Multi-class queueing systems widely used in operations management typically experience ambiguity in real-world settings in the form of unknown parameters. For such systems, we incorporate robustness in the control policies by applying a novel data-driven percentile optimization technique that allows for (1) expressing a controller’s optimism level toward ambiguity, and (2) utilizing incoming data in order to learn the true system parameters. We show that the optimal policy under the percentile optimization objective is related to a closed-form priority-based policy. We also identify connections between the optimal percentile optimization and cµ-like policies, which in turn enables us to establish effective but easy-to-use heuristics for implementation in complex systems. Using real-world data collected from a leading U.S. hospital, we also apply our approach to a hospital Emergency Department (ED) setting, and demonstrate the benefits of using our framework for improving current patient flow policies.

**Key words**: Robustness, Model Ambiguity, Multi-Class Queueing Systems, Percentile Optimization, ED Operations

1. **Introduction**

Multi-class queueing systems require dynamic control in environments where servers must process multiple types of jobs/customers that vary with respect to holding costs, service rates, and other defining characteristics. These types of queueing systems are widely used in operations management for modeling call centers, hospitals, manufacturing lines, and service operations (among others) in which elements in the queue can be classified based on differing levels of urgency, processing time, or other attributes. For example, in a hospital Emergency Department (ED), patients are typically classified through a triage system, which assesses their relative importance based on their severity, medical complexity, or other conditions (see, e.g., Saghaian et al. (2012), Saghaian et al. (2014), and the references therein). Therefore, a natural way to analyze ED patient flow is through a multi-class queueing system which separates patients based on their attributes.¹

In such systems, when all parameters are known, many well-established policies (e.g., round robin, priority disciplines, or index-based rules) have been shown to be optimal strategies for optimizing the system’s performance (see, e.g., Van Mieghem (1995) and Buyukkoc et al. (1985)). However, the assumption that all the model parameters are perfectly known is often unrealistic, especially in

¹See, e.g., Saghaian et al. (2015) for a recent review of various models used to optimize patient flow and improve ED operations.
settings with little supporting data, inaugural system launch, or various other sources of ambiguity. An operations manager with incorrect parameter specifications may enforce policies that perform poorly. Similarly, an operations manager may not have confidence in using a policy that is obtained from a model with parameters that s/he does not fully trust. In an effort to combat parameter ambiguity, we develop strategies that take into account such model misspecifications.

In traditional robust optimization approaches, protection against model misspecification is incorporated by introducing an uncertainty set of parameters which are assumed to contain the true system parameters. The manager then engages in a game against Nature in which, given a selected policy, Nature chooses the worst-case parameters for the chosen policy from the uncertainty set. In this way, the manager prepares for the worst-case scenario, and follows a “minimax” objective by minimizing the maximum expected cost with respect to the parameter ambiguity. However, this type of robustness (1) can result in overly pessimistic policies, especially as the uncertainty set becomes large and diverse, and (2) ignores the significant potential to learn about the true system parameters from data acquired both before and after system launch. Even in scenarios where this pessimism is reduced by choosing tighter uncertainty sets that better reflect the decision-maker’s belief about the parameters (see, e.g., Ben-Tal et al. (2009), Lagoa et al. (2005), and Bertsimas et al. (2013)), the policies generated are not capable of learning from incoming data, and hence, may perform poorly when used in various real-world applications.

In order to concurrently combat the problem of model ambiguity and avoid overly conservative policies that ignore learning, we extend a technique known as percentile optimization (see, e.g., Delage and Mannor (2010) and Nemirovski and Shapiro (2006)). Our extension allows for incorporating robustness to the belief about the model parameters rather than relying on a robustness generated directly from the parameters themselves. Our approach targets policies that guarantee an expected performance over a given proportion of the parameter belief space, enabling policies to learn from incoming data. In this way, we investigate strategies where the controller learns the main model parameters (e.g., unknown service rates) through data/observations while simultaneously controlling it for superior performance. This reduces the reliance of control policies on a specific value for unknown parameters.

