A Constrained Optimization Problem for a Two-Class Queueing Model

Cory Girard[1], Linda V. Green[2], Mark E. Lewis[3], and Jingui Xie[4]

[1]*School of Operations Research and Information Engineering*

Cornell University

Ithaca, NY

cjg264@cornell.edu

[2]*Graduate School of Business*

Columbia University

New York, NY

lvg1@columbia.edu

[3]*School of Operations Research and Information Engineering*

Cornell University

Ithaca, NY

mark.lewis@cornell.edu

[4]*School of Management*

University of Science and Technology of China

Hefei, Anhui, China

xiej@ustc.edu.cn

February 1, 2017

Abstract

We discuss dynamic server control in a two-class service system under a constraint on the number of high-priority customers. A class of randomized threshold policies is defined, and is proven to contain an optimal policy in the case without abandonments. The proof of optimality is then used to construct heuristic policies for the case of low-priority abandonments, which we test numerically. The experiments we run suggest that, even when abandonments are introduced, these classes of policies outperform priority policies, and, in some cases, are near-optimal. Both cases are considered under the average cost criterion.

This paper makes two primary contributions. First, the proof of optimality in the case without abandonments provides an alternative method for proving the existence of optimal policies satisfying certain structural results without relying on leveraging properties of the value function. The second contribution lies in the practicality of the heuristic policies. While implementing a stationary policy generally relies on the cumbersome task of observing the number of customers of each class, the threshold structure of the heuristic policies only requires focus on a particular aspect of the system, such as total number of customers.

1 Introduction

Typically, in multi-class service systems, customer classes are differentiated by arrival rates, service requirements, patience times and either rewards or holding costs. When analyzing these systems, each class is modeled as having its own queue and an optimal scheduling policy for the server(s) is sought. Take for example, the classical problem of allocating servers to queues to minimize the sum of the long-run expected average holding costs. When the patience times of customers are infinite, if the holding costs and service rates at the i^{th} station are denoted c_i and μ_i, respectively, the well-known $c\mu$-rule is optimal. In short, one need only create an index $c_i\mu_i$ for each queue and choose the next non-empty queue with the highest index to allocate servers. The proof technique (an interchange argument) has been cross-applied to a wide range of scheduling problems (see e.g. [11]).

One difficulty with this formulation is that the relative importance captured by the holding costs of each customer class is not always easily quantified. An example is a hospital emergency department (ED) where both urgent and non-urgent patients seek treatment. In this setting, it is crucial to assure that urgent patients are served within a specified amount of time in order to avoid adverse consequences. At the same time it is also important to minimize waiting times for non-urgent patients who may leave the system before being treated if wait times are too long. In this case, it seems prudent to consider a constrained version of the control problem, rather than choosing holding costs for the unconstrained problem so as to drive down the average holding cost of the urgent class. This constrained problem is closer to the way practitioners approach the trade-off between prioritized classes of customers/patients. Of course, the model is also relevant in any service system (such as call centers) which places great importance on the timeliness of service for a particular type of customer.

Motivated by the ED example above, we consider a two-class service system with a constraint on the holding costs for one class and the possibility of abandonments from the other. We formulate this problem as a constrained Markov decision process (CMDP) and use the model without abandonments to develop implementable heuristics for the more general problem. To do so, we make several important observations. First, a technical challenge. A classic result for CMDPs with one constraint states that there exists an optimal policy that randomizes between actions in at most one state (a 1-randomized policy) [14]. For our problem, this means that in at most one state, the policy will flip a (potentially) biased coin to determine which class of customers to serve. In the rest of the states, the policy chooses which class to serve deterministically. This result works well when the state space is single-dimensional (or multi-dimensional with a single infinite dimension), since the search in one dimension is quite often a search for a threshold value (at which point we randomize). Our formulation requires two dimensions, each of which is infinite, making the search for a 1-randomized policy more difficult. Second, a practical challenge. The 2-dimensional state space usually means the decision-maker (a medical service provider in the health care example) needs visibility into both dimensions. The heuristics we develop are more consistent with the single dimensional search and require that the decision-maker monitor only one dimension and then randomize at that threshold (no matter the value of the other dimension). This simplifies both the search and implementation. In most of the cases we consider, we show that the heuristics lead to policies that are within one percent of optimal. Moreover, it turns out that we are also able to show that our heuristic is optimal in the model without abandonments; which may be of independent interest.

The remainder of the paper is arranged as follows. Section 2 discusses related literature. In Section 3, we formulate the problem as a CMDP and review some preliminary results from Altman [1], which we use to develop sufficient optimality conditions. In Section 4, we present our major technical results, motivating the construction of the heuristic policies in Section 4.2. We define classes of randomized-threshold policies and prove that this class contains an optimal policy for the constrained problem in the case of no abandonments in Section 5. Section 6 covers numerical experiments testing the effectiveness of such policies when class 2 customers may abandon the system and includes a study of the $N-$server system numerically. Finally, Section 7 summarizes our findings and discusses future directions.

2 Related Literature

Our problem is closely related to the unconstrained problem (in the same setting) of minimizing the expected weighted total cost across both classes. In the absence of abandonments, structural results are given by the $c\mu$-rule in Buyukkoc et al. [11]. When abandonments are introduced, obtaining structural results becomes considerably more difficult. Two reasons for this are that interchange arguments as found in Nain [20] and the uniformization technique in Lippman [19] and Serfozo [23] are no longer applicable. There are two common approaches to deal with the difficulty of proving structural results in the presence of abandonments: (1) considering a subset of the parameter space in which structural results can be obtained, and (2) considering asymptotic performance of policies. In Down, Koole and Lewis [13], the former approach is taken. The authors consider parallel queues where both classes of customers may abandon. When the service rates are equal, they provide conditions under which a priority policy is optimal. These conditions mimic that of the $c\mu$-rule, with an additional condition that the abandonment rates of each class. The second approach (asymptotic analysis) can be further divided into two categories. When focusing on the overloaded regime, the fluid model approach is taken. In Atar et al. [7], the authors show that a generalization of the $c\mu$-rule is asymptotically optimal under a many-server fluid scaling for a general parallel queueing system with (possibly) more than two classes of customers. In the critically loaded regime a diffusion model is used in Ghamami and Ward [16],[17], Harrison and Zeevi [18], and Tezcan and Dai [24]. Arapostathis et al. [2] consider a limiting problem that is similar to the unconstrained problem we consider. Ward and Glynn [26] [27] are concered with approximating single-class systems with abandonments by a regulated Ornstein-Uhlenbeck process in heavy traffic. Surveys of fluid and diffusion approximations for queues with abandonments can be found in Dai and He [12] and Ward [25].

Other similar unconstrained problems have been considered. Argon et al. [3] considers the unconstrained problem for a related clearing system with abandonments from both classes. Ayesta et al. [8] further extends work in this direction, proving the optimality of an index rule for the scheduling problem with 1 or 2 customers in the system. For more customers, the authors derive a nearly-optimal index rule which recovers the $c\mu$-rule and coincides with the $\frac{c\mu}{\theta}$-rule under certain conditions. Argon et al. [4] and Armony and Maglaras [5] [6] consider the unconstrained problem, but where customers are given a call-back option to influence behavior.

A constrained optimization problem applied to call centers is considered by Gans and Zhou in [15]. The general setting they consider is similar to that considered here. With a few simplifying assumptions like an infinite backlog of class 2 customers and without abandonments. In the single server case, their assumptions allow for a single-dimensional state space as discussed above (with multiple servers, there is a finite search along that dimension as well). Furthermore, their objective is to maximize the rate at which class 2 customers receive service, rather than minimizing the number of class 2 customers in system. Their problem, like ours, places a constraint on the number of class 1 customers in system. In addition to considering a different system, Gans and Zhou [15] obtain structural results only in the case where the service rates of the two classes are equal.

The work of Bhulai [10] independently obtains results for the same problem as considered by [15], although the approach taken by the latter is closer to the one we consider. Berman et al. [9] considers a system similar to that studied in [15], focusing on a class of "switching point" policies. By focusing on this class of policies, they were able to obtain explicitly the long-run average number of class 2 customers in the system. They then considered two constrained optimization problems: (1) for a fixed number of workers, minimizing the expected waiting time of front-room jobs while meeting a service level requirement for back-room, and (2) minimizing the number of workers while meeting service level constraints for both classes of jobs. Yang et al. [28] considers general multiclass systems with no abandonments in which each class is differentiated by its arrival rate and cost function that varies in the number of workers assigned to work on class each class. For the problem of allocating workers to customers in such a way so as to minimize cost subject to quality of service (waiting time) constraints for each class of customer, the authors use an MDP value function approach to prove the following structural result for the optimal policy: if the number of customers for a particular class increases, then assign more servers to work on that class.

3 Preliminaries and Model Formulation

In this section, we formalize the dynamics of the system and formulate the decision problem as a CMDP. We then apply results in CMDP theory from [1] to develop sufficient optimality conditions.

3.1 System Dynamics

Suppose there are two classes of customers that arrive to a service system staffed by a single server. Class k ($k = 1, 2$) customers arrive to the system according to independent Poisson processes with rate $\lambda_k > 0$. The system is charged cost h_k per unit time class k customers are in the system. Class 2 customers have a patience time that is exponentially distributed with rate $\beta_2 \geq 0$, after which they leave the system charging a penalty of P_2. We consider the case where customers currently in service may abandon if they run out of patience before service is complete. Customer service requirements are exponentially distributed with rate 1, and are also independent of all else. The server can work on class k customers at rate $\mu_k > 0$. See Figure 3.1. Our objective is to create a schedule for the server that minimizes the long-run weighted (by $h_2 > 0$) average number of customers at station 2 while keeping the long-run weighted (by $h_1 > 0$) average number of customers at station 1 under a given threshold denoted by V.

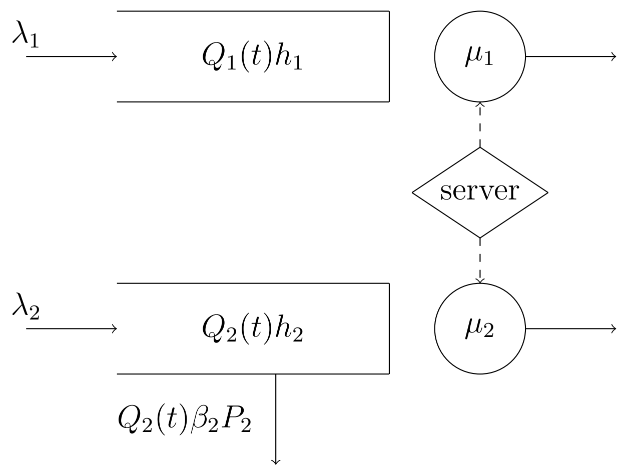


Figure 1: A two class queueing system with a single server. $Q_k(t)$ denotes the number of class k customers at time t.

