Two-Class Routing with Admission Control and Strict Priorities

Kenneth Chong, Shane G. Henderson, Mark E. Lewis

April 13, 2017

Abstract

We consider the problem of routing and admission control in a loss system featuring two classes of arriving jobs (high-priority and low-priority jobs) and two types of servers, in which decision-making for high-priority jobs is forced, and rewards influence the desirability of each of the four possible routing decisions. We seek a policy that maximizes expected long-run reward, under both the discounted reward and long-run average reward criteria, and formulate the problem as a Markov decision process. When the reward structure favors high-priority jobs, we demonstrate that there exists an optimal monotone switching curve policy with slope of at least -1. When the reward structure favors low-priority jobs, we demonstrate that the value function, in general, lacks structure, which complicates the search for structure in optimal policies. However, we identify conditions under which optimal policies can be characterized in greater detail. We also examine the performance of heuristic policies in a brief numerical study.

1 Introduction

A common decision faced by operators of service systems is the problem of dynamically allocating system resources to incoming demand. Complicating matters is the fact that in such systems, customers and servers are typically heterogeneous. Servers may have different capabilities, or receive training in varying sets of skills. Similarly, different types of customers may have varying needs, or take priority over other customer classes. This situation arises, for instance, in telecommunications [1, 6, 22], healthcare [4, 20], and rental systems [13, 23, 27]. Decision-making in such an environment has typically been modeled in the literature as a problem of admission control or routing in a queueing system, canonical examples of which include the models by Harrison [15] and Miller [21].

However, there may be restrictions in the decisions that can be made in certain system states. One example of this (the original motivation for the model we present in this paper) arises in Emergency Medical Service (EMS) systems. In this setting, arriving calls (customers) are categorized into priorities, based upon severity, and ambulances (servers) are staffed by personnel who receive varying levels of medical training. EMS providers are primarily evaluated by their responsiveness to calls of the highest priority: emergencies for which patients' lives are potentially at stake (such as cardiac

arrest). As a result, decision-making with these types of calls is typically forced, in that they receive an immediate response if resources are available. Moreover, if multiple ambulances can respond to such a high-priority call, EMS providers prefer to dispatch one staffed by paramedics (who receive the highest level of medical training), as doing so can have a measurable effect on patient outcomes. (See, for instance, Bakalos et al. [2] or Jacobs et al. [16] for discussions of these effects.) However, EMS providers are often required to provide a minimum level of service to lower-priority calls. As a result, decision-makers face a trade-off between keeping resources available to serve higher-priority calls, and allocating these resources to adequately serve lower-priority calls.

We study this situation through a loss system featuring two types of servers (Type A and Type B), that is tasked with processing two classes of jobs: Type H (or high-priority) jobs and Type L (or low-priority) jobs, in which decision-making with Type H jobs is forced, and Type L jobs are subject to admission control. Although this model cannot be directly used to guide decision making in practical contexts (as the locations of ambulances must also be taken into account), it is theoretically interesting in its own right. This is because the element of forced decision-making in our model presents significant technical challenges. In particular, standard techniques for characterizing the optimal admission control policy, which involve formulating our model as an MDP and analyzing the corresponding value function, are inadequate. Although there are conditions under which standard techniques suffice, in general, the value function associated with our model lacks classical structural properties, such as convexity or supermodularity. Moreover, this lack of structure can be directly attributed to our requirement that decision-making with Type H jobs is forced.

To address these difficulties, we instead identify situations in which we can provably recover structure in the optimal policy, and proceed in two directions. First, we provide a sufficient (but not necessary) condition on our model inputs under which our value function is convex and supermodular. Second, we demonstrate that when we restrict attention to a certain intuitive class of policies, the optimal policy in this class is "monotone" in a way that we later specify, the proof of which relies on a novel argument using renewal theory. One may wonder whether the technical challenges our model presents can be circumvented, for instance by devising an effective heuristic policy, but numerical experiments suggest that in service systems such as ours, that there is value in taking into account heterogeneity of both servers and jobs.

The remainder of this paper is organized as follows. After a review of relevant literature in Section 2, we explicitly formulate our model as a Markov decision process in Section 3, and present the corresponding optimality equations. We identify basic structural properties in our value function in Section 4, which we leverage in Section 5 to identify conditions under which the optimal policy can be characterized. Following a brief numerical study in Section 6, we conclude in Section 7.

2 Literature Review

Routing and admission control in queueing systems has been widely studied in the literature. Surveys, such as those by Stidham [29] and by Stidham and Weber [30], provide excellent overviews of work in this area; we restrict our attention to models closely related to our own— specifically, models featuring multiple classes of arriving jobs. Perhaps the most influential model of this type is due to Miller [21], who studies the problem of admission control to a loss system featuring homogeneous servers and n job classes, as well as class-dependent rewards. Extensions to Miller's model have been studied (see, for instance, Carrizosa et al. [8], Feinberg and Reiman [12], and Lewis et al. [18]), and his model has also been adapted to study telecommunications systems and call centers (examples of which include Altman et al. [1], Bhulai and Koole [6], Blanc et al. [7], Gans and Zhou [14], and Örmeci and van der Wal [22]). Our model differs from those previously mentioned in two respects. First, it features two types of servers, and we allow rewards to depend on the type of server to which jobs are assigned. Second, one of our arrival streams is uncontrolled, as Type H jobs must be admitted whenever possible. While Blanc et al. [7] also consider a model with a similar element of forced decision-making, and Bhulai and Koole [6] and Gans and Zhou [14] impose a service level constraint on Type H jobs, they do so in a system with homogeneous servers.

One system from this body of literature that bears a particularly close resemblance to ours is the N-network, which can be viewed as a variant of our model in which $R_{HA} = R_{HB}$, Type L jobs cannot be routed to Type A servers, and jobs can queue. See, for instance, Bell and Williams [3], Down and Lewis [11], and Harrison [15]. While our model does not feature a queue, we allow for servers to be flexible, in that they can serve both types of jobs, and discourage certain routing decisions through the reward structure we impose.

As previously mentioned, models of this type have also been used to study real-time decision making in EMS systems, for which the goal is to identify policies for dispatching ambulances to emergency calls, or to relocate idle ambulances to improve the system's responsiveness to future call arrivals. See, for instance, the models by Berman [4, 5], Jarvis [17], McLay and Mayorga [20], and Zhang [31]. These models are detailed, and take into account the effects of ambulance locations (and heterogeneity) when making decisions, and thus, can be used to guide decision-making in practical contexts. Although our model lacks this applicability, we consider a more nuanced objective function that incorporates the system's responsiveness to high-priority and low-priority calls, as well as the level of service that different types of ambulances can provide to these calls. It can also be used to draw basic insights; we use this model in [9] to study the effects of fleet composition (the mixture of "Type A" and "Type B" ambulances deployed) on the performance of EMS systems. Our choice to model the system as a loss system is partly due to tractability, and partly due to the fact that some

systems divert calls to an external service (such as the fire department) during periods of congestion.