The incorporation of policies that exhibit data-driven learning can be directly modeled using a partially observable Markov decision process (POMDP), an extension of Markov decision processes (MDPs), which allows for (1) imperfect state knowledge, and (2) learning in a Bayesian manner. This is ideal from a learning perspective; however, in a POMDP, the controller is assumed to have an initial prior belief which is often a subjective value, guided by scarce data, error-prone expert
opinion, intuition, or instinct. For these reasons, Bayesian critics distrust such learning mechanisms, citing the unreliability of the prior specification in real-world applications. To incorporate robustness to such a prior belief (hence gaining robustness to model misspecification), we integrate percentile optimization with a POMDP approach. In this way, robustness is layered to hedge against the sensitivity to the selection of the prior belief about true model parameters: percentile optimization is utilized to identify policies that offer a certain level of performance over a percentage of the belief space based on a manager's pessimism level. Thus, our framework allows generating policies that are robust to model misspecifications (considering a manager's pessimism level), while simultaneously learning about the true model from data/observation of the system's performance in a Bayesian manner.

Our main contributions stem from extending the robust percentile optimization approach for integration with POMDPs. We find that in this framework, the percentile optimization objective reduces to the minimax and minimin objectives when the optimism level is set to its lowest and highest values, respectively. Intuitively, we show that the optimal policies under these objectives are myopic $c\mu$ priority policies, which establishes percentile optimization as a bridge between fully pessimistic and fully optimistic objectives. In addition, understanding the non-robust problem proves to be essential in finding robust policies: we find that optimal robust policies can be formed using specific non-robust policies via a geometric structure known as convex floating body. More specifically, an optimal robust policy can be found by forming a minimax objective on a convex floating body, and is further confined to be based on a belief near the “worst-case” belief point associated with the minimax objective.

Therefore, to solve the robust control problem, we first solve the non-robust problem. As the rate of observations increases, we find that a priority-based policy that can be viewed as an extension of the well-known $c\mu$ rule becomes asymptotically optimal to the non-robust problem. This policy, which we term $Ec\mu$, is myopic and prioritizes the class with the largest expected $c\mu$ value, where taking expectations dynamically combines all the possible $c\mu$ values in a “cloud” of models. Furthermore, the $Ec\mu$ policy utilizes incoming data for learning (unlike the traditional $c\mu$ rule), and is extremely simple to implement. Thus, a manager concerned about robustness can simply rely on the $Ec\mu$ policy.

The robust framework we consider is computationally ambitious since its foundation lies on an already difficult problem, namely, the POMDP. This necessitates finding tractable methods of implementing percentile optimization. To this end, using the analytical insights gained from the connection between non-robust and robust policies, the constraints implied by the convex floating
body, and the relation of $E\mu$ to the non-robust objective, we develop a heuristic for the robust problem that (1) is highly scalable to large problem instances, and (2) shows a strong performance in extensive simulation experiments. To further hedge against intractability, we develop analytical bounds to the non-robust problem based on queueing systems with no ambiguity and service rates equal to that of current expected rate. We find that these bounds are (1) tight under a variety of conditions, and (2) can be used to more effectively compute optimal robust policies by further restricting the bounds implied by the convex floating body. Additionally, since these bounds are based on non-learning policies, they can be computed in an efficient manner.

We also find that robust optimal policies are not highly sensitive to the choice of the distribution that describes the likelihood assigned to each prior, which is an important element of our percentile optimization approach. This is due to the fact that (1) this distribution is a second-order distribution (i.e., a distribution over distributions/beliefs) and (2) perturbations in this distribution results in very similar convex floating bodies. In addition, our proposed policies perform well over the spectrum of optimism as compared to the minimax and minimin policies under the Conditional Value at Risk statistic, which further suggests that our proposed framework is resilient to the choice of the second-order distribution.