3.2 MDP Formulation

We model this problem as a CMDP, with countable state space $\mathbb{X} := \mathbb{Z}^+ \times \mathbb{Z}^+$, where the k^{th} component corresponds to the number of class k customers in the system. We model the action space as

$$A(i,j) = \begin{cases} [0,1] & i,j > 0, \\ \{1\} & i > 0, j = 0, \\ \{0\} & i = 0, j > 0, \\ \{-1\} & \text{otherwise}, \end{cases}$$

where $a = -1$ is a "dummy" action to denote idling when the system is empty, and action $a \in [0,1]$ represents serving class 1 with probability a and serving class 2 with probability $1-a$. The transition rates are, for $i,j > 0$:

$$G((k,\ell)|(i,j),a) = \begin{cases} \lambda_1 & (k,\ell) = (i+1,j), \\ \lambda_2 & (k,\ell) = (i,j+1), \\ (1-a)\mu_2 + j\beta_2 & (k,\ell) = (i,j-1), \\ a\mu_1 & (k,\ell) = (i-1,j), \\ -\lambda_1 - \lambda_2 - j\beta_2 - a\mu_1 - (1-a)\mu_2 & (k,\ell) = (i,j), \\ 0 & \text{otherwise}. \end{cases}$$

We note that we do not allow unforced idling, making the transition rates for $i > 0 = j$:

$$G((k,\ell)|(i,j),a) = \begin{cases} \lambda_1 & (k,\ell) = (i+1,0) \\ \lambda_2 & (k,\ell) = (i,1) \\ \mu_1 & (k,\ell) = (i-1,0) \\ -\lambda_1 - \lambda_2 - \mu_1 & (k,\ell) = (i,j) \\ 0 & \text{otherwise.} \end{cases}$$

Similarly for $i = 0 < j$ and $i = j = 0$. We make several observations. First, the instantaneous cost rate where there are i customers at station 1 and j customers at station 2 is $ih_1 + j(h_2 + \beta_2 P_2)$. In terms of computing optimal controls we may define $\widehat{h_2} = h_2 + \beta_2 P_2$. Second, since long-run average holding costs are directly proportional to the long-run average number of customers in the system, we assume (without loss of generality) that the per-unit holding costs of each class are 1; $h_1 = \widehat{h_2} = 1$. Define the immediate cost functions by

$$c_1(i,j) = i \qquad\qquad c_2(i,j) = j.$$

A stationary policy σ chooses an action $a_\sigma(x) \in A(x)$ for each $x \in \mathbb{X}$. We let Π^S denote the class of stationary policies. Letting $\{X^\sigma(t) : t \geq 0\}$ denote the Markov process induced by stationary policy σ, we can express the long-run average number of class k customers in the system (for $k = 1, 2$) as:

$$C_k(\sigma) := \limsup_{T\to\infty} \mathbb{E}\left[\frac{1}{T}\int_0^T c_k\left(X^\sigma(t)\right) dt\right].$$

Given a quality of service level, V, we formulate the control problem, denoted by $B(V)$, as

$$\inf_{\sigma\in\Pi^S} \{C_2(\sigma) : C_1(\sigma) \leq V\}. \tag{B(V)}$$

It is shown in Altman [1] that Π^S is a dominating class of policies for CMDPs; the search for an optimal policy in the broader class of non-anticpating policies for B(V) can be restricted to Π^S without loss of optimality. We let C_V denote the optimal value of B(V).

3.3 Assumptions

In this section, we state our assumptions and explain their significance. For most of the paper we consider the **single-server model** (the $N-$server system is studied numerically in Section 6.6). For $N \geq 2$, in the $N-$server case suppose a single server can work at rate μ. This simplification is justified in two cases:

1. When the number of customers of each class is greater than or equal to N, all servers are busy and the service rate is $N\mu$. The linearity of the service rates implies it is not optimal to split the servers between stations. In a system with a reasonable workload (either medium or high), a large proportion of time in the long run is spent in these states. Thus, a single-server proxy with service rate $\tilde{\mu} = N\mu$ is reasonable. To round out the policy in the original system, we simply ensure servers do not idle unnecessarily.

2. When the collaboration rate of servers is additive, N servers collaborating work at rate $N\mu$. Class 1 customers are given higher priority, and the objective is to minimize the long-run average class 2 holding cost, while keeping that of the class 1 customers under a given level.

Next we state the traffic assumption:

$$\textbf{Traffic Assumption: } \rho := \frac{\lambda_1 + \lambda_2}{\min(\mu_1, \mu_2)} < 1. \tag{T}$$

Under Assumption (T), it can be verified (via a *Lyapunov* function argument) that any stationary Markov policy induces a positive-recurrent Markov process with a unique stationary distribution. For a stationary policy $\sigma \in \Pi^S$, let π^σ denote this stationary distribution. The long-run average number of class k customers in the system under policy σ is

$$C_k(\sigma) := \sum_{i,j \in \mathbb{X}} c_k(i,j)\pi^\sigma(i,j).$$

In addition to giving us a more convenient way to express $C_1(\cdot)$ and $C_2(\cdot)$, Assumption (T) implies that $C_1(\cdot)$ and $C_2(\cdot)$ are bounded over the class of stationary policies. That is,

$$\max\{\sup_{\sigma \in \Pi^S} C_1(\sigma), \sup_{\sigma \in \Pi^S} C_2(\sigma)\} \leq B$$

for some finite $B > 0$. This can be seen by considering an $M/M/1$ queueing system with birth rate $\lambda_1 + \lambda_2$ and death rate $\min\{\mu_1, \mu_2\}$. Comparing this system to the total number of customers in the two station system under any stationary policy. Note that the $M/M/1$ system has more customers in the system than the orignal process since it sees no abandonments and serves one of the classes of customers at a (potentially) slower rate than it would in the original system. By Assumption (T) and standard queueing arguments, the system has a finite expected long-run average number of total customers in the system, and hence a finite number for each class.

Next, consider the two extremal policies that prioritize station 1 and 2 except to avoid idling; denoted P_1 and P_2, respectively. We make the following assumption on V:

$$\textbf{RHS Assumption: } V \in (C_1(P_1), C_1(P_2)). \tag{RHS}$$

Notice that P_2 spends the least amount of time working at station 1. Thus, in the case that $V \geq C_1(P_2)$, the problem B(V) is unconstrained and it is optimal to prioritize class 2 customers. Alternatively, if $V \leq C_1(P_1)$, the problem is infeasible unless $V = C_1(P_1)$, in which case P_1 is optimal. Any V in this range we refer to as feasible and non-trivial.

3.4 Lagrangian Dual and Optimality Conditions

We derive sufficient optimality conditions for $B(V)$. To do so, we establish the equivalence of the constrained problem $B(V)$ and its Lagrangian dual, L(V), given by

$$\sup_{\gamma \geq 0} \inf_{\sigma} \{C_2(\sigma) + \gamma(C_1(\sigma) - V)\}. \tag{L(V)}$$

We establish the equivalence between B(V) and L(V) by applying Theorem 12.7 from [1], summarized below in the context of our problem.

Theorem 3.1 (adapted from Thereom 12.7 in Altman [1])

1. *The optimal value C_V of the problem $B(V)$ can be computed, $C_V = \inf_{\sigma} \sup_{\gamma \geq 0} \{C_2(\sigma) + \gamma(C_1(\sigma) - V)\}$.*
2. *A policy σ^* is optimal for $B(V)$ if and only if*

$$C_V = \sup_{\gamma \geq 0} \{C_2(\sigma^*) + \gamma(C_1(\sigma^*) - V)\}$$

3. *Suppose Π^D is the set of all stationary, deterministic policies. For any class of policies Π such that $\Pi^D \subseteq \Pi$,*

$$C_V = \sup_{\gamma \geq 0} \min_{\sigma \in \Pi} \{C_2(\sigma) + \gamma(C_1(\sigma) - V)\},$$

where we can take $\Pi = \Pi^S$, the set of all stationary policies.

Statement 3 of Theorem 3.1 deserves some further comment. Recall that C_V is the value of the problem B(V). The policy that achieves the minimum at the supremum over γ may be an optimal policy. On the other hand, it is possible that the policy achieving the value C_V is not feasible; Statement 3 allows for a method to compute C_V. Verification of the assumptions needed to apply this theorem is completed in Appendix A. Given Theorem 3.1, note that we can now rewrite

$$C_V = \sup_{\gamma \geq 0} \{ \min_{\sigma \in \Pi^S} \{\gamma C_1(\sigma) + C_2(\sigma)\} - \gamma V \}. \tag{3.1}$$

For notational convenience, define $g(\gamma) := \min_{\sigma \in \Pi^S} \{\gamma C_1(\sigma) + C_2(\sigma)\} - \gamma V$. Note that the minimization in (3.1) is an unconstrained Markov decision process with the immediate cost function $\tilde{c}(i, j) = \gamma i + j$. Thus, the minimum is attained by a stationary policy by Theorem 5.5.1 in [21].

For $\gamma \in \mathbb{R}$, define

$$O_\gamma := \operatorname{argmin}_{\sigma \in \Pi^S} \{\gamma C_1(\sigma) + C_2(\sigma)\}$$

to be the set of all policies that are optimal for the unconstrained problem with class 1 per-unit holding cost γ. The following lemma is useful in the derivation of optimality conditions.

Lemma 3.2 *The following hold for all $\gamma \in \mathbb{R}$*

1. *$g(\gamma)$ is concave in γ.*
2. *For any $\sigma_\gamma \in O_\gamma$, $V - C_1(\sigma_\gamma) \in \partial(-g)(\gamma)$, where ∂f is the subdifferential (set of all subgradients) of the function f.*
3. *If $\gamma < \hat{\gamma}$, and $\sigma_\gamma \in O_\gamma$ and $\sigma_{\hat{\gamma}} \in O_{\hat{\gamma}}$, then $C_1(\sigma_\gamma) \geq C_1(\sigma_{\hat{\gamma}})$.*

Proof. Note that since sums of concave functions are concave, and the minimum of concave functions is concave the first result holds. To show the second result we need to show that for any $\gamma \in \mathbb{R}, \sigma_\gamma \in O_\gamma$,

$$-g(\gamma_0) \geq -g(\gamma) + (V - C_1(\sigma_\gamma))(\gamma_0 - \gamma) \qquad \forall \gamma_0 \in \mathbb{R}.$$

Fix $\gamma \in \mathbb{R}$. We have, for any $\gamma_0 \in \mathbb{R}$,

$$\begin{aligned} g(\gamma_0) - g(\gamma) &= \min_{\sigma \in \Pi^S} \{\gamma_0 C_1(\sigma) + C_2(\sigma)\} - \min_{\sigma \in \Pi^S} \{\gamma C_1(\sigma) + C_2(\sigma)\} - V(\gamma_0 - \gamma) \\ &= \min_{\sigma \in \Pi^S} \{\gamma_0 C_1(\sigma) + C_2(\sigma)\} - \gamma C_1(\sigma_\gamma) - C_2(\sigma_\gamma) - V(\gamma_0 - \gamma) \\ &\leq \gamma_0 C_1(\sigma_\gamma) + C_2(\sigma_\gamma) - \gamma C_1(\sigma_\gamma) - C_2(\sigma_\gamma) - V(\gamma_0 - \gamma) \\ &= -(V - C_1(\sigma_\gamma))(\gamma_0 - \gamma). \end{aligned}$$

Hence

$$-g(\gamma_0) \geq -g(\gamma) + (V - C_1(\sigma_\gamma))(\gamma_0 - \gamma),$$

as desired.

For the remaining result, fix $\gamma \in \mathbb{R}$ and let $\delta > 0$. Let $\nu \in O_\gamma$ and $\hat{\nu} \in O_{\gamma+\delta}$. This implies

$$\begin{aligned}\gamma C_1(\nu) + C_2(\nu) &\leq \gamma C_1(\hat{\nu}) + C_2(\hat{\nu}) \\ (\gamma + \delta) C_1(\hat{\nu}) + C_2(\hat{\nu}) &\leq (\gamma + \delta) C_1(\nu) + C_2(\nu).\end{aligned}$$

Using the fact that $A \leq B$ and $C \leq D$ implies $C - B \leq D - A$ yields

$$\delta(C_1(\hat{\nu}) - C_1(\nu)) \leq 0,$$

so that $C_1(\hat{\nu}) \leq C_1(\nu)$. ∎

Using this result in combination with Theorem 3.1 leads to the following sufficient optimality conditions.