Another application area includes capacity management in rental systems, a relatively small subfield of revenue management in which individual resources are not perishable (as is the case, for instance, with seats on a particular flight itinerary), but can be reused to generate revenue from multiple customers. In this setting, servers are viewed as resources that can be rented to customers for random (exponentially distributed) durations of time. When resources are scarce, there is a decision as to which resources (if any) to make available to arriving customers. This problem has been studied, for instance, by Gans and Savin [13], Savin et al. [27], and Papier and Thonemann [23], all of whom model resources as homogeneous. In this context, our model can be viewed as that of a rental system with two classes of resources, in which product substitutions can be made during periods of high demand. For instance, assigning a Type L job with a Type A server may correspond to upgrading a low-priority customer, whereas assigning a Type H job to a Type B server may correspond to downgrading (and compensating) a high-priority arrival.

Finally, we draw a connection between our model and a body of literature relating to the stochastic sequential assignment problem, a generalization of the coupon collection problem in which coupons must be placed into "buckets", a coupon may be eligible for placement in more than one type of bucket, and coupons cannot be removed once assigned to a bucket. Ross and Wu [25, 26] seek an assignment policy that minimizes the expected number of coupons that arrive before every bucket is filled. Derman et al. [10] consider a variant in which there are no eligibility constraints, and instead maximize a reward function that depends on the coupon, and the bucket to which it is assigned. Although our model is similar in spirit to this problem, in that "coupons" of varying types must be assigned to heterogeneous "buckets", coupons also leave the system after completing service, and we seek a policy that is optimal over an infinite horizon.

3 Model Formulation

Consider a system operating N_A Type A and N_B Type B servers. Type H and Type L jobs arrive according to independent Poisson processes with rates λ_H and λ_L, respectively. An arriving Type H job must be admitted into the system if at least one server is idle, and if this is the case, must be processed by a Type A server if one is available. Routing a Type H job to a Type B server is less desirable, but is permitted when all Type A servers are busy, to provide the job with some level of service during periods of congestion. Type L jobs can be processed by either type of server, but can be diverted from the system upon arrival, so as to reserve system resources for future Type H jobs. If a job (of either type) is admitted (with either type of server), it leaves the system after a time that is exponentially distributed with rate μ. Jobs that are diverted or that arrive when all servers

are busy immediately leave the system.

Let R_{HA} and R_{HB} denote the reward associated with assigning a Type H job to a Type A and Type B server, respectively. Similarly, let R_L denote the reward associated with admitting a Type L job (and assigning it to either type of server). We assume $R_{HA} \geq \max\{R_{HB}, R_L\}$, but we make no assumptions about the relative ordering of R_{HB} and R_L. Our goal is to find an admission control policy for low-priority jobs that maximizes the expected long-run reward collected by the system, and consider both the discounted reward and the long-run average reward criteria. We use this objective to quantify the level of service that the system is able to provide. By setting R_{HA} to be the largest reward in our model, we prioritize serving Type H jobs adequately, and we can model the trade-off between serving Type H jobs and Type L jobs through the value of R_L (relative to R_{HA} and R_{HB}).

We formulate the problem described as a Markov decision process (MDP). Let $\mathcal{S} = \{0, 1 \ldots, N_A\} \times \{0, 1 \ldots, N_B\}$ be the state space, where $(i, j) \in \mathcal{S}$ denotes the state in which i Type A servers and j Type B servers are busy. To determine the actions $\mathcal{A}(i, j)$ that are available when the system is in state (i, j), it suffices to only consider arriving Type L jobs. If either $i < N_A$ or $j < N_B$, two actions can be taken: admitting (action 1) or rejecting (action 0) the next Type L job, if one arrives during the next decision epoch. If $j < N_B$, action 1 entails assigning the call to a Type B server. Although we do not allow Type L calls to be assigned to Type A servers in this situation, this is without loss of optimality; we prove this in Proposition 4.3 below. If $j = N_B$, but $i < N_A$, then action 1 entails a Type A service. Finally, if $i = N_A$ and $j = N_B$, only a dummy action (action 0) can be taken.

Because interevent times are exponentially distributed with a rate that is bounded above by $\Lambda := \lambda_H + \lambda_L + (N_A + N_B)\mu$, our MDP is uniformizable in the spirit of Lippman [19] and Serfozo [28], and we can consider an equivalent process in discrete time. Without loss of generality, assume $\Lambda = 1$. We define a policy to be a sequence of decision rules $\pi = \{\pi^0, \pi^1, \ldots\}$, where $\pi^k : \mathcal{S} \to \{0, 1\}$ specifies a deterministic action to be taken during the k^{th} decision epoch, given the state of the system. Let Π be the set of all such policies. Given a fixed policy $\pi \in \Pi$ and an initial state (i, j), let S_k^π be the state of the system at the start of the k^{th} decision epoch, and A_k^π be the action $\pi^k(S_k^\pi)$ selected by policy π at this time. Considering first the case of discounted rewards. We define the *expected total discounted reward* collected by π to be

$$v_\alpha^\pi(i, j) = \mathbb{E}\left[\sum_{k=0}^{\infty} \alpha^k r(S_k^\pi, A_k^\pi) \,\middle|\, S_0^\pi = (i, j)\right], \tag{1}$$

where $r(s, a)$ is the reward collected when the system is in state $s \in \mathcal{S}$ and action $a \in \mathcal{A}(s)$ is taken, and $\alpha \in [0, 1)$ is the discount factor. This quantity is well-defined for each initial state (i, j), since $0 \leq r(s, a) \leq R_{HA}$ for each s and a. Next, we define $v_\alpha(i, j) = \sup_{\pi \in \Pi} v_\alpha^\pi(i, j)$. Because state

and action spaces are finite, the supremum is attained, and so $v_\alpha(i,\, j)$ denotes the total discounted reward obtained by an optimal policy, given a system initialized in state $(i,\, j)$. Theorem 6.2.6 of Puterman [24] implies that the value function v_α is the unique solution to the optimality equations:

$$v_\alpha(i,j) = T_\alpha v_\alpha(i,\, j) \tag{2}$$

$$\begin{aligned} &:= \lambda_H \Big[\mathbf{1}_{\{i<N_A\}} \big[R_{HA} + \alpha v_\alpha(i+1,\, j))\big] + \mathbf{1}_{\{i=N_A,\, j<N_B\}} \big[R_{HB} + \alpha v_\alpha(i,\, j+1)\big] \\ &\qquad + \mathbf{1}_{\{i=N_A,\, j=N_B\}} \alpha v_\alpha(i,\, j)\big] \Big] \\ &\quad + \lambda_L \Big[\mathbf{1}_{\{j<N_B\}} \max\{R_L + \alpha v_\alpha(i,\, j+1), \alpha v_\alpha(i,\, j)\} \\ &\qquad + \mathbf{1}_{\{i<N_A,\, j=N_B\}} \max\{R_L + \alpha v_\alpha(i+1,\, j),\, \alpha v_\alpha(i,\, j)\} + \mathbf{1}_{\{i=N_A,\, j=N_B\}} \alpha v_\alpha(i,\, j) \Big] \\ &\quad + i\mu\, \alpha v_\alpha(i-1,\, j) + j\mu\, \alpha v_\alpha(i,\, j-1) + (N_A + N_B - i - j)\mu\, \alpha v_\alpha(i,\, j). \end{aligned} \tag{3}$$

The first term on the right-hand side of (3) corresponds to the case when a Type H arrival occurs in the next decision epoch. The reward collected and the resulting transition depends on the system state; we capture this dependence using indicators for brevity. The remaining four terms correspond to cases where a Type L arrival, a Type A service completion, a Type B service completion, and a dummy transition occur, respectively. Note that the optimality equations (3) imply that it is optimal to accept a Type L arrival when either $j < N_B$ and $R_L > \alpha[v_\alpha(i,\, j) - v_\alpha(i,\, j+1)]$ or $i < N_A$, $j = N_B$, and $R_L > \alpha[v_\alpha(i,\, j) - v_\alpha(i+1,\, j)]$.

We also consider a finite-horizon analogue of this problem, in which we terminate the decision process after n decision epochs. Define the functions $v^\pi_{n,\,\alpha}$ and $v_{n,\alpha}$ analogously to v^π_α and v_α, but with the sum in (1) terminating at n instead of ∞. Here, we allow $\alpha = 1$. The optimality equations for this problem can be constructed analogously by replacing v_α on the left-hand side of (3) with $v_{n,\alpha}$, and occurrences of v_α on the right-hand side of (3) with $v_{n-1,\alpha}$ (and specifying the boundary condition $v_{0,\alpha}(i,\, j) = 0$ for all i and j).

Given any initial state $(i,\, j)$ and a policy π, we define the *long-run average reward* attained to be $J^\pi = \lim_{n\to\infty} v_{n,\,1}(i,\, j)/n$. By Theorem 8.3.2 of Puterman [24], J^π is well-defined and independent of $(i,\, j)$, as the Markov chain induced by π is irreducible. To see this, suppose $(i,\, j)$, $(i',\, j') \in \mathcal{S}$. Then state $(i',\, j')$ can be reached from state $(i,\, j)$ under π via $i + j$ consecutive service completions, followed by $i' + j'$ Type H arrivals. Next, define $J = \sup_{\pi\in\Pi} J^\pi$, the long-run average reward attained by an optimal policy. By Theorem 8.4.3 of [24], this can be found by solving the optimality equations

$$J + h(i,\, j) = T_1 h(i,\, j) \tag{4}$$

for J and $h(\cdot)$, where we defined the operator T_α in (2). One property of h that we use extensively is

$$h(i,j) - h(i',j') = \lim_{n\to\infty} \left[v_{n,1}(i,j) - v_{n,1}(i',j')\right], \tag{5}$$

which is proven, for instance, in Section 8.2.1 of Puterman [24].

4 Basic Structural Properties

We begin our analysis by identifying structural properties of the value functions v_α and h. The first such property we examine confirms two intuitive notions: that additional idle servers are beneficial to the system, and that an idle Type A server is preferable to an idle Type B server (implying that we prefer to assign Type L calls to Type B servers, rather than to Type A servers).

Lemma 4.1. *For all $\alpha \in [0,1)$, and $w = v_\alpha$, $v_{n,\alpha}$, or h (depending on the optimality criterion):*

1. $w(i,\, j) - w(i+1,\, j) \geq 0 \qquad i = 0,\, \ldots,\, N_A - 1,\, j = 0,\, \ldots, N_B,$
2. $w(i,\, j) - w(i,\, j+1) \geq 0 \qquad i = 0,\, \ldots,\, N_A,\, j = 0,\, \ldots, N_B - 1,$ *and*
3. $w(i,\, j+1) - w(i+1,\, j) \geq 0 \qquad i = 0,\, \ldots,\, N_A - 1,\, j = 0,\, \ldots, N_B - 1.$

Proof. We show Statement 2 holds using a sample path argument; the proofs of Statements 1 and 3 are similar. Suppose $w = v_\alpha$; the case where $w = v_{n,\alpha}$ follows via a nearly identical proof, from which the case where $w = h$ follows by leveraging Equation (5). Fix $\alpha \in [0,1)$, $i \in \{0,\, \ldots,\, N_A\}$, and $j \in \{0,\, \ldots,\, N_B - 1\}$. We construct two processes on the same probability space. Process 2 begins in state $(i,\, j+1)$ and follows the optimal policy π^*, whereas Process 1 starts in $(i,\, j)$ and uses a potentially suboptimal policy π that imitates the actions taken by Process 2.

By construction, arrivals occur simultaneously in both processes, and any job admitted by Process 2 is also admitted by Process 1. Furthermore, any service completion occurring in Process 1 also occurs in Process 2. Thus, both process move in "parallel" (in that the same changes in state occur simultaneously in both processes) until one of the following events occurs:

1. Process 2 sees a service completion that is not observed by Process 1.
2. A Type H arrival when Process 1 is in state $(N_A,\, N_B - 1)$, and Process 2 is in state $(N_A,\, N_B)$.

Either event causes both processes to *couple*, in that they transition into the same state, and behave identically from this time onward. Let Δ be a random variable denoting the difference in reward collected by the two processes until coupling occurs. By (1), $\mathbb{E}\Delta = v_\alpha^\pi(i,\, j) - v_\alpha(i,\, j+1)$. Thus, it suffices to show that $\mathbb{E}\Delta \geq 0$, as this implies $v_\alpha(i,\, j) - v_\alpha(i,\, j+1) \geq v_\alpha^\pi(i,\, j) - v_\alpha(i,\, j+1) = \mathbb{E}\Delta \geq 0$.