Finally, we demonstrate the benefits of our approach in a real-world setting by utilizing data collected from a leading U.S. hospital, and establishing the advantages of using our framework in improving the current ED patient flow policies. Hospital EDs currently face overcrowding issues that can put patients at risk due to long waiting periods before treatment. Queueing models can be used to improve the patient flow and reduce waiting times. However, estimating the required parameters inevitably causes model misspecifications due to various ambiguities in triage misclassification probabilities, unpredictable service rates, and volatile patient characteristics among others. Our percentile optimization framework is the first study in the literature to yield policies for use in EDs that hedge against model misspecifications. We find that these highly congested EDs are well-suited to our percentile optimization framework, especially in geographical areas with uncertain/unstable patient population characteristics. This is mainly because the current policies for such EDs are based on queueing models that assume a full parameter knowledge, while higher levels of performance can be gained by managing the system with direct consideration of underlying ambiguities. Additionally, our approach explicitly avoids overly conservative policies that focus only on the “worst-case” scenarios. Since we find that percentile optimization performs well over a large spectrum of optimism/pessimism, this suggests implementation regardless of an ED manager’s disposition.
The rest of the paper is organized as follows. In Section 2, we provide a literature review of the related studies. Section 3 introduces the non-robust continuous-time formulation of our problem, which is uniformized into a discrete-time problem in Section 3.1, and lays the foundation for the percentile framework developed in Section 3.2. We provide the majority of our analytical insights in Sections 4 and 5, where we establish optimal policies for the non-robust and robust formulations, and identify upper/lower bound results. Section 6 introduces a heuristic to the robust problem that is rooted in the analytical insights generated from Section 4. In Section 7, we present various numerical experiments, discuss the application of our work for improving patient flow in EDs, and use real-world data obtained from a leading U.S. hospital to evaluate the potential benefits of our approach. Finally, in Section 8, we present our concluding remarks.

2. Literature Review

Multi-class queueing systems, which allow different types of customers based on arrival rates, holding costs, and service needs, are widely used in real-world applications. The literature surrounding such systems aims to analyze complex structures and discover their optimal control policies such as the $c\mu$ policy and its variations (see, e.g., Buyukkoc et al. (1985), Van Mieghem (1995), Saghaian and Veatch (2016), and the references therein). A common tool used to analyze and control such systems is Markov decision processes (MDPs). However, their use is limited to the unrealistic case where the decision-maker is assumed to completely know all the parameters of the model (e.g., service rates). Most notably, this includes a perfect knowledge assumption of the transition matrices that guide a system's state transitions. This assumption can be problematic in various practical applications in which service rates (or other parameters that guide the transitions) are not perfectly known. Mannor et al. (2007) and Nilim and El Ghaoui (2005) found that small changes in such parameters can result in significant differences in decision-making strategies. However, a synthesis of most studies on dynamic control in queueing systems indicates the use of tools that heavily rely on a full knowledge about the system's parameters. This is despite the fact that in practice such parameters are typically unknown and often hard to estimate.

Cavazos-Cadena and Hernandez-Lerma (1990) investigate a discrete time multi-class queue where the service probabilities are unknown and need to be estimated using observations. However, they focus on policies that become asymptotically optimal as the number of observations increase. Therefore, their main results identify various asymptotic properties of queues and associated policies. Unlike their work, we are interested in how policies perform while they are still experiencing uncertainty as opposed to after a time in which arbitrary amounts of data have been collected. Other
papers explore queueing settings with unknown parameters such as Debo and Veeraraghavan (2014) and Kremer and Debo (2015), but mainly focus on the queue’s evolution due to customer behavior under their uncertainty rather than the dynamic learning of queue dynamics from a managerial perspective. Other related work such as Argon and Ziya (2009) and Sun et al. (2014) study multi-class settings where the customer class is subject to ambiguity. Unlike this stream of research, we are interested in data-driven policies that handle ambiguities within system parameters as opposed to those in customer specification.

Robust methods applied to queueing models are largely involved with reducing the computational burden of characterizing queueing metrics and policies. Su (2006) studies a fluid approximation of a multi-class queueing model’s holding cost under a robust paradigm established by Bertsimas et al. (2004) and Bertsimas and Sim (2004). Bertsimas et al. (2011) focuses on finding bounds for performance measures through a method rooted in robust optimization, and studies the performance of this method on tandem and multi-class single server queueing networks. Jain et al. (2010) finds that a queueing network with control over traffic intensities has a simple threshold type policy under a robust objective. For more recent studies on robust techniques used in queueing systems we refer to Pedarsani et al. (2014), Bandi and Bertsimas (2012), Bandi et al. (2015), and the references therein. This stream of research is mainly aimed at increasing tractability by focusing on “worst-case” (i.e., fully pessimistic) scenarios, and establishing related performance metrics. Unlike this stream, our goal is to provide policies that (1) are more optimistic (i.e., less conservative), and (2) incorporate learning from online system-run data/observations.