Lemma 3.3 *(Sufficient optimality conditions) Suppose that* $(\sigma^*, \gamma^*) \in \Pi^S \times \mathbb{R}^+$ *satisfies*

$$\sigma^* \in O_{\gamma^*} \tag{3.2}$$

$$C_1(\sigma^*) = V. \tag{3.3}$$

The policy σ^* *is optimal for B(V).*

Proof. It follows immediately from (3.2) and (3.3) that $0 \in \partial(-g)(\gamma^*)$. Thus,

$$C_V = \sup_{\gamma \geq 0} g(\gamma) = g(\gamma^*) = \min_{\sigma \in \Pi^S} \{\gamma^* C_1(\sigma) + C_2(\sigma)\} - V\gamma^*.$$

Using (3.2) and (3.3) yields

$$\begin{aligned}C_V &= \gamma^* C_1(\sigma^*) + C_2(\sigma^*) - V\gamma^* \\ &= \sup_{\gamma \geq 0}\{C_2(\sigma^*) + \gamma(C_1(\sigma^*) - V)\}.\end{aligned}$$

Hence, by Statement 2 of Theorem 3.1, σ^* is optimal for B(V). ∎

3.5 Cost continuity

The goal of this section is to prove that the long-run average costs for each class are pointwise continuous over the set of stationary policies. This is used to prove some of the structural results in the next section. The result is stated more precisely in the following theorem.

Theorem 3.4 *Suppose $\{\sigma_n, n \geq 0\} \subseteq \Pi^S$ is a sequence of policies such that $\sigma_n(x) \to \sigma(x)$, for each $x \in \mathbb{X}$, where $\sigma \in \Pi^S$. For $k = 1, 2$, $\lim_{n\to\infty} C_k(\sigma_n) = C_k(\sigma)$.*

Proving Theorem 3.4 requires two steps: first, applying Theorem 11.2 from Altman [1] to prove a result on stationary distribution convergence, and second, showing a uniform convergence result. For the sake of clarity, we state the stationary distribution result and leave the technical details to Appendix A. The next two lemmas summarize the two steps needed to prove Theorem 3.4.

Lemma 3.5 *Let σ be a stationary policy, and let $\{\sigma_n, n \geq 0\}$ be a sequence of stationary policies. Suppose that $\sigma_n \to \sigma$ pointwise. Let π^{σ_n} and π^{σ} denote the stationary distributions of the respective induced (Markov) processes. We have $\pi^{\sigma_n} \to \pi^{\sigma}$ pointwise.*

Proof. See Appendix A. ∎

For an arbitrary stationary policy $\sigma \in \Pi^S$, define the truncated costs for non-negative integer N,

$$C_1^N(\sigma) := \sum_{(i,j):i+j\leq N} i\pi^{\sigma}(i,j) \qquad\qquad C_2^N(\sigma) := \sum_{(i,j):i+j\leq N} j\pi^{\sigma}(i,j).$$

Lemma 3.6 *Given any $\epsilon > 0$, there exists $N \in \mathbb{Z}^+$ such that*

$$C_k(\sigma) - C_k^N(\sigma) < \epsilon$$

simultaneously for every $\sigma \in \Pi^S$, for $k = 1, 2$.

Notice that Lemma 3.6 states that the truncated costs converge uniformly in the class of stationary policies.

Proof. We prove the result for $k = 1$ by contradiction. The proof for $k = 2$ is analogous. Suppose not. There exists $\tilde{\epsilon} > 0$ such that, given any $N \in \mathbb{Z}^+$, there exists a policy $\sigma_N \in \Pi^S$ satisfying

$$C_1(\sigma_N) - \sum_{(i,j):i+j\leq N} i\pi^{\sigma_N}(i,j) \geq \tilde{\epsilon}. \tag{3.4}$$

Let $D(N) := \{\sigma \in \Pi^S : C_1(\sigma) - \sum_{(i,j):i+j\leq N} i\pi^{\sigma}(i,j) \geq \tilde{\epsilon}\}$ denote the set of policies such that (3.4) holds at N. Note that since the summands are non-negative, we have that $D(N) \subseteq D(N-1) \subseteq ... \subseteq D(1)$. Hence, (3.4) (and the discussion immediately preceding) can be rewritten

$$D := \bigcap_{N\in\mathbb{Z}^+} D(N) \neq \emptyset.$$

Now note that for any $\hat{\sigma} \in D$, we have

$$C_1(\hat{\sigma}) - \sum_{(i,j):i+j\leq N} i\pi^{\hat{\sigma}}(i,j) \geq \tilde{\epsilon}$$

for all $N \in \mathbb{Z}^+$, and hence $\sum_{(i,j):i+j\leq N} i\pi^{\hat{\sigma}}(i,j) \not\to C_1(\hat{\sigma})$, a contradiction. To see this, note that since every stationary policy has finite average cost, $\sum_{(i,j)\in\mathbb{X}} i\pi^{\hat{\sigma}}(i,j)$ is absolutely convergent, and so rearrangement of the terms of the series does not affect its value. Hence $C_1(\hat{\sigma}) = \lim_{N\to\infty} \sum_{(i,j):i+j\leq N} i\pi^{\hat{\sigma}}(i,j)$. ∎

Lemmas 3.5 and 3.6 imply Theorem 3.4. The proof is omitted for brevity, and can be found in Appendix A. The implications of this result allow us to find policies that satisfy the constraint at equality. We show that this condition is sufficient for optimality in the case without class 2 abandonments. While we have not been able to extend these results to the case with abandonments, we use this result as motivation to construct highly-structured heuristic policies that perform well numerically.

4 The case without abandonments

We use the continuity results established in the previous section to construct and prove the optimality of a general class of threshold policies in the case without abandonments; assume $\beta_2 = 0$ throughout this section.

4.1 Simplification of the optimality conditions

In this special case, we first simplify the optimality conditons (3.2) and (3.3) by leveraging the well-known $c\mu$ rule. Define the priority policies P_k for $k = 1, 2$, where P_k denotes the policy that prioritizes class k. That is, policy P_k serves exhaustively at station k, switching stations only if it is empty to avoid idling. Recall that the Lagrangian dual problem, L(V), can be rewritten as

$$\sup_{\gamma\geq 0}\{\min_{\sigma}\{\gamma C_1(\sigma) + C_2(\sigma)\} - \gamma V\},$$

and that the minimization is an unconstrained MDP with immediate cost function $\gamma i + j$ when in state (i, j). That is to say, the problem within the minimum is the classic scheduling problem that we know has an optimal contol, the $c\mu$ rule: if $\gamma \geq \frac{\mu_2}{\mu_1}$, then P_1 is optimal, and if $\gamma \leq \frac{\mu_2}{\mu_1}$, then P_2 is optimal. Hence we have

$$\begin{aligned}
C_V &= \max\left(\sup_{\gamma\in\left[0,\frac{\mu_2}{\mu_1}\right]}\{\min_{\sigma\in\Pi^S}\{\gamma C_1(\sigma) + C_2(\sigma)\} - \gamma V\}, \sup_{\gamma\geq\frac{\mu_2}{\mu_1}}\{\min_{\sigma\in\Pi^S}\{\gamma C_1(\sigma) + C_2(\sigma)\} - \gamma V\}\right)\\
&= \max\left(\sup_{\gamma\in\left[0,\frac{\mu_2}{\mu_1}\right]}\{C_2(P_2) + \gamma(C_1(P_2) - V)\}, \sup_{\gamma\geq\frac{\mu_2}{\mu_1}}\{C_2(P_1) + \gamma(C_1(P_1) - V)\}\right)\\
&= \max\left(C_2(P_2) + \frac{\mu_2}{\mu_1}(C_1(P_2) - V), C_2(P_1) + \frac{\mu_2}{\mu_1}(C_1(P_1) - V)\right)\\
&= \max\left(\frac{\mu_2}{\mu_1}C_1(P_2) + C_2(P_2), \frac{\mu_2}{\mu_1}C_1(P_1) + C_2(P_1)\right) - \frac{\mu_2}{\mu_1}V,
\end{aligned}$$

where the third equality follows from the assumption that $V \in (C_1(P_1), C_1(P_2))$. Since the supremum is attained at $\gamma = \frac{\mu_2}{\mu_1}$, the previous discussion implies that both P_1 an P_2 are optimal; the two terms in the maximum are equal. Now note that in the case without abandonments Condition (3.2) becomes $\sigma^* \in O_{\frac{\mu_2}{\mu_1}}$. We proceed to show that we can eliminate this condition by proving that $O_{\frac{\mu_2}{\mu_1}} = \Pi^S$. In doing so, we show that to find an optimal policy for B(V) in the case with no abandonments, it suffices to find a binding policy.

To show that $O_{\frac{\mu_2}{\mu_1}} = \Pi^S$, consider the α-discounted version of the problem for arbitrary $\alpha \in (0, 1)$ and consider its (unique) value function, v. Define the interior of the state space $\mathbb{X}$,

$$\widehat{\mathbb{X}} := \{(i, j) \in \mathbb{X} \mid i, j \geq 1\}.$$

Consider the discounted-cost optimality equations for the scheduling problem implied by $O_{\frac{\mu_2}{\mu_1}}$ for a fixed state $(i,j) \in \widehat{\mathbb{X}}$,

$$
\begin{aligned}
v(i,j) = \frac{\mu_2}{\mu_1} i + j + [&\lambda_1 v(i+1,j) + \lambda_2 v(i,j+1) + (1-\lambda_1-\lambda_2) v(i,j) \\
&+ \min_{a \in [0,1]} \{a\mu_1(v(i-1,j) - v(i,j)) + (1-a)\mu_2(v(i,j-1) - v(i,j))\}]. \qquad (4.1)
\end{aligned}
$$

Since the holding cost in the first station for this problem is $\frac{\mu_2}{\mu_1}$ the $c\mu$ index is $\mu_1 \frac{\mu_2}{\mu_1} = \mu_2$. Similarly, the index for the second station is $\mu_2(1)$. Thus, the $c\mu$-rule result from [11] states that for this problem, both P_1 and P_2 are optimal. This implies that the minimum on the right-hand side of (4.1) is attained at both $a = 0$ and $a = 1$. Noting that the quantity inside the minimization term is linear in a, we conclude that any action is optimal. Since the action space is a singleton in states (i,j) where $i = 0$ or $j = 0$, it follows that every stationary policy is optimal for the α-discounted problem, for every $\alpha \in (0,1)$. In particular, every stationary policy is Blackwell optimal, and therefore average-cost optimal [22]. Hence, applying Lemma 3.3 to find an optimal policy for B(V), it suffices to find a stationary policy σ^* with $C_1(\sigma^*) = V$.

4.2 Optimal control policies

We now define a class of threshold policies that we prove contains an optimal policy for any $V \in (C_1(P_1), C_1(P_2))$. This class of policies has the special property of containing optimal policies that randomize on general subsets of the state space. As alluded to earlier, this differs from the classic theory of constrained MDPs (see [1],[14]), that explains the existence of optimal policies that randomize in one state. When the state space is multidimensional (as in the present study) finding such a state may be difficult. Most importantly, in the hospital application, implementing said policy is impractical. The existence of optimal policies in the more general class simplifies our search from multidimensional (finding a single state to randomize in) to single-dimensional (finding a subset of the state space to randomize in).

