle periods due to non-zero switch-in times.

We now consider the first of three cases, where $\gamma_{ji}^{[0]} = 1 \ \forall \ i, j \in \{0, 1, 2\}, i \neq j$. Clearly, this implies that $\mathrm{E}[BP^{[C,f]}_{\text{swi}}] = 0$, and Equation (45) simplifies to

$$\lambda_r^{[C,f]} = \frac{\mathrm{E}[BP^{[C,f]}_{\text{ser}}]/\mathrm{E}[Z^m]}{\frac{1}{C\alpha} + \mathrm{E}[BP^{[C,f]}_{\text{ser}}]}. \tag{46}$$

Since $\mathrm{E}[BP^{[C,f]}_{\text{ser}}] \geq \mathrm{E}[Z^m] > 0 \ \forall \ C = 1, 2, \ldots$ and $\mathrm{E}[BP^{[C,f]}_{\text{ser}}]$ is an increasing function in C

(as we will discuss shortly), by taking the limit of Equation (46), we observe that

$$\begin{aligned}
\lambda_r^{[\infty]} &= \lim_{C\to\infty} \lambda_r^{[C,f]} \\
&= \lim_{C\to\infty} \frac{\mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]/\mathrm{E}[Z^m]}{\frac{1}{C\alpha} + \mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]} \\
&= \lim_{C\to\infty} \left(\frac{1}{\frac{1}{C\alpha \mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]} + 1} \right) \frac{1}{\mathrm{E}[Z^m]} \\
&= \frac{-\alpha}{\alpha_1 \underline{\beta}_1 B_1^{-1} \underline{e}'_{b_1} + \alpha_2 \underline{\beta}_2 B_2^{-1} \underline{e}'_{b_2}}. \qquad (47)
\end{aligned}$$

Therefore, Equation (17) follows immediately from Little's Law and Equations (41) and (47).

Next, suppose that only switches out of or into class 0 can have positive durations. It then follows that $\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}]$ is a constant with respect to C, and so it still holds that

$$\lambda_r^{[\infty]} = \lim_{C\to\infty} \left(\frac{1}{\frac{(C\alpha)^{-1} + \mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}]}{\mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]} + 1} \right) \frac{1}{\mathrm{E}[Z^m]} = \frac{-\alpha}{\alpha_1 \underline{\beta}_1 B_1^{-1} \underline{e}'_{b_1} + \alpha_2 \underline{\beta}_2 B_2^{-1} \underline{e}'_{b_2}},$$

resulting in the statement of Equation (17).

Finally, we consider the cases where positive switch-in times are observable in at least one direction between the class-1 and class-2 queues (i.e., $\gamma_{12}^{[0]}$ and/or $\gamma_{21}^{[0]}$ are less than 1). We now make the seemingly obvious claim that both $\mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]$ and $\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}]$ are increasing functions in C. This is intuitive, as increasing C increases the probability flow, and hence the transition probabilities, for a given state to states within the CTMC corresponding to longer queue lengths. Also, increasing C increases the maximum total queue lengths that if visited, represent more potential total work that must be completed before the end of the busy period than a corresponding "full queue" state (i.e., $X_1 + X_2 = C + f$) in a maintenance system with a smaller C. Thus, the expected number of machine failures within a renewal period must increase with C, implying that $\mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]$ is an increasing function in C.

If machine failures are more frequent, then it also follows that the probability of observing no arrivals to the opposite queue while emptying their current queue goes to zero as $C \to \infty$. To see this, consider the system at the start of a class-i service while $X_i = 1$ and $X_j = 0$, $j \neq i$. If we assume that $f \geq 1$ and let $W_C \sim \mathrm{Exp}(C\alpha)$ and $Z_i \sim \mathrm{PH}(\underline{\beta}_i, B_i)$ be independent random variables, then the probability of having no failures during this class-i service is $P(W_C > Z_i)$, where

$$P(W_C > Z_i) = \int_0^\infty e^{-C\alpha z} \underline{\beta}_i \exp\{B_i z\} \underline{e}'_{b_i} dz = \mathrm{E}[e^{-C\alpha Z_i}] = \tilde{Z}_i(C\alpha)$$

is the Laplace transform of Z_i at $C\alpha$. If instead we had $f = 0$, then C would be replaced by $C - 1$ in the above equation. Applying the dominated convergence theorem, it is easy to

confirm that

$$\lim_{C\to\infty} P(W_C > Z_i) = \lim_{C\to\infty} \tilde{Z}_i(C\alpha) = 0,$$

Thus, as we increase C, it becomes more likely that there is a combination of class-1 and/or class-2 arrivals by the end of the service. If at least one failure was from class j, $j \neq i$, then the server will have to undergo a class-j switch-in after eventually emptying the class-i queue. If every failure was class i, then the server will have at least one more independent and probabilistically identical opportunity to observe class-j failures before either switching to class j or to class 0 (and ending the busy period). Thus, the expected number of transitions between queues after emptying a queue increases with C, which are present for every service policy. Similarly, the number of switches from positive queue lengths will be non-decreasing in C due to the CTMC spending more time at higher queue lengths, as discussed previously. Therefore, we can conclude that $\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}]$ is also an increasing function in C.