Adding robustness when facing model ambiguity is a topic of significant interest to a variety of fields including economics, operations research/management, computer science, and decision theory among others. Typically, robustness in MDPs is added using “minimax” (or robust optimization) techniques, since this often results in tractable analyses as shown in Nilim and El Ghaoui (2005), Iyengar (2005), and the references therein. Other methods of incorporating robustness include regret minimization (Lim et al. (2012)), relative entropy (Bagnell et al. (2001)), and martingale-based approaches (Hansen et al. (2006)) that provide less conservative, and hence, potentially more realistic alternatives to minimax techniques. In particular, Delage and Mannor (2010) identify a robust approach applied to MDPs called percentile optimization that effectively avoids over-conservatism (see also Nemirovski and Shapiro (2006) and Wiesemann et al. (2013) for related studies). Instead of finding policies that are tailored to work well in worst-case scenarios, the percentile optimization method finds policies that maximize performance with respect to a level of belief about the true
model for a given level of optimism. However, a significant deficit in current percentile optimization approaches is the lack of ability to learn about the true model over time. In this paper, we extend the percentile optimization approach in order to use it in a POMDP setting in which system data/observations can be used for learning the true operational model. As we will show, this ability to learn itself adds a strong layer of robustness for controlling queueing systems that face model misspecification. Learning models with respect to parameter ambiguity are also discussed in Bassamboo and Zeevi (2009), which models a call center application under model ambiguity using a data-driven technique. However, their work (1) does not include any notion of robustness, and (2) focuses on near-optimal policies with performance bounds. Our work differs in modeling approach by our joint focus on learning and robustness, and in methodology by our contributions in characterizing the exact optimal policies.

Data-driven parameter learning has been incorporated in POMDPs: Ross et al. (2011) explores a finite-horizon POMDP model that updates a posterior of its parameter belief in a Bayesian manner, and Thrun (1999) investigates a POMDP in continuous action and state spaces that relies on particle filtering techniques to determine the belief state. Unlike learning mechanisms, robust methods are almost non-existent in POMDP frameworks. However, in a recent paper, Saghaian (2015) extends POMDPs to a new class termed Ambiguous POMDPs (APOMDPs) which incorporates ambiguity in transition and observation probabilities in a robust fashion. The robustness is achieved by considering $\alpha$-maximin ($\alpha$-MEU) preferences, and by incorporating the decision-maker’s temperament toward model ambiguity. Different from the APOMDP approach of Saghaian (2015), we utilize a percentile optimization objective to hedge against ambiguity.

Our work is also preceded by a great quantity of literature on robust Bayesian mechanisms that perform well under a family of prior distributions or perturbations of belief. Berger and Berliner (1986) use perturbations to the prior distribution termed “contaminations”. Berger (1990) provides a review of the earlier work of such robust techniques, but for more recent literature on the subject we refer to Ruggeri (2008) and the references therein. Robustness to prior distributions has also been studied in dynamic settings; Epstein and Schneider (2003), consider a dynamic “multiple priors” setting, and Hansen and Sargent (2007) propose robust dynamic belief updating operators when the model, prior, or both are mistrusted. (See also Gilboa and Schmeidler (1989), Maenhout (2004), and Liu (2002) for other relevant work on robustness under prior misspecification).