Definition 4.1 *Given a sequence of sets $G = \{G_n, n \geq 0\}$ satisfying $G_0 = \emptyset$ and $G_n \uparrow \widehat{\mathbb{X}}$, we define the class of* ***threshold policies*** *with respect to G by*

$$\Pi^G := \{\sigma_n : a_{\sigma_n}(x) = \mathbb{1}_{G_n^C}(x), n \in \mathbb{Z}^+, x \in \mathbb{X}\}.$$

Similarly, for $p \in [0,1]$, we define the set of ***p-randomized threshold policies*** *with respect to G by*

$$\Pi^{G,p} := \{\sigma_{n,p} : a_{\sigma_{n,p}}(x) = p\mathbb{1}_{G_n \setminus G_{(n-1)^+}}(x) + \mathbb{1}_{G_n^C}(x), n \in \mathbb{Z}^+, x \in \mathbb{X}\}.$$

It is worth noting that for any $p \in [0,1]$, $\sigma_{0,p} = P_1$ and $\sigma_{n,p} \to P_2$ pointwise. A simple example for G_n are the (half open) rectangles $G_n = \{(i,j) \in \widehat{\mathbb{X}} \mid j \leq n\}$ so that the decision-maker works at station 1 as long as there are less than n people at station 2 and at station 2 otherwise. The main result is summarized in the theorem below.

Theorem 4.2 *For any $V \in (C_1(P_1), C_1(P_2))$, any sequence of sets $G = \{G_n, n \geq 0\}$ satisfying $G_0 = \emptyset$ and $G_n \uparrow \widehat{\mathbb{X}}$, there exists $p \in [0,1]$ and a $p-$randomized threshold policy, $\sigma^* \in \Pi^{G,p}$ that is optimal for B(V).*

Before we proceed, we need one small result.

Lemma 4.3 *Let $\{x_n, n \geq 0\}$ be a real sequence of numbers such that $x_n \uparrow x \in \mathbb{R}$. For any $v \in [x_0, x)$, there exists $m \in \mathbb{Z}_+$ such that $v \in [x_m, x_{m+1})$.*

Proof. The result is trivial if $v = x_0$. Assume $v > x_0$. Since $x_n \uparrow x$, there exists m so that $x_m > v > x_0$. If $x_m = v$, the result follows since $v \in [x_m, x_{m+1}]$. Otherwise, decrease m in increments of one until $x_m \leq v$. The algorithm is guaranteed to terminate since $x_0 < v$, and results in $v \in [x_m, x_{m+1})$, as desired. ∎

Proof of Theorem 4.2. Fix arbitrary $V \in (C_1(P_1), C_1(P_2))$ and the sequence of sets $G = \{G_n, n \geq 0\}$ satisfying the conditions stated in the theorem. Since $\sigma_0 = P_1$ and $\sigma_n \to P_2$ pointwise, Theorem 3.4 yields that $C_1(\sigma_n) \to C_1(P_2) \geq C_1(P_1) = C_1(\sigma_0)$, where the inequality holds since the policy that prioritizes station 1 has the lowest average cost associated with station 1 of all stationary policies. By Lemma 4.3 and our assumption on V, there exists $N \in \mathbb{Z}^+$ so that $V \in [C_1(\sigma_N), C_1(\sigma_{N+1}))$. Now note that $\sigma_N = \sigma_{N,0}$ is the $p-$randomized threshold policy at N with $p = 0$ as is $\sigma_{N+1} = \sigma_{N,1}$ with $p = 1$. Moreover, $\sigma_{N,p} \to \sigma_{N+1}$ as $p \uparrow 1$. Thus, again by Theorem 3.4, $C_1(\sigma_{N,p})$ is continuous in p on $[0,1]$, with $C_1(\sigma_{N,0}) < C_1(\sigma_{N,1})$. Applying the intermediate value theorem yields the existence of $p^* \in (0,1)$ such that $C_1(\sigma_{N,p^*}) = V$. Clearly, $\sigma_{N,p^*} \in \Pi^{G,p^*}$, and by Lemma 3.3 it is optimal for B(V). ∎

5 The case with abandonments

We now return to the case where $\beta_2 > 0$. Unlike the case where $\beta_2 = 0$, we can no longer simplify the Condition (3.2). Hence, a policy that satisfies the constraint at equality may fail to achieve the optimal class 2 cost. Furthermore, as discussed in [13], the structure of an optimal policy for $\min_\sigma \left\{\frac{\mu_2}{\mu_1} C_1(\sigma) + C_2(\sigma)\right\}$, even when the service rates are equal is difficult to obtain. Thus, the analysis performed in the previous section no longer holds.

5.1 Heuristic Policies

The construction of optimal policies introduced in Section 4.2 suggests a family of heuristics for constructing policies that meet the condition (3.3):

1. Choose an increasing sequence of sets $\{G_k, k \geq 0\}$ with $G_k \uparrow \widehat{\mathbb{X}}$ and $G_0 = \emptyset$.

2. Define for each $k \in \mathbb{Z}_+$ the policy σ_k as follows:

$$a_{\sigma_k}(x) = \begin{cases} 0 & x \in G_k, \\ 1 & x \in G_k^C. \end{cases}$$

   We have that $C_1(\sigma_0) = C_1(P_1) \leq V$.

3. Iterate over k until we reach k such that $C_1(\sigma_k) \leq V < C_1(\sigma_{k+1})$.

4. Perform left bisection search to find p yielding $C_1(\sigma_{k,p}) > V - \epsilon$ for desired accuracy $\epsilon > 0$.

In particular, we are interested in cases such that $G_k = \{(i,j) \in \mathbb{X} : i \leq k\}$ (the "vertical threshold" class of policies), $G_k = \{(i,j) \in \mathbb{X} : j \leq k\}$ (the "horizontal threshold" class), and $G_k = \{(i,j) \in \mathbb{X} : i + j \leq k\}$ (the "total thresholds"). These classes are of particular interest since they reduce the amount of information needed in order to make a decision.

6 Numerical Experiments

Numerical experiments are performed on five parameter sets to test the effectiveness of the horizontal, vertical, and total classes of heuristic policies against that of the priority policies (those most likely to be implemented in a hospital setting). All experiments used the truncated state space $\mathbb{X}_{100} := \{0, 1, ..., 100\}^2$, and with the abandonment rate varying in the range $[0, 0.1]$ in increments of 0.002. The truncation of the state space allows us to calculate costs for a given policy by solving a sparse linear system. For each parameter set, we choose three values of V as follows: we first calculate the class 1 costs for P_1 $(C_1(P_1, \beta_2))$ and P_2 $(C_1(P_1, \beta_2))$ for each value of β_2. Note the added dimension to the nomenclature for the dependence on the class 2 abandonment rate. Letting $a = \max_{\beta_2} C_1(P_1, \beta_2)$ and $b = \min_{\beta_2} C_1(P_2, \beta_2)$, we use the values:

$$V_{low} = 0.75a + 0.25b$$
$$V_{med} = 0.5a + 0.5b$$
$$V_{high} = 0.25a + 0.75b$$

as our choices for V. Given each of these values of V, we find the optimal value for the (truncated) constrained problem by solving the dual LP. We then perform the algorithm described in Section 5, with a left bisection search to find p so that the class 1 cost of each heuristic policy is larger than $V - 0.0001$. Parameter set 1 below serves as the baseline case, and subsequent parameter sets are altered to explore the effects of heavier traffic and/or unequal service rates.

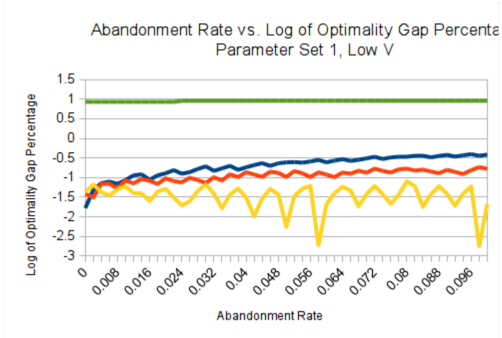
6.1 Parameter set 1

The first parameter set has $\lambda_1 = 0.2, \lambda_2 = 0.1, \mu_1 = \mu_2 = 1$; the load is relatively light for both classes. Table 1 below summarizes the feasibility gaps of the heuristic policies with that of the policy that prioritizes class 2. The minimums and maximums are taken with respect to the abandonment rate. Note that a negative number implies that the policy is feasible, and the minimum gap means that it is the furthest from the bound V. A positive gap implies that the policy is infeasible with the maximum (minimum) being the furthest (closest) from the upper bound V. The class (horizontal (h), vertical (v), or total (t)) of heuristic policy that attains the minimum feasibility gap (over all abandonment rates) is noted in parentheses.

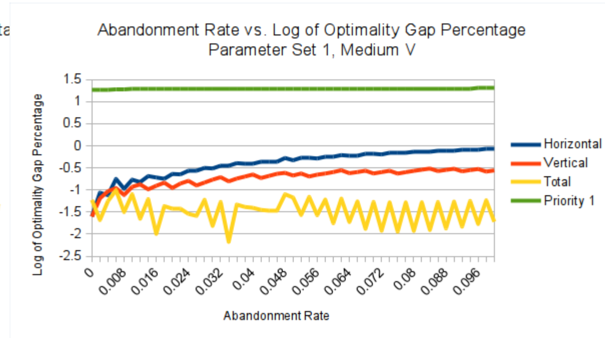
V	Min Heuristic Feas. Gap	Min Priority 2 Feas. Gap	Max Priority 2 Feas. Gap
0.2641	-0.0374 % (h)	16.05 %	20.19 %
0.2783	-0.0357 % (h)	10.16 %	14.09 %
0.2924	-0.0339 % (h)	4.83 %	8.57 %

Table 1: Baseline case.

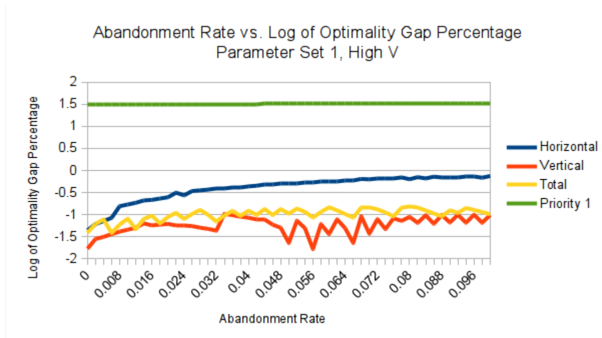
For the low value of V, all three heuristic policies are feasible, and stray at most 0.0374% from V_{low} over the entire range of abandonment rates tested. In contrast, the policy that prioritizes class 2 patients (P_2) has a feasibility gap of at least 16.05% over the range of abandonment rates. For V_{med}, we see that all three heuristic policies come within 0.0357% of binding, while P_2 violates the constraint by at least 10.16% over the range of β_2. Finally, for V_{high}, all heuristic policies are feasible and within 0.0339% within binding. The priority policy P_2 violates the constraint by at least 4.83%. For each policy, the optimality gap percentage of each policy is calculated. The abandonment rate is then plotted against the base-10 log of the optimality gap percentage. For V_{low}, all three of the heuristic policies significantly outperform P_1. The optimality gap of the total heuristic



(a) Low bound



(b) Medium bound



(c) High bound

Figure 2: Parameter set 1: log optimality gap

policy appears to be the smallest in this case, falling in the range $[0.002\%, 0.080\%]$ over the range of abandonment rates. This is followed by the vertical class ($[0.031\%, 0.184\%]$), and finally the horizontal class ($[0.017\%, 0.394\%]$). The optimality gap of P_1 falls within the range $[8.595\%, 9.097\%]$. For V_{high}, the ordering of the heuristic policies in terms of objective performance again remains the same. The optimality gap percentage for the total class is within the range $[0.039, 0.154]$. The ranges for the optimality gap percentages are $[0.017, 0.104,]$ for the vertical policy, $[0.044, 0.749]$ for the horizontal policy, and $[31.136, 32.419]$ for P_1.