Indeed, prior to the coupling event, both processes observe the same transitions and collect the same rewards. When coupling occurs, Process 1 collects a reward at least as large as that by Process 2. This implies $\Delta \geq 0$ pathwise, and so $\mathbb{E}\Delta \geq 0$, as desired. □

We next establish two upper bounds: one on the benefit of an idle Type B server, and one on the benefit associated with substituting an idle Type B server with an idle Type A server.

Lemma 4.2. *For all $\alpha \in [0, 1)$, and $w = v_\alpha$, $v_{n,\alpha}$, or h, we have that*

1. $w(i, j+1) - w(i+1, j) \leq R_{HA} - R_{HB} \qquad i = 0, \ldots, N_A - 1,\ j = 0, \ldots, N_B$ *and*
2. $w(i, j) - w(i, j+1) \leq \max\{R_{HB}, R_L\} \qquad i = 0, \ldots, N_A,\ j = 0, \ldots, N_B - 1$

Proof. We show Statement 1 holds via a sample path argument; the proof of Statement 2 is similar. As in the proof of Lemma 4.1, it suffices to show that the above properties hold when $w = v_\alpha$. Fix $\alpha \in [0, 1)$, $i \in \{0, \ldots, N_A - 1\}$, and $j \in \{0, \ldots, N_B - 1\}$. Construct two processes on the same probability space. Process 1 begins in state $(i, j+1)$ and follows the optimal policy π^*, whereas Process 2 begins in state $(i+1, j)$, and imitates the actions taken by Process 1.

There is a Type A server that is idle in Process 1, but busy in Process 2, and a Type B server that is busy in Process 1, but idle in Process 2. We construct our probability space so that both units complete service simultaneously. Both processes move in parallel until one of the following occurs:

1. The coupled Type A server (in Process 1) and Type B server (in Process 2) complete service.
2. A Type H arrival occurs when Process 1 is in state $(N_A - 1, j' + 1)$, and Process 2 is in state (N_A, j'), for some $j' \in \{0, 1, \ldots, N_B - 1\}$. (In this case, Process 1 admits the job with a Type A server, and Process 2 admits the job with a Type B server.)
3. A Type L arrival occurs when Processes 1 and 2 are in states $(i' - 1, N_B)$ and $(i', N_B - 1)$, respectively, for some $i' \in \{1, 1, \ldots, N_A\}$ and Process 1 admits the job with a Type A server. (In this case, Process 2 admits the job with a Type B server.)

Let Δ be the difference in reward collected by the two processes until coupling occurs. Since $\mathbb{E}\Delta = v_\alpha(i, j+1) - v_\alpha^\pi(i+1, j)$, it suffices to show that $\mathbb{E}\Delta \leq R_{HA} - R_{HB}$. We observe that $\Delta(\omega) = 0$ on all paths ω in which events 1 or 3 occur, and $R_{HA} - R_{HB}$ on paths in which event 2 occurs (modulo the effects of discounting). Thus $\Delta \leq R_{HA} - R_{HB}$ pathwise. □

Lemma 4.2 has two implications on the structure of optimal policies.

Proposition 4.3. *If a Type H job arrives when both types of servers are available, then it is preferable to assign a Type A server. Moreover, if $R_{HB} \leq R_L$, then it is optimal to admit low-priority jobs when at least one Type B server is idle.*

Proof. We prove the first claim; the second follows using a similar argument. It suffices to show that our claim holds under the discounted reward criterion. Let (i, j) be such that $i < N_A$ and $j < N_B$, and suppose, contrary to the optimality equations (3), that we serve an arriving Type H job in this state with a Type B server. It is preferable to assign a Type A server if $R_{HA} + \alpha v_\alpha(i+1, j) \geq R_{HB} + \alpha v_\alpha(i, j+1)$, which by Statement 1 of Lemma 4.2, always holds. □

5 Optimal Policy

If $N_B = 0$, we can use the main result of Miller [21] to characterize the optimal policy.

Proposition 5.1 (Miller 1960)**.** *If $N_B = 0$, then there exists an optimal policy with the property that if it admits Type L jobs when i servers are busy, then it also does so when $i' < i$ servers are busy.*

Stated another way, *threshold-type* policies are optimal. When $N_B > 0$, it is reasonable to conjecture that a multi-dimensional analogue to this class of policies is optimal:

Definition 5.2. *A policy is of* ***monotone switching curve*** *type if there exists a monotone curve $s(\cdot)$ dividing the state space into two connected regions, one in which action 0 is taken, and one in which action 1 is taken.*

Threshold-type policies are special cases of monotone switching curve policies. We analyze the cases $R_L \leq R_{HB}$ and $R_L > R_{HB}$ separately.

5.1 The Case $R_L \leq R_{HB}$

If $R_L \leq R_{HB}$, the optimal policy is fairly structured, and our main result in this section as follows:

Theorem 5.3. *If $R_L \leq R_{HB}$, there exists an optimal monotone switching curve policy with slope of at least -1 under both the discounted reward and long-run average reward criteria.*

A monotone switching curve policy can be viewed as one that keeps some number of Type B servers in reserve to respond to future Type H arrivals, and grows this reserve as more Type A servers become busy. A bound on the slope of the switching curve implies the size of this reserve does not change dramatically in response to "small" changes in system state. In particular, if a Type A server becomes free or busy, the size of the reserve can change by at most one. To prove Theorem 5.3, we use the fact that our value functions v_α, $v_{n,\alpha}$, and h have the following structural properties:

Lemma 5.4. *If $R_L \leq R_{HB}$, then for all $\alpha \in [0, 1)$ and $w = v_\alpha$, $v_{n,\alpha}$, or h, we have that*

1. *(Convexity in j) For all $i \in \{0, 1, \ldots, N_A\}$ and $j \in \{0, 1, \ldots, N_B - 2\}$, we have that*

$$w(i, j) - w(i, j+1) \leq w(i, j+1) - w(i, j+2) \tag{6}$$

2. *(Supermodularity) For all* $i \in \{0, 1, \ldots, N_A - 1\}$ *and* $j \in \{0, 1, \ldots, N_B - 1\}$*, we have that*

$$w(i, j) - w(i, j+1) \leq w(i+1, j) - w(i+1, j+1). \tag{7}$$

3. *(Convexity in* i *when* $j = N_B$*) For all* $i \in \{0, 1, \ldots, N_A - 2\}$*, we have that*

$$w(i, N_B) - w(i+1, N_B) \leq w(i+1, N_B) - w(i+2, N_B). \tag{8}$$

4. *(Slope property) For* $0 \leq i \leq N_A - 1$ *and* $0 \leq j \leq N_B - 2$*:*

$$w(i+1, j) - w(i+1, j+1) \leq w(i, j+1) - w(i, j+2). \tag{9}$$

The proof, which we defer to the Online Appendix, follows by demonstrating that properties (6)–(9) hold when $w = v_{n,\alpha}$, via a straightforward induction argument on n, then reasoning as in Lemma 4.1 to show that the same properties hold for the value functions v_α and h.