---

2 The percentile objective originally arose in single-period contexts (see, e.g., Charnes and Cooper (1959) and Prekopa (1995)).
3. The Multi-Class Queueing Control Problem with Model Misspecification

We begin by considering a multi-class queueing control problem where a single server is responsible for serving \( n \) classes of customers over an infinite time horizon. Unlike the traditional version of this model, we assume the controller does not know the main parameters of the system, and hence, is faced with model ambiguity/misspecification. For simplicity, and clarity of insights, we first focus on the case where the ambiguity is on service rates. To this end, we exclude dynamic arrivals to the system, and instead consider a clearing system\(^3\) version of the problem. We relax this assumption in Section 7 and Section A.4 (within Online Appendix A) by allowing for dynamic arrivals, and find that many of the major results are transferable from the clearing system. Our general approach can also be used for systems where arrival rates or other parameters are ambiguous by modifying the underlying non-robust problem and expanding the belief space to account for the additional parameters, though this increases the problem’s complexity without providing additional insights.

With \( \mathcal{N} = \{1, \ldots, n\} \) denoting the set of customer classes, we assume each customer of class \( i \in \mathcal{N} \) accrues a cost \( \hat{c}_i > 0 \) for each unit of time spent in the system. Let \( \hat{c} = (\hat{c}_1, \hat{c}_2, \ldots, \hat{c}_n) \) be the cost vector, \( \alpha \in (0, \infty) \) the discount rate, and \( \mathbf{X}(t) = (X_1(t), X_2(t), \ldots, X_n(t)) \) the vector of the number of customers in the system, where \( X_i(t) \) is number of class \( i \) customers in the system at time \( t \).

In line with many robust approaches, we begin by outlining an ambiguity set (i.e. a “cloud” of models) that is assumed to include the true model. To this end, and for tractability, we assume service times for each class are i.i.d. exponential\(^4\) random variables with unknown rates for each class; the true service rate for each class \( i \in \mathcal{N} \) is chosen by Nature at time \( t = 0 \), and lies within ambiguity set\(^5\) \( \mathcal{M}_i = \{\hat{\mu}_{i,1}, \ldots, \hat{\mu}_{i,m_i}\} \). We further assume that service times for different classes are independent. For future notational convenience, we let \( \mathcal{J}_i = \{1, \ldots, m_i\} \).

Throughout the paper, we assume \( m_i \in \mathbb{N} \), and \( \hat{\mu}_{i,j} \neq \hat{\mu}_{i,k} \) for each \( i \in \mathcal{N} \) and distinct \( j, k \in \mathcal{J}_i \). Though the ambiguity sets \( \mathcal{M}_i \) are discrete, the continuous case can be approximated arbitrarily closely by increasing the number of potential service rates \( m_i \) to make the mesh size of \( \mathcal{M}_i \) close to zero.

Over time, the controller can learn the true service rates by observing the process history which includes all previous service durations, control actions, and observations of service completions.

---

\(^3\) Clearing systems are typically used to model busy periods by focusing on the customers/jobs already in the system. The goal is then to clear the system with the minimum cost.

\(^4\) In Section 7 we relax the exponential distribution assumption. For instance, our data shows that service times in EDs are close to log-normal. As we will show, our main insights and heuristic control procedures remain effective even when the service times are not exponential.

\(^5\) The general nature of our ambiguity sets enhances the flexibility of our framework. For ambiguity sets reminiscent of other robust literature, we may choose build each \( \mathcal{M}_i \) to surround some nominal value estimated from historical data. This is in fact the strategy we use in our ED application of Section 7.1.
For Markovian systems with incomplete information, it has been shown in Bertsekas (1995) that the Bayesian belief on the unknown parameters with respect to the observed process history is a sufficient statistic. We let $\mathcal{B}$ be the set of all such sufficient statistics, i.e., the set of possible belief distributions on the system’s service parameters. Letting $m = \sum_{i \in \mathcal{N}} m_i$, each $b \in \mathcal{B}$ is an $m$-dimensional vector of the form $b = (b_{1,1}, b_{1,2}, \ldots, b_{1,m_1}, b_{2,1}, \ldots, b_{n,m_n})$ with the condition that each $b_{i,j} \geq 0$ and that $\sum_{j=1}^{m_i} b_{i,j} = 1$ for each $i \in \mathcal{N}$. In this setting, if $\hat{\mu}_i^* \in \mathcal{M}_i$ is the true (unknown) service rate for class $i \in \mathcal{N}$, $\mathbb{P}(\hat{\mu}_{i,j} = \hat{\mu}_i^* | b) = b_{i,j}$. We further assume that observations from each class are independent: observed process histories of one class do not affect the other, which is aligned with the assumption that service time of one class is independent of that of another class.