6.2 Parameter set 2

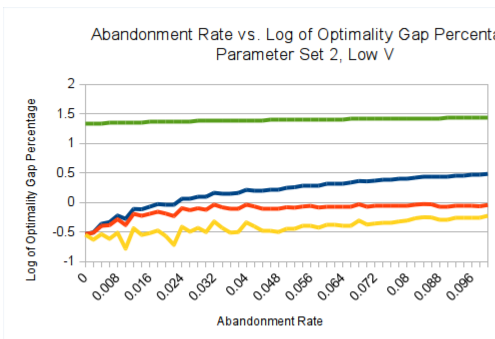
The second parameter set has $\lambda_1 = 0.4, \lambda_2 = 0.5, \mu_1 = 1, \mu_2 = 2$; the load is heavier for each class, and class 2 has a heavier load than class 1. The feasibility gaps are provided in Table 2.

For V_{low}, all of the heuristic policies are feasible and come within 0.0123% of binding. The feasibility gap for P_2 is at least 53.74%. For V_{med}, we see that all heuristic policies

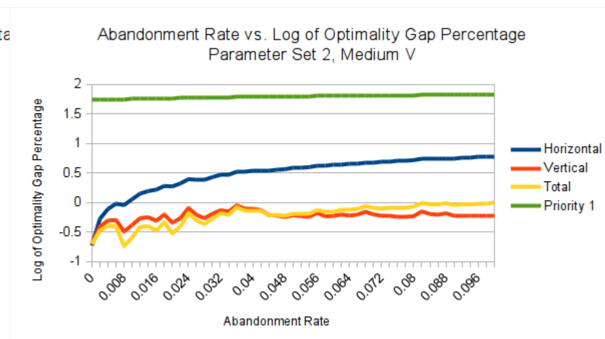
V	Min Heuristic Feas. Gap	Min Priority 2 Feas. Gap	Max Priority 2 Feas. Gap
0.8121	-0.0123 % (v)	53.74 %	64.17 %
0.9576	-0.0103 % (h)	30.38 %	39.23 %
1.1030	-0.0088 % (t)	13.19 %	20.87 %

Table 2: Parameter set 2; with higher workloads

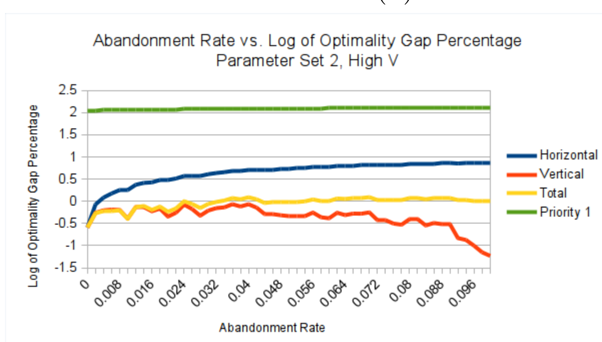
come within 0.0103% of binding, compared to P_2, which has a feasibility gap of at least 30.38% over the range of abandonment rates. Lastly, for V_{high}, all of the heuristic policies come within 0.0088% of binding, where as P_2 is always at least 13.19% from feasible. In terms of objective performance for V_{low}, the same ordering of total (optimality gap



(a) Low bound



(b) Medium bound



(c) High bound

Figure 3: Parameter set 2: log optimality gap

percentage in the range $[0.165, 0.592]$), vertical ($[0.290, 0.934]$), horizontal ($[0.290, 3.033]$), and P_1 ($[21.491, 27.139]$) is maintained, remaining unchanged from the first parameter set. The optimality gap percentages in this case seem to be much larger than for the first parameter set, which could be a consequence of the heavier traffic. The performance

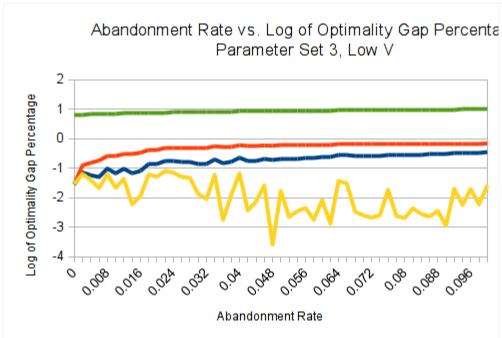
V	Min Heuristic Feas. Gap	Min Priority 2 Feas. Gap	Max Priority 2 Feas. Gap
0.4743	-0.0210 % (t)	144.89 %	321.63 %
0.6987	-0.0142 % (h)	64.22 %	186.25 %
0.9230	-0.0107 % (h)	24.31 %	116.68 %

Table 3: Feasibility gaps for parameter set 3

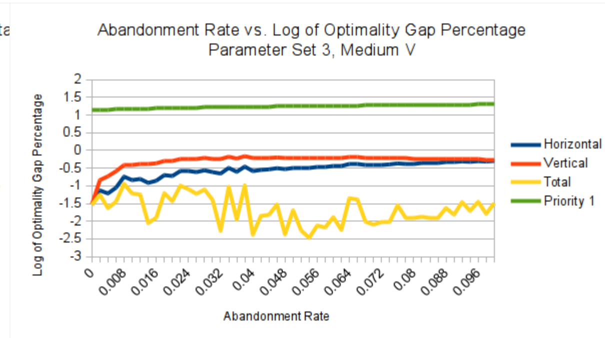
ordering for V_{med} changes slightly from the ordering for V_{low}, in that the vertical policy now performs better than the total policy for higher abandonment rates. The horizontal policy and P_1 still perform worse than both. The optimality gap percentage range is $[0.181, 0.992]$ for the total policy, $[0.195, 0.888]$ for the vertical policy, $[0.186, 6.005]$ for the horizontal policy, and $[53.919, 66.959]$ for P_1. For V_{high}, the optimality gap percentage of the total heuristic policy falls in the range $[0.256, 1.236]$. The range is $[0.059, 0.857]$ for the vertical heuristic policy, $[0.257, 7.275]$ for the horizontal policy, and $[110.449, 127.700]$ for P_1.

6.3 Parameter set 3

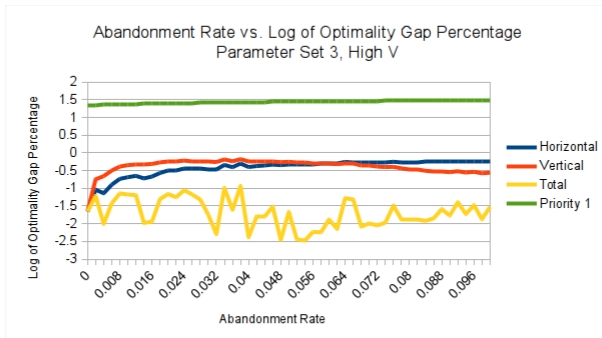
The third parameter set has $\lambda_1 = 0.4, \lambda_2 = 0.5, \mu_1 = 2, \mu_2 = 1$. The feasibility gaps are provided in Table 3. The heuristic policies are all within 0.0210% of V_{low} and feasible, compared to P_2 which is 141.89% from feasible. As the constraint level is relaxed to V_{med}, we observe that all three heuristic policies are within 0.0142% of binding, and P_2 is 64.22% from feasibility. Lastly, for V_{high}, we see that all three heuristic policies are feasible and within 0.0107% of binding, whereas P_2 has a feasibility gap percentage of at least 24.31% over the range of tested abandonment rates. For V_{low}, the total heuristic policy dominates the other two heuristic policies, with an optimality gap (%) falling in the range $[0.000, 0.080]$ over the tested abandonment rates. In contrast to the previous parameter set, the horizontal heuristic policy performs second best, with an optimality gap percentage falling in the range $[0.028, 0.351]$. The vertical policy did the worst of the heuristic policies (optimality gap percentage in the range $[0.029, 0.688]$), but still offering a steep increase in performance over P_1 ($[6.391, 9.938]$). For V_{med}, the ordering of the heuristic policies in terms of performance remains the same, with the total policy performing the best (optimality gap percentage in the range $[0.003, 0.113]$), followed by the horizontal policy ($[0.030, 0.506]$), the vertical policy ($[0.029, 0.690]$), and finally P_1 ($[13.622, 20.139]$). For V_{high}, the ordering of the heuristic policies in terms of objective performance seems to remain unchanged for smaller ($\beta_2 \leq 0.05$) values



(a) Low bound



(b) Medium bound



(c) High bound

Figure 4: Parameter set 3: log optimality gap

of the abandonment rate, at which point the vertical policy sees steady improvement as β_2 increases, eventually outperforming the horizontal policy. The optimality gap percentage over the range of abandonment rates falls in the range $[0.003, 0.116]$ for the total policy, $[0.022, 0.586]$ for the horizontal policy, $[0.023, 0.669]$ for the vertical policy, and $[21.899, 30.766]$ for P_1.

6.4 Parameter Set 4

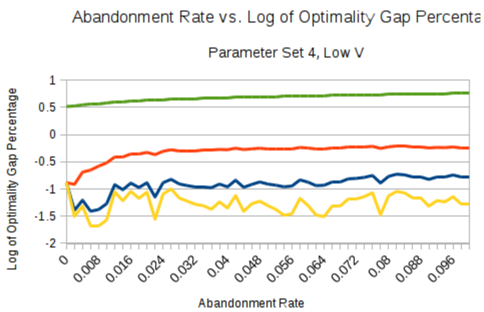
The fourth parameter set had $\lambda_1 = 0.1, \lambda_2 = 0.7, \mu_1 = \mu_2 = 1$. Here we adjusted the arrival rates to more accurately reflect the arrival rates of higher and lower acuity patients in our motivating ED example. The feasibility gaps are provided in Table 4.

6.5 Parameter Set 5

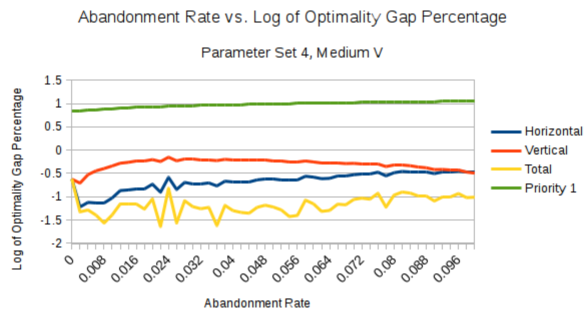
The fifth parameter set has $\lambda_1 = 0.1, \lambda_2 = 0.7, \mu_1 = 1, \mu_2 = 2$. Here we tried to mimic the dynamics of the ED setting even more closely by having the same arrival rates as in

V	Min Heuristic Feas. Gap	Min Priority 2 Feas. Gap	Max Priority 2 Feas. Gap
0.2299	-0.0435 % (h)	155.03 %	624.84 %
0.3488	-0.0284 % (v)	68.14 %	377.89 %
0.4676	-0.0213 % (h)	25.41 %	256.44 %

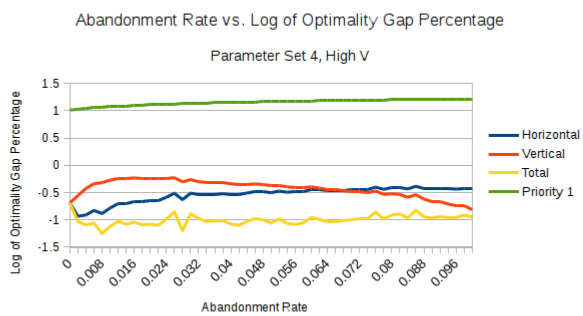
Table 4: Feasibility gaps for the ED example.



(a) Low bound



(b) Medium bound



(c) High bound

Figure 5: Parameter set 4: log optimality gap

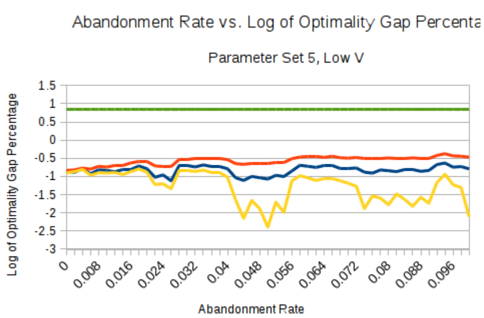
parameter set 4 and by making the service requirement lower for lower acuity patients. The feasibility gaps are provided in Table 5.