Now, we rewrite Equation (45) as

$$\lambda_r^{[C,f]} = \left(1 + \frac{1}{C\alpha \mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]} + \frac{\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}]}{\mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]}\right)^{-1} \frac{1}{\mathrm{E}[Z^m]}. \tag{48}$$

Clearly,

$$\lim_{C\to\infty} \frac{1}{C\alpha \mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]} = 0,$$

and so the limit of $\lambda_r^{[C,f]}$ depends on the rates at which $\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}]$ and $\mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]$ increase with C. If they increase at a comparable rate, i.e.,

$$\lim_{C\to\infty} \frac{\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}]}{\mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]} = d > 0,$$

then

$$\lambda_r^{[\infty]} = \left(\frac{1}{1+d}\right) \frac{1}{\mathrm{E}[Z^m]} < \frac{1}{\mathrm{E}[Z^m]},$$

implying a strict inequality in Equation (16) after applying Little's Law and Equation (41). It also follows that if

$$\lim_{C\to\infty} \frac{\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}]}{\mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]} = 0,$$

then Equation (16) is an equality.

□

Algorithm for Section 5.3: Smart Bernoulli Optimization

Letting precision $\in \mathbb{Z}^+$ denote the number of decimal places we are interested in approximating to and $\mathrm{E}[N_W](p_2^{SB})$ represent the expected number of working machines as a function of p_2^{SB}, we apply:

start $= 0$

size $= 0.1$

steps $= 11$

For $i = 1, 2, \ldots,$ precision:

For $j = 1, 2, \ldots,$ steps:

$p_{2,j}^{SB} =$ start $+ (j-1) \times$ size

$E_j = \mathrm{E}[N_W](p_{2,j}^{SB})$

$j_m = \{j \in \{1, 2, \ldots, \text{steps}\} : E_j = \max_k\{E_k\}\}$

if$(p_{2,j_m}^{SB} > 0)$

start $= p_{2,j_m}^{SB} -$ size

if$(p_{2,j_m}^{SB} < 1)$ steps $= 21$

size $=$ size$/10$

$\hat{p}_2^{SB} = p_{2,j_m}^{SB}$

What this algorithm does in iteration $i \in \{1, 2, \ldots,$ precision$\}$ is divide an interval of probabilities into increments of width 10^{-i}, solve for $\mathrm{E}[N_W]$ at each p_2^{SB} which separate the increments and determine which of these resulted in the maximum value, then restart the loop for the next i investigating an interval with length 2×10^{-i} centered around that probability, or if it is a boundary value of 0 or 1, an interval of length 10^{-i} including said boundary. The above is a condensed version of the algorithm for readability and space considerations, which may have its efficiency improved slightly by being altered to not re-calculate $\mathrm{E}[N_W]$ at any previously considered p_2^{SB}'s. We do not propose this algorithm for its speed, but rather for its accuracy to a given decimal place without the need of derivatives, and the fact that it is able to return a probability of exactly 0 or 1.

References

[1] Abboud N.E. (1996) The Markovian two-echelon repairable item provisioning problem. Journal of the Operational Research Society 47 (2): 284-296.

[2] Asmussen, S., Nerman, O., Olsson, M. (1996). Fitting phase-type distributions via the EM algorithm. Scandinavian Journal of Statistics 23 (4): 419-441.

[3] Avrachenkov K., Perel E., Yechiali U. (2016) Finite-buffer polling systems with threshold-based switching policy. TOP 24 (3): 541-571.

[4] Avram F., Gómez-Corral A. (2006) On the optimal control of a two-queue polling model. Operations research letters 34 (3): 339-348.

[5] Blanc J. P. C. (1990) A numerical approach to cyclic-service queueing models. Queueing Systems 6 (1): 173-188.

[6] Blanc J. P. C. (1991) The power-series algorithm applied to cyclic polling systems. Communications in statistics. Stochastic models 7 (4): 527-545.

[7] Blanc J. P. C., van der Mei R. D. (1995) Optimization of polling systems with Bernoulli schedules. Performance Evaluation 22 (2): 139-158.

[8] Boon M. A. A. (2011) Polling models: from theory to traffic intersections. Dissertation, Eindhoven University of Technology.

[9] Boon M. A. A., van der Mei R. D., Winands E. M. M. (2011) Applications of polling systems. Surveys in Operations Research and Management Science 16 (2): 67-82.