Proof of Theorem 5.3. We consider only the discounted reward criterion, as the proof for the long-run average reward criterion is nearly identical. Consider an optimal policy π^*, and let $(i, j+1)$ be a state in which π^* takes action 1— that is, admits Type L jobs into the system. (If such a state does not exist, then our claim trivially holds.) If $j+1 < N_B$, then the optimality equations (3) imply $R_L + \alpha v_\alpha(i, j+2) \geq \alpha v_\alpha(i, j+1) \iff R_L \geq \alpha[v_\alpha(i, j+1) - v_\alpha(i, j+2)]$. Statement 1 of Lemma 5.4 implies that $R_L \geq \alpha[v_\alpha(i, j) - v_\alpha(i, j+1)]$, and so it is also optimal to take action 1 in state (i, j). If $j + 1 = N_B$, action 1 routes the Type L job to a Type A server, and

$$\begin{aligned} R_L + \alpha v_\alpha(i+1, j+1) \geq \alpha v_\alpha(i, j+1) &\iff R_L \geq \alpha[v_\alpha(i, j+1) - v_\alpha(i+1, j+1)] \\ &\implies R_L \geq \alpha[v_\alpha(i, j) - v_\alpha(i+1, j)] \\ &\implies R_L \geq \alpha[v_\alpha(i, j) - v_\alpha(i, j+1)], \end{aligned}$$

where the second line follows by Statement 2 of Lemma 5.4, and the third by Statement 3 of Lemma 4.1. Thus, it is again optimal to admit Type L jobs in state (i, j). Similar reasoning yields that the same holds in all states (i, j') where $j' < j$.

Now consider a state $(i+1, j)$ at which π^* admits Type L jobs (assuming without loss of generality that one exists). If $j < N_B$, then $R_L \geq \alpha[v_\alpha(i+1, j) - v_\alpha(i+1, j+1)]$, and Statement 2 of Lemma 5.4 implies $R_L \geq \alpha[v_\alpha(i, j) - v_\alpha(i, j+1)]$. If $j = N_B$, then $R_L \geq \alpha[v_\alpha(i+1, j) - v_\alpha(i+2, j)]$, and Statement 3 of Lemma 5.4 implies that $R_L \geq \alpha[v_\alpha(i, j) - v_\alpha(i+1, j)]$. In either case, it is optimal to admit Type L jobs in state (i, j). Similar reasoning can be used to show that Type L jobs are

also admitted in state (i', j) where $i' < i$.

Thus, if π^* admits Type L jobs in state (i, j), then it also does so in all states (i', j') for which $i' \leq i$ and $j' \leq j$. For each $i \in \{0, 1, \ldots, N_A\}$, define the function $s(i) = \max\{j : \pi^*(i, j) = 1\}$; we claim this function is nonincreasing. Indeed, if this is not the case, then there exists an i for which $s(i+1) > s(i)$, implying that for some j, the policy admits Type L jobs in state $(i+1, j+1)$, but not in state $(i, j+1)$. Contradiction.

To prove that s has slope of at least -1, it suffices to show that if π^* admits Type L jobs when the system is in state $(i, j+1)$, then it also does so in state $(i+1, j)$. Let $(i, j+1)$ be such a state, and suppose first that $j+1 < N_B$. Then $R_L \geq \alpha[v_\alpha(i, j+1) - v_\alpha(i, j+2)]$, and Statement 4 of Lemma 5.4 implies $R_L \geq \alpha[v_\alpha(i+1, j) - v_\alpha(i+1, j+1)]$. If $j+1 = N_B$, then Lemma 4.1 implies $R_L \geq \alpha[v_\alpha(i, j+1) - v_\alpha(i+1, j+1)] \geq \alpha[v_\alpha(i+1, j) - v_\alpha(i+1, j+1)]$. □

We conclude this section by noting that in the special case where $R_L = R_{HB}$, the optimal policy is simpler to characterize, as Lemma 4.2 implies that this policy must admit Type L jobs whenever Type B servers are idle. Combining this insight with Theorem 5.3 implies the existence of an optimal threshold-type policy.

5.2 The Case $R_L > R_{HB}$

As in the case where $R_L = R_{HB}$, we can leverage Lemma 4.2, and consider decision-making only in states (i, N_B), where $i < N_A$. We conjecture that a threshold-type policy is optimal here. However, we cannot reason as in Section 5.1, as the value function v_α is, in general, neither convex nor supermodular. We demonstrate this with an example. While we specifically consider the discounted reward criterion, we can adopt the example below to the case of long-run average rewards.

Example 5.1. Let $\lambda_A = 40$, $\lambda_B = 30$, $R_{HA} = 1$, $R_{HB} = 0.1$, $R_L = 0.9$, $N_A = 2$, $N_B = 28$, $\mu = 1$, and $\alpha = 0.995$. Policy iteration yields $v_\alpha(2, 28) = 28.479$, $v_\alpha(1, 28) = 29.545$, $v_\alpha(2, 27) = 28.914$, $v_\alpha(1, 27) = 30.125$, and $v_\alpha(0, 28) = 30.620$, Thus,

$$0.580 = v_\alpha(1, 27) - v_\alpha(1, 28) > v_\alpha(2, 27) - v_\alpha(2, 28) = 0.435 \quad \textit{and}$$
$$1.075 = v_\alpha(0, 28) - v_\alpha(1, 28) > v_\alpha(1, 28) - v_\alpha(2, 28) = 1.066,$$

and so v_α is neither supermodular nor convex. □

Example 5.1 establishes that the value functions v_α and h, in general, are unstructured. It can be shown that this lack of structure can be directly attributed to our assumption that decision-making with Type H jobs is forced. In particular, if the decision-maker could subject Type H jobs

to admission control, then we return to a setting in which standard techniques suffice to characterize optimal policies:

Proposition 5.5. *Consider a modified system in which Type H jobs are subject to admission control, in that they can be rejected upon arrival in any system state. Let $\tilde{v}_\alpha$, $\tilde{v}_{n,\alpha}$, and $\tilde{h}$ denote the value functions associated with optimal policies in this setting. If $R_L > R_{HB}$, then for all $\alpha \in [0,1)$, these functions are convex (that is, convex in i, and convex in j when $i = N_A$) and supermodular, and there exists an optimal monotone switching curve policy for Type H jobs.*