6.6 The N-Server Model

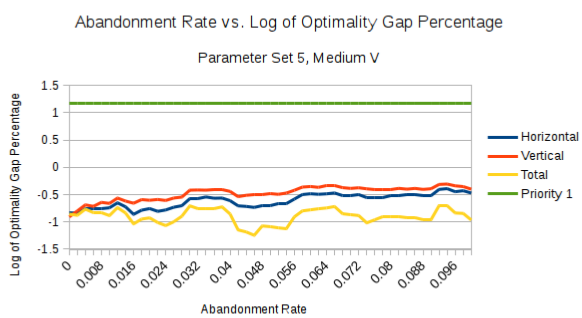
Assume now that there is a pool of N servers, where each customer can only be worked on by at most one server. Recall we consider only non-idling policies so that our action

V	Min Heuristic Feas. Gap	Min Priority 2 Feas. Gap	Max Priority 2 Feas. Gap
0.1362	-0.0734 % (v)	55.34 %	69.38 %
0.1614	-0.0617 % (v)	31.15 %	43.00 %
0.1865	-0.0534 % (t)	13.48 %	23.73 %

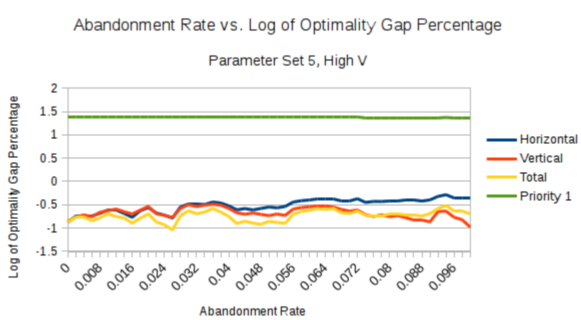
Table 5: Feasibility gaps for ED example with lower service level requirement.



(a) Low bound



(b) Medium bound



(c) High bound

Figure 6: Parameter set 5: log optimality gap

space becomes:

$$\mathbb{A}(i,j) := \begin{cases} \{(i,j)\} & i+j < N, \\ \{(n_1,n_2) \in \mathbb{Z}_+^2 : n_1+n_2 = N\} & i+j \geq N, \\ \{(0, \min(j,N))\} & i=0, j>0, \\ \{(\min(i,N),0)\} & i>0, j=0, \\ \{-1\} & i=j=0, \end{cases}$$

where -1 is a dummy action that denotes idling. The action space is interpreted as the number of servers to allocate to each class of customers. We analyze the performance of a class of heuristic policies numerically. As in Section 6, we construct near-binding policies within each of the horizontal, vertical, and total classes of heuristic policies. However, these classes of policies are defined differently to account for the multi-server setting: for a given heuristic policy, if the class to be served has $q < N$ customers, then q servers are assigned to that class and the rest are assigned to the other class. For example, if we have $N = 5$ servers and the policy (as constructed in the single-server case) tells us to serve class 2 in state $(1,4)$, then in the multi-server case the analogous policy would send 1 server to class 2 and 4 servers to class 1. If the other class, however, has less than $N-q$ customers, then as many as possible are sent to that class, and the rest idle. Continuing our 5-server example, if the single-server policy says to serve class 2 in state $(1,1)$, then the analogous multi-server policy will allocate 1 server apiece to each class, and have the remaining 3 servers idle. Note that if we were to perform the same style of numerical analysis performed in Section 6, we would need to choose a truncation level. Since the relationship between the truncation level and number of servers on the accuracy of our cost approximation is unclear, we will instead evaluate policies via simulation. This comes at the expense of not having an optimal value to compare to when judging the performance of the heuristic policies. We attempt to address this issue by comparing the objective values of heuristic policies against those of the priority policies. The rest of the methodology is as follows. For each parameter set, three values of V (low, medium, and high) were obtained using the same method as used in Section 6. Abandonment rates (β_2) ranging from 0 to 0.1 in increments of 0.01 are tested for each V. For every combination of V and β_2, a near-binding policy is obtained within each heuristic class. All simulation costs reported are average costs over 100 steady-state simulation runs. In order to make the difference in objective values between the heuristic policies and the priority policies more visually obvious, we plot the base-10 log of the class 2 costs.

6.7 Parameter Set 1

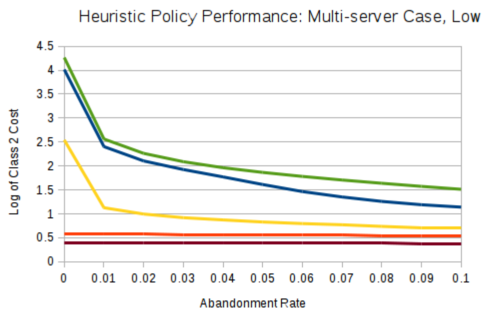
The first parameter set tested has $\lambda_1 = 2.5, \lambda_2 = 5, \mu_1 = 1, \mu_2 = 2, N = 8$ and serves as a baseline case. The feasibility gaps are given below in Table 6. Again note that since all

V	Min Heuristic Feas. Gap	Min Priority 2 Feas. Gap	Max Priority 2 Feas. Gap
2178.643	-3.603 % (h)	299.655 %	319.350 %
4354.780	-1.889 % (h)	99.942 %	109.796 %
6530.917	-1.438 % (h)	33.321 %	39.891 %

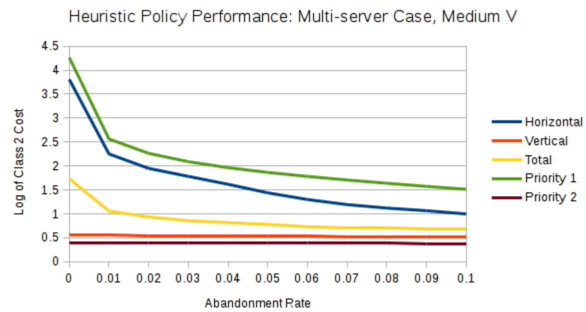
Table 6: Feasibility gaps for multi-server case; parameter set 1.

heuristic policies constructed are feasible, each of them have a negative feasibility gap. Hence, the minimum feasibility gap listed in the table represents the largest deviation from V. Similarly, P_2 will be infeasible for each V, and will thus have positive feasibility gaps. As in the single-server case, P_2 is not close to feasible, violating the constraint by at least 33% in the best case.

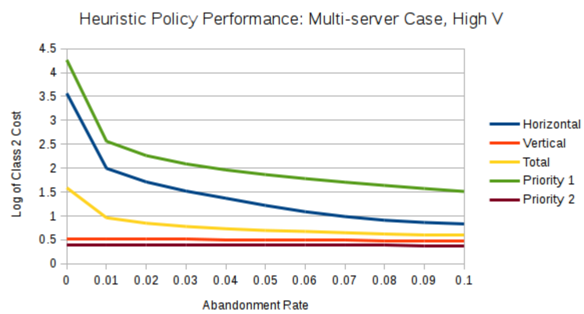
As the policy cost evaluations were obtained via simulation, we cannot obtain optimality gaps for the heuristic policies. Instead, we evaluate their performance as compared with the absolute best and worst case policies: P_1 (worst) and P_2 (best). As in Section 6, we take the base-10 log of the expected long-run average class 2 costs of each policy for visual clarity. It should first be noted that the "least binding" of all of the heuristic policies are those taken from the horizontal class. Not surprisingly, these are the heuristic policies that perform the worst. Another observation to make is the separation in the performance of the three heuristic (classes of) policies, at least initially when the abandonment rate is small. One reasonable conjecture is that this could be due to a light offered load: $\rho_1 := \frac{\lambda_1}{N\mu_1} = 0.3125 = \frac{\lambda_2}{N\mu_2} =: \rho_2$. Intuitively, if the offered load is very high, then the time spent in states where there is at least one class with less than N customers in system, then the policy behaves the same (probabilistically) as in the single-server system. Looking at the graphs for the single-server case in Section 6, there is a relatively small performance gap between the three classes of heuristic policies. The graphs of performance across the low, medium, and high values of V look similar, with the only difference being a slight shift downward for all of the heuristic policies.



(a) Low bound



(b) Medium bound



(c) High bound

Figure 7: Parameter set 1, multi-server: log optimality gap

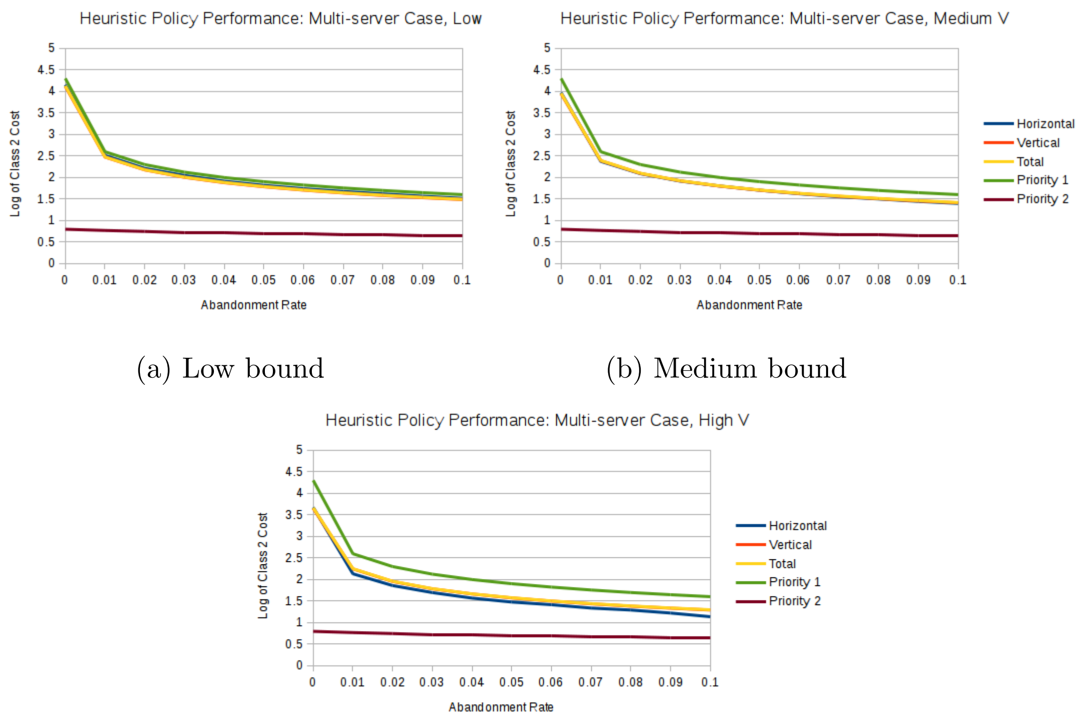
6.8 Parameter Set 2

The second parameter set tested had $\lambda_1 = 2, \lambda_2 = 4, \mu_1 = 0.45, \mu_2 = 1, N = 5$. Here the parameters are tweaked to increase the total offered load. The offered load of each class remains about the same: $\rho_1 = 0.889$ and $\rho_2 = 0.800$. The feasibility gaps are provided in Table 7. Again we see that the horizontal class of policies is the loosest in terms of

V	Min Heuristic Feas. Gap	Min Priority 2 Feas. Gap	Max Priority 2 Feas. Gap
2442.501	-1.350 % (h)	298.694 %	303.064 %
4874.366	-4.710 % (h)	99.782 %	101.972 %
7306.231	-3.188 % (h)	33.285 %	34.746 %

Table 7: Feasibility gaps for multi-server case; parameter set 2.

being binding, for all three values of V. P_2 is still far from being feasible: the closest it gets to feasibility over all tested values of V and β_2 is within 33.285%. One immediate observation from the performance plots is the large gap (relative to the first parameter



(a) Low bound

(b) Medium bound

(c) High bound

Figure 8: Parameter set 2, multi-server: log optimality gap

set) between the class 2 cost of the heuristic policies and P_2. In addition, the class 2 costs of the heuristic policies are close together (notice the curve for the vertical class is obscured in all three plots), which is consistent with our conjecture that when the total offered load is sufficiently high, the system behaves similarly to the single-server system with the analogous policy. As in the previous parameter set, the plots over all three values of V look similar, with the only difference being a downward shift of the three heuristic classes of policies. Unlike in the previous parameter set, we see a slight separation between the performance of the horizontal class of policies from the other two heuristic classes.