[10] Boxma O. J., Koole G. M., Mitrani I. (1995) Polling models with threshold switching. In: Baccelli F, Jean-Mario A, Mitrani I (eds) Quantitative Methods in Parallel Systems, Esprit basic research series. Springer, Berlin, Heidelberg.

[11] Buyukkramikli N.C., van Ooijen H.P.G., Bertrand J.W.M. (2015) Integrating inventory control and capacity management at a maintenance service provider. Annals of Operations Research 231 (1): 185-206.

[12] Gaver D., Jacobs P., Latouche G. (1984) Finite birth-and-death models in randomly changing environments. Advances in Applied Probability 16 (4): 715-731.

[13] Granville K., Drekic S. (2018) A 2-Class Maintenance Model with a Finite Population and Competing Exponential Failure Rates. Queueing Models and Service Management 1 (1): 141-176.

[14] Gross D., Miller D.R., Soland R.M. (1983) A closed queueing network model for multi-echelon repairable item provisioning. IIE Transactions 15 (4): 344-352.

[15] He Q. M. (2014) Fundamentals of matrix-analytic methods, vol. 365. Springer, New York.

[16] Iravani S. M., Kolfal B. (2005) When does the $c\mu$ rule apply to finite-population queueing systems?. Operations Research Letters 33 (3): 301-304.

[17] Iravani S. M., Krishnamurthy V., Chao G. H. (2007) Optimal server scheduling in nonpreemptive finite-population queueing systems. Queueing Systems 55 (2): 95-105.

[18] Keilson J., Servi L. D. (1986) Oscillating random walk models for GI/G/1 vacation systems with Bernoulli schedules. Journal of applied Probability 23 (3): 790-802.

[19] Kim S.K. Dshalalow J.H. (2003) A versatile stochastic maintenance model with reserve and super-reserve machines. Methodology and Computing in Applied Probability 5 (1): 59-84.

[20] Lakatos L., Szeidl L., Telek M. (2012) Introduction to queueing systems with telecommunication applications. Springer Science & Business Media, Berlin..

[21] Lee D. S., Sengupta B. (1993) Queueing analysis of a threshold based priority scheme for ATM networks. IEEE/ACM Transactions on Networking (TON) 1 (6): 709-717.

[22] Levy H. Sidi M. (1990) Polling systems: applications, modeling and optimization. IEEE Transactions on Communications COM-38 (10): 1750-1760.

[23] Liang W. K., Balcıoğlu B., Svaluto R. (2013) Scheduling policies for a repair shop problem. Annals of Operations Research 211 (1): 273-288.

[24] Lin C., Madu C. N., Kuei C. H. (1994) A closed queuing maintenance network for a flexible manufacturing system. Microelectronics Reliability 34 (11): 1733-1744.

[25] Little J. D. (1961) A proof for the queuing formula: $L = \lambda W$. Operations Research 9 (3): 383-387.

[26] Mack C. (1957) The efficiency of N machines uni-directionally patrolled by one operative when walking time is constant and repair times are variable. Journal of the Royal Statistical Society. Series B (Methodological) 19 (1): 173-178.

[27] Mack C., Murphy T., Webb N. (1957) The efficiency of N machines uni-directionally patrolled by one operative when walking time and repair times are constants. Journal of the Royal Statistical Society. Series B (Methodological) 19 (1): 166-172.

[28] Madu C. N. (1988) A closed queueing maintenance network with two repair centres. Journal of the Operational Research Society 39 (10): 959-967.

[29] Meilijson I., Yechiali U. (1977) On optimal right-of-way policies at a single-server station when insertion of idle times is permitted. Stochastic Processes and Their Applications 6 (1): 25-32.

[30] Perel E., Yechiali U. (2017) Two-queue polling systems with switching policy based on the queue that is not being served. Stochastic Models 33 (3): 1-21.

[31] Righter R. (2002) Optimal maintenance and operation of a system with backup components. Probability in the Engineering and Informational Sciences 16 (3): 339-349.

[32] Ross S. M. (2014) Introduction to probability models. Academic press, San Diego.

[33] Syski R. (1992) Passage Times for Markov Chains. IOS Press, Amsterdam.

[34] Takagi H. (1988) Queueing analysis of polling models. ACM Computing Surveys 20 (1): 5-28.

[35] Van Mieghem J. A. (1995) Dynamic scheduling with convex delay costs: The generalized $c\mu$ rule. The Annals of Applied Probability 5 (3): 809-833.

[36] Vishnevskii V.M. Semenova O.V. (2006) Mathematical methods to study the polling systems. Automation and Remote Control 67 (2): 173-220.

[37] Weststrate J. A., van der Mei R. D. (1994) Waiting times in a two-queue model with exhaustive and Bernoulli service. Zeitschrift für Operations Research 40 (3): 289-303.