The proof, which we again defer to the Online Appendix, is similar to that used to prove Lemma 5.4, in that we prove that the value function $v_{n,\alpha}$ (and consequently, v_α and h) has certain structural properties via induction on n. Although forced-decision making complicates the analysis of our model, to ensure the existence of an optimal threshold-type policy, it is only necessary to show that the value functions $v_{n,\alpha}$, v_α, and h satisfy the "single-crossing" property

$$v_\alpha(i-1,\, N_B) - v_\alpha(i,\, N_B) \leq R_L \implies v_\alpha(i-1,\, N_B) - v_\alpha(i,\, N_B) \leq R_L \quad i \in \{0, \ldots, \leq N_A - 2\}. \tag{10}$$

We conjecture that this holds, but we have been unable to develop a proof. Nor have we found a counterexample, as extensive numerical experiments on a wide range of problem instances have all yielded optimal threshold-type policies. We proceed by imposing additional assumptions that allow us to identify structure, and conclude this section with two results in this vein.

5.2.1 A Sufficient Condition for Convexity

Example 5.1 corresponds to an unrealistically overloaded system in which the arrival rate greatly exceeds the system's service capacity. It may not be of practical interest to study such systems, even if threshold policies can be shown to be optimal. By restricting our attention to more reasonable parameter values, we identify conditions under which the value functions v_α and h are convex, implying the optimality of threshold-type policies. One such condition is the following:

Proposition 5.6. *Fix $\alpha \in [0,1)$. If*

$$R_L \leq R_{HB} + \frac{\mu}{\lambda_L} R_{HB} + \frac{\mu}{\lambda_H}\left(1 + \frac{\mu}{\lambda_L} + \frac{\lambda_H}{\lambda_L}\right) R_{HA}, \tag{11}$$

then for all $i \in \{0, 1 \ldots, N_A - 2\}$, $j \in \{0, 1, \ldots, N_B\}$, and $n \geq 0$, the value functions v_α, $v_{n,\alpha}$, and h are all convex in i.

The proof, which we provide in the Online Appendix, is a sample path argument. The intuition is that $v_\alpha(i, j) - v_\alpha(i+1, j)$ can be viewed as the expected difference in rewards collected by two

stochastic processes defined on the same probability space— one initialized in state $(i,\, j)$, the other in state $(i{+}1,\, j)$— until coupling occurs; call this expectation $\mathbb{E}\Delta$. The quantity $v_\alpha(i{+}1,\, j){-}v_\alpha(i{+}2,\, j)$ can be interpreted in a similar fashion; call the corresponding expectation $\mathbb{E}\Delta'$. There may be sample paths on which $\Delta > \Delta'$, and if the probability of this collection of paths is too large, we may have $\mathbb{E}\Delta > \mathbb{E}\Delta'$; this is the case in Example 5.1. Condition (11) guards against this possibility. While this condition is sufficient to prove convexity, it is certainly not necessary; numerical experiments suggest that convexity holds for a wide range of parameter values violating inequality (11).

5.2.2 Threshold Policies

Policies that are not of threshold type are unappealing from a practical standpoint, and so it may be reasonable to omit them from consideration. Restricting attention to the set of threshold-type policies may still yield value functions that are neither convex or supermodular; the optimal policy in Example 5.1 never admits Type L jobs when $j = N_B$, and thus is of threshold type.

Nonetheless, the optimal policy in this setting satisfies a monotonicity property, in that the optimal choice of threshold is nonincreasing in R_{HA}, nondecreasing in R_{HB}, and nondecreasing in R_L. More formally, for $i \in \{-1,\, 0, \ldots,\, N_A\}$, let π_i denote the threshold-type policy that admits Type L jobs in all states $(i',\, N_B)$ where $i' \leq i$. (Policy π_{-1} never assigns Type A servers to Type L jobs.) Since the set of threshold-type policies is finite, there exists an optimal policy, but it may not be unique; we break ties by selecting the policy with the highest threshold.

Proposition 5.7. *Consider a system with rewards R_{HA}, R_{HB}, and R_L ,and let π_{i^*} denote the largest optimal threshold-type policy. Suppose we modify the system so that it has rewards R'_{HA}, R'_{HB}, and R'_L, where $R'_{HA} \geq R_{HA}$, $R'_{HB} \leq R_{HB}$, and $R'_L \leq R_L$. Let π_{ℓ^*} be the largest threshold-type policy that is optimal in the modified system. Then $\ell^* \leq i^*$.*

The proof, which we again defer to the Online Appendix, involves a novel application of renewal theory. We define two stochastic processes on the same probability space (under the original reward structure), one using the optimal policy π_{i^*}, and one using a suboptimal policy π_ℓ, where $\ell > i^*$. We initialize both systems in some state $(i_0,\, j_0)$, and define renewal epochs to be the points in time at which both processes return to state $(i_0,\, j_0)$. When we consider the difference in rewards collected by the two processes during a single renewal epoch (this suffices, due to the Renewal Reward Theorem), the gap widens when we increase R_{HA}, decrease R_{HB}, or decrease R_L. Thus, any policy with a larger threshold than i^* remains suboptimal when when we modify the reward structure in this way.

Proposition 5.7 is intuitive, as if we modify rewards in our system so as to more heavily prioritize Type A responses to Type H jobs, or to decrease the importance of serving Type L jobs, we would be less willing to assign Type A servers to Type L jobs.

Our proof of Proposition 5.7 does not hinge upon our assumption that $R_L > R_{HB}$, and so it can be easily extended into a statement about the monotonicity of optimal monotone switching curves when $R_L \leq R_{HB}$. In particular, given two monotone switching curve policies s_1 and s_2, we define s_1 to be *larger* than s_2 if $s_1(i) \geq s_2(i)$ for all $i \in \{0, 1 \ldots, N_A\}$, with strict inequality holding for at least one i. That is, the set of states in which s_1 admits Type L jobs into the system is a strict superset of that associated with s_2. If we modify the reward structure (or the discount factor) as in Proposition 5.7, then the optimal monotone switching curve policy cannot increase.

Corollary 5.8. *Consider again the original and modified systems from Proposition 5.7, and suppose that $R_L \leq R_{HB}$. Let π^* be an optimal monotone switching curve policy for the original system, and s^* be the corresponding curve. Also, let π be a policy described by a monotone switching curve $s \geq s^*$. Then π is not optimal for the modified system.*

The proof follows via reasoning that is identical to that used in the proof of Proposition 5.7. Note that Corollary 5.8 does not preclude the existence of an optimal monotone switching curve $\tilde{s}^*$ for the modified system in which $\tilde{s}^*(i) > s^*(i)$ for some (but not all) values of i. It only excludes strictly larger switching curves from consideration.