We are interested in identifying policies that prescribe which customer class the server should serve at any time, given (1) the available information summarized in the current belief about the service rates, and (2) the number of customers in each queue. To find an optimal policy, it is known that one can restrict attention to policies that are deterministic, stationary, and Markovian. Consequently, an admissible non-anticipative policy $\pi$ at each time $t$ maps the current belief and queue length information (information state) to the set of admissible actions: $\pi: \mathcal{X}(t) \times \mathcal{B} \rightarrow \mathcal{A}(\mathcal{X}(t)) = \{a \in \mathcal{N}: \mathcal{X}(t) - e_a \geq 0\}$, where $e_a$ refers to an $n$-dimensional vector with a one at the $a$-th position and zeros elsewhere. Our model described above is schematically illustrated in Figure 1.

We let $\Pi$ be the set of all admissible policies, and $\mathcal{X}^\pi(t) = (X_1^\pi(t), X_2^\pi(t), \ldots, X_n^\pi(t)) \in \mathbb{Z}_+^n$ be the number of customers in the system under policy $\pi \in \Pi$ at time $t$. To incorporate the possibility of idling the server, we designate a fictitious class 0 with no holding cost, service rate of zero, and a single customer. By “serving” class 0, the server idles since there is no penalty associated with the customer of class 0, no chance of successful service, and no change in information state $b$. For a given policy $\pi$, the expected discounted true cost the system experiences is

$$
E_\pi \left[ \int_{t=0}^{\infty} e^{-\alpha t} \hat{c}\mathcal{X}^\pi(t)^T dt | \mathcal{X}(0) \right],
$$

given the true transition parameters chosen by Nature at time $t = 0$, where the notation “$T$” represents transpose, and $E_\pi$ is expectation with respect to the probability measure induced by $\pi$. However, since the controller does not know the true transition matrix (as service rates are unknown), we are interested in the expected cost with respect to the controller’s belief:

$$
J^\pi(\mathcal{X}(0), b(0)) = E_{\pi,b(0)} \left[ \int_{t=0}^{\infty} e^{-\alpha t} \hat{c}\mathcal{X}^\pi(t)^T dt | \mathcal{X}(0) \right], \quad (1)
$$

where $E_{\pi,b(0)}$ denotes expectation with respect to both the initial belief $b(0)$ and $\pi$. We refer to $J^\pi(\mathcal{X}(0), b(0))$ as the non-robust cost, since it assumes a perfectly assigned $b(0)$ (which is inevitably
The server serves a class $a$ customer with an unknown rate $\hat{\mu}_a^*$ belonging to uncertainty set $M_a$. Hard to quantify for any decision-maker who is faced with model ambiguity. The optimal non-robust cost is then given by $J(X(0), b(0)) = \inf_{x \in \Pi} J^*(X(0), b(0))$. In what follows, we first use uniformization to work with the discrete-time model of the non-robust scenario. We then adopt percentile optimization to enable the decision-maker/controller to reduce his/her reliance on $b(0)$, and thereby make robust decisions.

3.1. A Discrete-Time Non-Robust Framework

The continuous-time Markov chain $\{X^*(t) : t \geq 0\}$ can be converted to a discrete-time equivalent using the well-known uniformization technique (Lippman (1975)). In accordance with this method, we first select a uniformized exponentially distributed random variable $\xi$ with a rate $\psi > \max_{i \in \mathcal{N}, j \in \mathcal{J}} \hat{\mu}_{i,j}$ which serves as our rate of observations made as follows. If the server completes service to a customer of class $i$ a uniformized unit of time (i.e., at the end of each period), an observation indicating the “successful” service to class $i$ is recorded. Otherwise, if no service completion is observed within this time, an observation is recorded indicating an “incomplete” service to class $i$. We note that this uniformization rate $\psi$ may be arbitrarily large so as to approximate continuous observations.