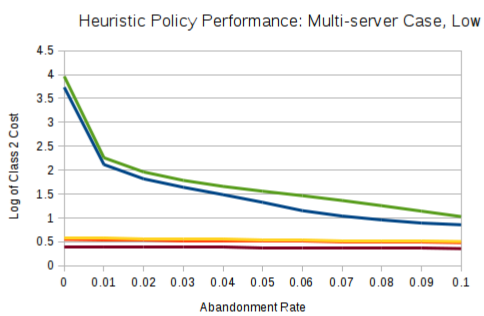
6.9 Parameter Set 3

The third parameter set tested is $\lambda_1 = 5, \lambda_2 = 2.5, \mu_1 = 2, \mu_2 = 1, N = 8$. Note that this matches the first parameter set, except the indices are interchanged. The reason for this is to see if the vertical policy still remains the best within the three heuristic classes tested. The feasibility gaps are provided in Table 8. Again we see a large feasibility gap for P_2 and heuristic policies that are all very close to binding. It should be noted that

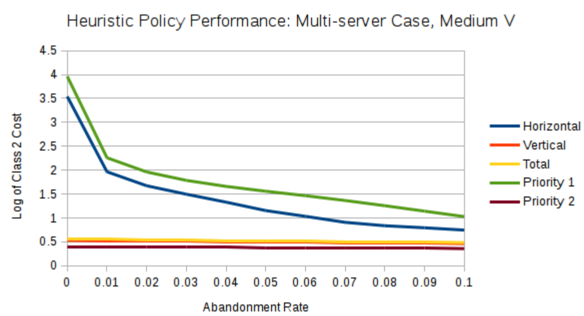
V	Min Heuristic Feas. Gap	Min Priority 2 Feas. Gap	Max Priority 2 Feas. Gap
4169.078	-3.281 % (h)	299.820 %	340.951 %
8335.651	- 2.010 % (h)	99.970 %	120.542 %
12502.224	- 1.412 % (h)	33.327 %	47.042 %

Table 8: Feasibility gaps for multi-server case; parameter set 3.

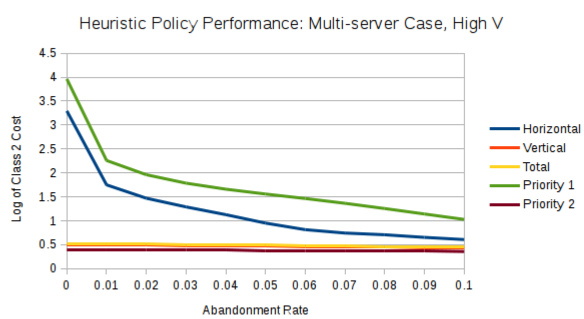
the vertical and total heuristic policies came closer to binding than did the horizontal class: they are binding up to three decimal places. This is somewhat expected, as the class 1 cost should be less sensitive to policy changes within the horizontal class that places decisions based on the number of class 2 customers in the system. Naturally, since these policies are further from binding, we also see that they perform significantly worse than the other two heuristic policy classes, as seen in the graphs below. As with the



(a) Low bound



(b) Medium bound



(c) High bound

Figure 9: Parameter set 3, multi-server: log optimality gap

first parameter set, we see that both the vertical and total classes produce policies that perform vastly better than those of the horizontal class. In addition, the horizontal class produces policies that perform almost as poorly as P_1 for small values of the abandonment rate, especially when V is high. We also see that the total and vertical classes produce policies that perform similarly across all abandonment rates tested. This differs from the results from the first parameter set, where the vertical class provides significantly better results than the total class for small values of the abandonment rate. Again we see that under a small offered load (the same as in the first parameter set), the performance gaps between the heuristic policies is large.

7 Conclusion/Future Work

This work is motivated in part by the patient flows question in emergency departments. Some of the last three authors' earlier work makes steps toward providing insights into the improvement of patient flows when it comes to lower acuity patients. In this case, we consider when there are either both low and high acuity patients or even just high acuity patients with various levels of injury. We model this system as a two-class queueing system, with the objective of minimizing the expected long-run average holding cost of class 2 customers while keeping that of class 1 customers under a given level, V. This is the most natural way to describe the issue to medical service providers and happens to also be a technically challenging problem.

Applying a theorem from [1], we used the equivalent Lagrangian dual problem to develop sufficient optimality conditions. In the case where class 2 patients have infinite patience, we leveraged the well-known $c\mu$-rule, simplifying these conditions to finding a stationary policy that satisfies the constraint at equality. We then proved weak continuity results for the expected long-run average holding costs for both classes. Motivated by these results, we constructed classes of policies with nice structural properties. This fits the implicit constraint that the policies obtained need to be easily implementable in a medical setting.

We then prove that for each such sequence of sets, the corresponding class of policies contains an optimal policy for the constrained problem in the case of no class 2 abandonments. The construction of these policies motivated a method of constructing simple threshold policies that are in turn tested as heuristics for the problem with abandonments. In both the case of a single or multiple servers our heuristics seem to provide significant improvements over priority policies when considering both the feasibility and the minimal cost.

While the results outlined above are promising, there are still some aspects of this problem to explore. It is desirable to prove structural results for the case with class

2 abandonments. However, literature suggests (see e.g. [13]) that solving this problem directly may be difficult. A more promising direction may be to consider asymptotic optimality of highly-structured policies, following the approach of [7]. Such a fluid approach may also be useful in approximating the parameters of optimal randomized-threshold policies.

In addition to considering the fluid model approach, we could attempt to solve our problem under a more general setting. Features which could make this problem both more interesting and, in certain cases (such as the ED setting), more realistic are the addition of multiple customer classes, each associated with an additional constraint. It would be of particular interest to develop a set of optimality conditions and heuristics which generalizes those developed in this paper. Adding multiple servers with the ability to cooperate (in a non-linear fashion) would more accurately describe the ED setting. Lastly, it would be interesting to develop theoretical guarantees for heuristic policies, specifically for those tested in our numerical experiments.

Acknowledgements

The authors would like to thank Douglas Down from McMaster University for suggesting we consider threshold policies in the early stages of our problem formulation.

A Appendix

A.1 Verification of Assumptions

We verify assumptions (A1)-(A4), that are needed to apply **Theorem 12.7** from [1] and hence establish the equivalence between the constrained problem and its Lagrangian dual. The assumptions are stated as follows. Let $\mathcal{K} := \{(x,a) : x \in \mathbb{X}, a \in A(x)\}$ and let $\mathbb{K}$ denote its Borel σ-algebra generated by the rectangles $\{(x,A) : x \in \mathbb{X}, A \subseteq A(x)\}$. We need to show:

(A1) For any state $x \in \mathbb{X}$, if $\{a_n\} \subseteq A(x)$ is a sequence of actions with $a_n \to a \in A(x)$ as $n \to \infty$, then for every $y \in \mathbb{X}$, $\lim_{n\to\infty} P(y|x,a_n) = P(y|x,a)$.

(A2) Under any stationary policy, the induced MC contains a single ergodic class, and absorption into the positive recurrent class takes place in a finite expected time. This corresponds to assumption (B1) in Chapter 11 of [1].

(A3) The immediate cost functions c_1 and c_2 are bounded below.

(A4) There exists an increasing sequence of compact sets $\{K_n\}_{n\in\mathbb{N}} \subseteq \mathcal{K}$ with $\bigcup_{n\in\mathbb{N}} K_n = \mathcal{K}$ and $\liminf_{n\to\infty}\{r(k) : k \notin K_n\} = \infty$ for $r = c_1, c_2$.

To see that (A1) holds, note that in state $(i,j) > 0$, we only need to check that for $a_n \to a$, we have $\mathbb{P}((i-1,j)|(i,j),a_n) \to \mathbb{P}((i-1,j)|(i,j),a)$ and $\mathbb{P}((i,j-1)|(i,j),a_n) \to \mathbb{P}((i,j-1)|(i,j),a)$. This is seen by noting that $a_n\mu_1 \to a\mu_1$ and $(1-a_n)\mu_2 \to (1-a)\mu_2$. It can be easily checked that every stationary policy yields an irreducible, aperiodic MC. Under the traffic condition $\rho < 1$, every such MC is also positive-recurrent, and so assumption (A2) holds. To see why, first note that under any stationary policy, the total number of customers in system increases by one with rate $\lambda_1 + \lambda_2$, and decreases by one at a rate bounded below by $\min(\mu_1, \mu_2)$. Finite total cost follows since the birth-death process with birth rate $\lambda_1 + \lambda_2$ and death rate $\min(\mu_1, \mu_2)$ is positive-recurrent under our traffic condition. (A3) holds trivially since $c_1, c_2 \geq 0$. To see that (A4) holds, take $K_n = \{((i,j),0) : i,j \leq n\}$ so that for every n,

$$\inf\{r(k) : k \notin K_n\} = n + 1$$

for $r = c_1, c_2$, which diverges as $n \to \infty$.

A.2 Notation and Definitions

Formally, a policy $\sigma \in \Pi^S$ is defined as a set of measures $\{\sigma_x : x \in \mathbb{X}\}$. For $A \subseteq A(x)$, $\sigma_x(A) \in [0,1]$ is the probability that an action $a \in A$ is taken in state x under policy σ. In some settings, it is easier to work with the "decision" made by the policy in state x, given by $a_\sigma(x) \in [0,1]$. Here $a_\sigma(x)$ can be interpreted as the probability of serving class 1 customers in state x under policy σ. For a sequence of stationary policies $\{\sigma_n, n \geq 0\}$, we say that $\{\sigma_n, n \geq 0\}$ **converges pointwise** to σ (for some $\sigma \in \Pi^S$) if $a_{\sigma_n}(x) \to a_\sigma(x)$ for all $x \in \mathbb{X}$. If, for all bounded and continuous functions $g : A(x) \mapsto \mathbb{R}$, we have

$$\lim_{n\to\infty} \int_{A(x)} g(z)d(\sigma_n)_x(z) = \int_{A(x)} g(z)d\sigma_x(z)$$

for every $x \in \mathbb{X}$, we say that $\{\sigma_n\}$ **converges weakly** to σ. Under the assumption (A2), each stationary policy ν induces an **occupation measure** f^ν satisfying

$$f^\nu(x, A) = \nu_x(A)\pi^\nu(x), \ \ x \in \mathbb{X}, A \subseteq A(x),$$

where π^ν is the stationary distribution of the process induced by policy ν. Intuitively, the occupation measure of a particular policy σ describes the long-run average fraction of time the induced process spends in each state-action pair. An important property of a set of occupation measures $\{f^\nu\}_{\nu\in I}$ over some index set I is tightness. In our context, $\{f^\nu\}_{\nu\in\Pi^S}$ being tight means that for any desired probability level, p, we can find a set of state-action pairs K such that the long-run fraction of time spent in K is at least p under any stationary policy. A more formal definition is provided below.