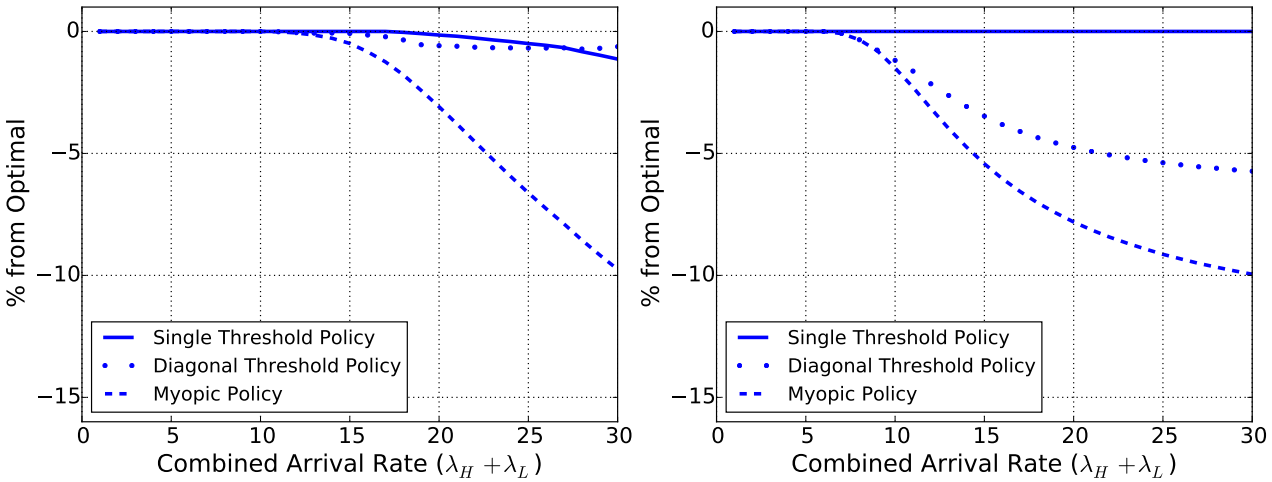
6 Computational Study

In this section, we compare our optimal policy to three heuristic policies:

- A *myopic* policy that admits every incoming job, regardless of the system state (as long as servers are available),
- A *single threshold* policy that admits Type H and Type L jobs whenever $i < N_A$, but admits Type L jobs according to a threshold policy when $i = N_A$.
- A *diagonal threshold* policy that always admits Type H jobs whenever possible, but rejects Type L jobs if the total number of busy servers in the system exceeds a predetermined threshold t—that is, in all states (i, j) where $i + j > t$.

The primary motivation for the latter two policies is that they are two-dimensional analogues of the threshold-type policies that were shown to be optimal in Miller [21]. The single threshold policy considers a one-dimensional cross-section of the state space, whereas the diagonal threshold policy does not distinguish between busy Type A and busy Type B servers.

We evaluate the performance of these three policies on a system with ten Type A and Type B servers, each of which operate at a rate of $\mu = 1$. Without loss of generality, we assume $R_{HA} = 1$, and consider two reward regimes: one that favors Type H jobs ($R_{HB} = 0.6$, $R_L = 0.4$), and one



Case 1: $R_{HB} = 0.4$, $R_L = 0.6$

Case 2: $R_{HB} = 0.6$, $R_L = 0.4$

that favors Type L jobs ($R_{HB} = 0.4$, $R_L = 0.6$). We also consider performance under varying levels of congestion, ranging from a severely underloaded system (in which $\lambda_H = \lambda_L = 0.5$) to a severely overloaded system (in which $\lambda_H = \lambda_L = 15$). In each of the MDP instances we consider, we solve for the optimal policy numerically using policy iteration. We find the optimal single threshold policy by computing the stationary distribution of the Markov chain induced by the policy with threshold t (from which long-run average reward can be easily calculated), and finding the threshold in the set $\{0, 1, \ldots, N_A\}$ that maximizes reward. We compute the optimal diagonal threshold policy in a similar fashion. Our findings are summarized in Figure 6.

When the system is lightly loaded, our heuristic policies perform optimally, as there is no need to reserve servers for Type H jobs when the system is rarely congested. As the system load increases, the optimal policy performs noticeably better, as routing decisions affect performance primarily during periods of congestion. The greedy policy, unsurprisingly, performs the poorest, particularly as arrival rates increase. In moderately-loaded systems, the myopic policy appears to struggle more when $R_L < R_{HB}$. This is also not surprising, given that setting $R_L < R_{HB}$ decreases the number of states in which it is optimal for the decision-maker to behave greedily.

The single threshold policy is optimal when $R_{HB} > R_L$, but this is a consequence of Proposition 4.3. When $R_{HB} \leq R_L$, it performs comparably to the optimal policy, but also struggles in more heavily loaded systems. The diagonal threshold policy performs quite well when $R_{HB} \leq R_L$, likely due to the fact that the structure of the policy closely mimics the monotone switching curve (with slope of at least -1) that was shown to be optimal in Theorem 5.3. However, it performs noticeably poorer when $R_{HB} > R_L$, as the policy becomes too conservative in terms of admitting Type L jobs.

Although we could consider a combination policy that uses either the single threshold or diagonal

threshold policy, depending on the reward structure, the above experiments demonstrate that there is value in taking the heterogeneity of resources into account when making decisions in our system. We observe similar behavior when we consider different systems (that are less "balanced" in terms of the relative values of N_A and N_B, and of λ_H and λ_L), and perform sensitivity analyses, but we omit the corresponding results for brevity.

7 Conclusion

In this paper, we consider the problem of routing and admission control in a service system featuring two classes of arriving jobs and two types of servers, where one arrival stream is uncontrolled, and our reward structure influences the desirability of each of the four possible routing decisions. We seek a policy that maximizes long-run reward, and consider both the discounted reward and long-run average reward criteria. We search for structure in the optimal policy by formulating the problem as a Markov decision process. When $R_L \leq R_{HB}$, we prove the existence of an optimal monotone switching curve policy with a slope of at least -1. When $R_L > R_{HB}$, we conjecture that threshold-type policies are optimal, but encounter difficulties because the value function, in general, is neither convex nor supermodular. These difficulties can be attributed directly to our modeling assumption that decision-making with Type H jobs is forced. We instead prove a sufficient (but not necessary) condition for convexity, and show that when we restrict attention to the set of threshold-type policies, the optimal policy is monotone in our choice of reward structure. Numerical experiments suggest there is value in taking heterogeneity of servers into account when making decisions.

We propose two directions for future research. The first involves finding, for the case $R_L > R_{HB}$, stronger conditions under which threshold-type policies are optimal. This may entail strengthening the sufficient condition (11) presented in Section 5.2.1, via a more refined analysis of sample paths, as our upper bound on the probability of the collection of "bad" paths is somewhat loose. Alternatively, this may involve proving our conjecture that the value functions v_α and h satisfy the single-crossing property (10). A second direction for future research is to study a model in which jobs can be placed in buffers. This is especially relevant in EMS systems, as Type L emergency calls may be queued during periods of congestion until ambulances finish with more urgent calls. Incorporating buffers into our model would likely increase the dimension of the state space. Although it may still be possible to study such a system analytically, work in this direction may entail a numerical study of near-optimal heuristic policies.

8 Acknowledgements

This research was partially supported by National Science Foundation grants CMMI 1200315, CMMI 1537394, and Army Research Office grant W911NF-17-1-0094.

9 References

[1] Altman, E., T. Jimenez, G. Koole. 2001. On optimal call admission control in resource-sharing system. *IEEE Trans. Commun.* **49**(9) 1659–1668.

[2] Bakalos, G., M. Mamali, C. Komninos, E. Koukou, A. Tsantilas, S. Tzima, T. Rosenberg. 2011. Advanced life support versus basic life support in the pre-hospital setting: a meta-analysis. *Resuscitation* **82**(9) 1130–1137.

[3] Bell, S. L., R. J. Williams. 2001. Dynamic scheduling of a system with two parallel servers in heavy traffic with resource pooling: asymptotic optimality of a threshold policy. *The Annals of Applied Probability* **11**(3) 608–649.

[4] Berman, Oded. 1981. Dynamic repositioning of indistinguishable service units on transportation networks. *Transportation Science* **15**(2) 115–136.

[5] Berman, Oded. 1981. Repositioning of distinguishable urban service units on networks. *Computers & Operations Research* **8**(2) 105–118.

[6] Bhulai, S, G. Koole. 2000. A queueing model for call blending in call centers. *IEEE Transactions on Automatic Control* **48**(8) 1434–1438.

[7] Blanc, J.P.C., P.R. de Wall, P. Nain, D. Towsley. 1992. Optimal control of admission to a multi-server queue with two arrival streams. *IEEE Transactions on Automatic Control* **37**(6) 785–797.

[8] Carrizosa, E., E. Conde, M. Muoz-Mrquez. 1998. Admission policies in loss queueing models with heterogeneous arrivals. *Management Science* **44**(3) 311–320.

[9] Chong, Kenneth C., Shane G. Henderson, Mark E. Lewis. 2015. The vehicle mix decision in emergency medical service systems. *Manufacturing & Service Operations Management* To appear.

[10] Derman, Cyrus., Gerald J. Lieberman, Sheldon M. Ross. 1972. A sequential stochastic assignment problem. *Management Science* **18**(7) 349–355.

[11] Down, Douglas G., Mark E. Lewis. 2010. The N-network model with upgrades. *Prob. Eng. Inf. Sci.* **24**(02) 171–200.

[12] Feinberg, Eugene A., Martin I. Reiman. 1994. Optimality of randomized trunk reservation. *Prob. Eng. Inf. Sci.* **8**(04) 463–489.

[13] Gans, Noah, Sergei Savin. 2007. Pricing and capacity rationing for rentals with uncertain durations. *Management Science* **53**(3) 390–407.

[14] Gans, Noah, Yong-Pin Zhou. 2003. A call-routing problem with service-level constraints. *Operations Research* **51**(2) 255–271.

[15] Harrison, J. Michael. 1998. Heavy traffic analysis of a system with parallel servers: asymptotic optimality of discrete-review policies. *The Annals of Applied Probability* **8**(3) 822–848.

[16] Jacobs, L.M., A. Sinclair, A. Beiser, R. B. D'agostino. 1984. Prehospital advanced life support: benefits in trauma. *The Journal of Trauma* **24**(1) 8–12.

[17] Jarvis, J. P. 1975. Optimization in stochastic service systems with distinguishable servers. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA.

[18] Lewis, Mark E., Hayriye Ayhan, Robert D. Foley. 1999. Bias optimality in a queue with admission control. *Proability in the Engineering and Informational Sciences* **13**(3) 309–327.

[19] Lippman, Steven A. 1975. Applying a new device in the optimization of exponential queuing systems. *Operations Research* **23**(4) 687–710.

[20] McLay, Laura A., Maria E. Mayorga. 2013. A model for optimally dispatching ambulances to emergency calls with classification errors in patient priorities. *IIE Transactions* **45**(1) 1–24.

[21] Miller, Bruce L. 1969. A queueing reward system with several customer classes. *Management Science* **16**(3) 234–245.

[22] Örmeci, E. Lerzan, Jan van der Wal. 2006. Admission policies for a two class loss system with general interarrival times. *Stochastic Models* **22**(1) 37–53.

[23] Papier, Felix, Ulrich W. Thonemann. 2010. Capacity rationing in stochastic rental systems with advance demand information. *Operations Research* **58**(2) 274–288.

[24] Puterman, Martin L. 1994. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley and Sons.

[25] Ross, Sheldon M., David T. Wu. 2013. A generalized coupon collecting model as a parsimonious optimal stochastic assignment model. *Annals of Operations Research* **208**(1) 133–146.

[26] Ross, Sheldon M., David T. Wu. 2015. A stochastic assignment problem. *Naval Research Logistics* **62**(1) 23—31.

[27] Savin, Sergei V., Morris A. Cohen, Noah Gans, Ziv Katalan. 2005. Capacity management in rental businesses with two customer bases. *Operations Research* **53**(4) 617–631.

[28] Serfozo, Richard F. 1979. Technical notean equivalence between continuous and discrete time Markov decision processes. *Operations Research* **27**(3) 616–620.

[29] Stidham, S. 1985. Optimal control of admission to a queueing system. *IEEE Transactions on Automatic Control* **30**(8) 705–713.

[30] Stidham, Shaler, Richard Weber. 1993. A survey of Markov decision models for control of networks of queues. *Queueing Systems* **13** 291–314. doi:10.1007/bf01158935. URL `http://dx.doi.org/10.1007/BF01158935`.

[31] Zhang, L. 2012. Simulation optimisation and Markov models for dynamic ambulance redeployment. Ph.D. thesis, The University of Auckland, New Zealand.