Definition A.1 *Let $\mathcal{K} := \mathbb{X} \times A$ and let $\mathbb{K}$ denote its Borel σ-algebra. We say the collection of occupation measures $\{f^\nu\}_{\nu\in\Pi^S}$ over the set of stationary policies is **tight** if, for every $\epsilon > 0$, there exists $K_\epsilon \in \mathbb{K}$ such that*

$$f^\nu(K_\epsilon) > 1 - \epsilon$$

for every $\nu \in \Pi^S$.

A.3 Pointwise Continuity of Costs

Given these preliminaries, we can now state the following theorem from [1].

Theorem A.2 (adapted from Theorem 11.2 (i) in [1])
Let $\Pi \subseteq \Pi^S$. If $\{f^\nu\}_{\nu\in\Pi}$ is tight, then f is weakly continuous over Π. That is, for any sequence of polices $\{\nu_n\}_{n=0}^\infty \subseteq \Pi$ converging weakly to $\nu \in \Pi$, we have that $\{f^{\nu_n}\}$ converges to f^ν weakly.

Verifying the assumptions of the theorem involves two steps: showing that pointwise convergence of policies implies weak convergence, and showing that the set of occupation measures over the class of stationary policies is tight. We verify both of these with the following lemmas.

Lemma A.3 *Let σ be a stationary policy, and let $\{\sigma_n\}_{n=0}^\infty$ be a sequence of stationary policies. Suppose that $\sigma_n \to \sigma$ pointwise. Then $\sigma_n \to \sigma$ weakly.*

Proof. Note that since $\sigma_n \to \sigma$ pointwise, we have that $a_{\sigma_n}(x) \to a_\sigma(x)$ for all $x \in \mathbb{X}$ as $n \to \infty$. Recall that under any stationary policy ν, we let $a_\nu(x) \in A(x)$ denote the action ν chooses in state x. That is to say, that by the definition of the action set $A(x)$ randomized policies are obtained by choosing the action $a \in A(x)$ in the set $(0, 1)$ which has the interpretation of choosing to work at station 1 with probability a. Consequently, for each n the action chosen in x is done with probability 1. That is, $(\sigma_n)_x(a_n) = 1$ and $\sigma_x(a) = 1$. Hence for all bounded and continuous functions $g : A(x) \to \mathbb{R}$, we have

$$\lim_{n\to\infty} \int_{A(x)} g(z)d(\sigma_n)_x(z) = \lim_{n\to\infty} g(a_{\sigma_n}(x)) = g(a_\sigma) = \int_{A(x)} g(z)d\sigma_x(z)$$

as $n \to \infty$, proving the result. ■

Lemma A.4 *The collection of occupation measures $\{f^\nu\}_{\nu\in\Pi^S}$ is tight.*

Proof. Note that under our assumptions, every stationary policy yields a stable Markov process with finite expected long-run average number of customers in system. Hence for $k = 1, 2$, we have that $\sup_{\nu \in \Pi^S} C_k(\nu) =: \hat{C}_k < \infty$. Define for $N \in \mathbb{N}$ the compact set $S_N := \{(i, j, A(i, j)) : i, j \le N\}$. If $\{f^\nu\}_{\nu \in \Pi^S}$ is not tight, then by definition, there exists $\tilde{\epsilon} > 0$ such that for every $N \in \mathbb{N}$, there is a policy $\nu_N \in \Pi^S$ with

$$1 - \tilde{\epsilon} \ge f^{\nu_N}(S_N) = \sum_{(i,j): i,j \le N} \pi^{\nu_N}(i, j),$$

and thus

$$\sum_{(i,j): i,j > N} \pi^{\nu_N}(i, j) \ge \tilde{\epsilon}.$$

Hence, picking $\hat{N}$ such that $\tilde{\epsilon}\hat{N} > \hat{C}_1$, we have that

$$C_1(\nu_{\hat{N}}) \ge \sum_{(i,j): i,j > \hat{N}} i\pi^{\nu_{\hat{N}}}(i, j) > \hat{N} \sum_{(i,j): i,j > \hat{N}} \pi^{\nu_{\hat{N}}}(i, j) \ge \tilde{\epsilon}\hat{N} > \hat{C}_1,$$

a contradiction. Thus $\{f^\nu\}_{\nu \in \Pi^S}$ is tight, completing the proof. ∎

Proof of Lemma 3.5:

Proof. We know for any bounded and continuous $h : \mathcal{K} \mapsto \mathbb{R}$, **Theorem** A.2 yields

$$\lim_{n \to \infty} \int_{\mathcal{K}} h(z) df^{\sigma_n}(z) = \int_{\mathcal{K}} h(z) df^{\sigma}(z).$$

Noting that $\mathbb{1}_x(y, a)$ is bounded and continuous on $\mathcal{K}$, applying the theorem yields that, for any $x \in \mathbb{X}$,

$$\lim_{n \to \infty} \pi^{\sigma_n}(x) = \lim_{n \to \infty} \int_{\mathcal{K}} \mathbb{1}_x(z) df^{\sigma_n}(z) = \int_{\mathcal{K}} \mathbb{1}_x(z) df^{\sigma}(z) = \pi^{\sigma}(x),$$

as desired. ∎

A.4 Proof of Pointwise Continuity

Proof of Theorem 3.4. Again, we prove the case $k = 1$, as the other case is proved in the same manner. Pick any $\epsilon > 0$. Let $\Delta_N := \{(i, j) \in \mathbb{X} : i + j \le N\}$. Lemma 3.6 allows us to pick $\hat{N}$ such that

$$|C_1(\sigma_n) - \sum_{(i,j) \in \Delta_{\hat{N}}} i\pi^{\sigma_n}(i, j)| < \frac{\epsilon}{3} \qquad \forall n$$

$$|C_1(\sigma) - \sum_{(i,j) \in \Delta_{\hat{N}}} i\pi^{\sigma}(i, j)| < \frac{\epsilon}{3}.$$

Since $\pi^{\sigma_n} \to \pi^{\sigma}$ pointwise, for all sufficiently large n, we have

$$|\pi^{\sigma_n}(i,j) - \pi^{\sigma}(i,j)| < \frac{\epsilon}{3|\Delta_{\hat{N}}|\hat{N}}$$

for all $(i,j) \in \Delta_{\hat{N}}$. Hence

$$\begin{aligned}
|C_1(\sigma_n) - C_1(\sigma)| &\le |C_1(\sigma_n) - \sum_{(i,j)\in\Delta_N} i\pi^{\sigma_n}(i,j)| + |\sum_{(i,j)\in\Delta_N} i\pi^{\sigma_n}(i,j) - \sum_{(i,j)\in\Delta_N} i\pi^{\sigma}(i,j)| \\
&\quad + |\sum_{(i,j)\in\Delta_N} i\pi^{\sigma}(i,j) - C_1(\sigma)| \\
&< \frac{\epsilon}{3} + \sum_{(i,j)\in\Delta_{\hat{N}}} i|\pi^{\sigma_n}(i,j) - \pi^{\sigma}(i,j)| + \frac{\epsilon}{3} \\
&\le \frac{2\epsilon}{3} + \hat{N} \sum_{(i,j)\in\Delta_{\hat{N}}} \frac{\epsilon}{3|\Delta_{\hat{N}}|\hat{N}} = \epsilon,
\end{aligned}$$

as desired. ∎

References

[1] E. Altman. *Constrained Markov Decision Processes*. Chapman and Hall/CRC, 1999. 2, 4, 6, 8, 11, 14, 29, 30, 32

[2] A. Arapostathis, A. Biswas, and G. Pang. Ergodic control of multi-class m/m/n+m queues in the halfin-whitt regime. *Ann. Appl. Probab.*, 25:3511–3570, 2015. 3

[3] N. T. Argon, S. Ziya, and R. Righter. Scheduling impatient jobs in a clearing system with insights on patient triage in mass casualty incidents. *Probability in The Engineering and Informational Sciences*, 22:301–332, 2008. 3

[4] M. Armony and C. Maglaras. Contact centers with a call-back option and real-time delay information. *Operations Research*, 52:527–545, 2004. 3

[5] M. Armony and C. Maglaras. On customer contact centers with a call-back option: Customer decisions, routing rules and system design. *Operations Research*, 52:271–292, 2004. 3

[6] M. Armony, N. Shimkin, and W. Whitt. The impact of delay announcements in many-server queues with abandonment. *Operations Research*, 57:66–81, 2009. 3

[7] R. Atar, G. Chanit, and N. Shimkin. The $c\mu/\theta$ rule for many-server queues with abandonment. *Operations Research*, Sept-Oct, 2010. 3, 30

[8] U. Ayesta, P. Jacko, and V. Novak. A nearly-optimal index rule for scheduling of users with abandonment. *Proceedings of IEEE INFOCOM*, 2011. 3

[9] O. Berman, J. Wang, and K. P. Sapna. Optimal management of cross-trained workers in services with negligible switching costs. *Eur. J. Oper. Res.*, 167:349–369, 2005. 3

[10] S. Bhulai and G. Koole. A queueing model for call blending in call centers. *IEEE Transactions on Automatic Control*, 48:1434–1438, 2000. 3

[11] C. Buyukkoc, P. Varaiya, and J. Walrand. The $c\mu$-rule revisited. *Advances in Applied Probability*, 17(1):237–238, 1985. 1, 2, 14

[12] J. Dai and S. He. Many-server queues iwth customer abandonment: A survey of diffusion and fluid approximations. *Journal of Systems Science and Systems Engineering*, 21:1–36, 2012. 3

[13] D. G. Down, G. Koole, and M. E. Lewis. Dynamic control of a single server system with abandonments. *Queueing Systems: Theory and Applications*, 67, January 2011. 2, 15, 30

[14] E. Feinberg. Constrained semi-markov decision processes with average rewards. *Mathematical Methods of Operations Research*, 39:257–288, 1994. 1, 14

[15] N. Gans and Y.-P. Zhou. A call-routing problem with service-level constraints. *Operations Research*, 51:255–271, 2003. 3

[16] S. Ghamami and A. Ward. Dynamic scheduling of an n-system with reneging. *Preprint*, 2010. 3

[17] S. Ghamami and A. Ward. Dynamic scheduling of a two-server parallel server sysetm with complete resource pooling and renegin in heavy traffic: asymptotic optimality of a two-threshold policy. *Mathematics of Operations Research*, 38:761–824, 2013. 3

[18] J. Harrison and A. Zeevi. Dynamic scheduling of a multiclass queue in the halfin and whitt heavy traffic regime. *Operations Research*, 52:243–257, 2004. 3

[19] S. Lippman. Applying a new device in the optimization of exponential queueing systems. *Operations Research*, 23:687–710, 1975. 2

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

[21] M. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., 1994. 9

[22] L. Sennott. Average cost optimal stationary policies in infinite state markov decision processes with unbounded costs. *Operations Research*, 37:626–633, 1989. 14

[23] R. Serfozo. An equivalence between continuous and discrete time markov decision processes. *Operations Research*, 27:616–620, 1978. 2

[24] T. Tezcan and J. Dai. Dynamic control of n-systems with many servers, asymptotic optimality of a static priority policy in heavy traffic. *Operations Research*, 58:94–110, 2010. 3

[25] A. Ward. Asymptotic analysis of queueing systems with reneging: A survey of results for fifo, single class models. *Oper. Res. Management Sci.*, 16:1–14, 2011. 3

[26] A. R. Ward and P. W. Glynn. A diffusion approximation for a markovian queue with reneging. *Queueing Systems*, 43:103–128, 2003. 3

[27] A. R. Ward and P. W. Glynn. A diffusion approximation for a $gi/gi/1$ queue with balking or reneging. *Queueing Systems*, 50:371–400, 2005. 3

[28] R. Yang, S. Bhulai, and R. Mei. Optimal resource allocation for multiqueue systems with a shared server pool. *Queueing Systems*, 68:133–163, 2011. 4