We let $\sigma$ be the Bayesian learning operator such that $\sigma(b, a, \theta)$ is an $m$-dimensional vector representing the updated belief after taking action $a$ and receiving observation $\theta$, when the prior belief is $b$. Since there are only two outcomes for observations for any given action, we let “+” signify an observed service completion (“success”) during the uniformized time period, and “−” represent an incomplete service (“failure”) in that period. In this setting, we use a discrete-time dynamic program with uniformized parameters $\mu_{i,j} = \mu_{i,j}^*/\psi$. For notational convenience, we let $E[\mu_i|b] = \sum_{j=1}^{m_i} \mu_{i,j} b_{i,j}$ be the expected service transition probability of class $i \in \mathcal{N}$ given belief $b$. In this way, the Bayesian learning operator updates belief $b$ with components $b_{i,j}$ to belief $\bar{b} = \sigma(b, a, \theta)$ with components $\bar{b}_{i,j} = \sigma(a, \theta)_{i,j}$, where $a, i \in \mathcal{N}, j \in \mathcal{J}$, and

$$
\sigma(b, a, +)_{i,j} = \begin{cases} 
\frac{\mu_{a,j} b_{a,j}}{\sum_{k=1}^{m_a} \mu_{a,k} b_{a,k}} = \frac{a_{a,j} b_{a,j}}{E[\mu_a|b]} : i = a \\
b_{i,j} : i \neq a 
\end{cases} 
$$

(2)
for a successful service observation, and
\[
\sigma(b, a, -)_{i,j} = \left\{ \begin{array}{ll}
\frac{(1-\mu_{a,j})b_{a,j}}{\sum_{k=1}^{N_a}(1-\mu_{a,k})b_{a,k}} : i = a \\
\frac{(1-\mu_{a,j})b_{a,j}}{(1-\mathbb{E}[\mu_a|b])} : i \neq a
\end{array} \right.
\]  
for a failed service observation. Equations (2) and (3) are established due to the fact that under realized parameter \(\mu_{a,j}\), the probability of successful service in a given period is \(\mu_{a,j}\) and probability of incomplete service is \((1 - \mu_{a,j})\). With this, and defining a discrete-time discounting factor \(\beta = \frac{\psi}{\psi + \alpha}\) and instantaneous cost \(cX^T = eX^T\psi + \alpha\), we can identify the non-robust optimal policy and the associated cost via the dynamic program
\[
V_{t+1}(X, b) = cX^T + \beta \left[ \min_{a \in A(X)} \left\{ \mathbb{E}[\mu_a|b] V_t(X - e_a, \sigma(b, a, +)) + (1 - \mathbb{E}[\mu_a|b]) V_t(X, \sigma(b, a, -)) \right\} \right],
\]
with the terminal condition \(V_0(X, b) = cX^T\). In this setting, taking the limit as \(t \to \infty\), we define \(V(X, b) = \lim_{t \to \infty} V_t(X, b)\), and note that \(V(X, b) = \inf_{\pi \in \Pi} J^\pi(X, b)\) (see Lemma 11 in Online Appendix B for a rigorous treatment), where \(J^\pi(X, b)\) is defined in (1). To account for evaluating non-optimal policies, we let \(V^\pi_{t+1}(X, b)\) be a value function similar to that of the dynamic program (4) with minimization operator replaced by serving the class prescribed by policy \(\pi\). Likewise, we let \(V^\pi(X, b) = \lim_{t \to \infty} V^\pi_t(X, b)\) be the infinite-horizon dynamic program value function under policy \(\pi\).

### 3.2. Gaining Robustness via Percentile Optimization

Since the controller is facing ambiguity with respect to the true model, s/he may distrust his/her initial prior on the cloud of models, \(b(0)\). The specification of \(b(0)\) is subject to model sensitivities, especially in applications in which there is little or highly variable data to perfectly quantify it. Often, the selection of a prior is a process that requires sussing out probabilities and parameter values from experts in the field, which can be a highly subjective and inaccurate task\(^6\). In order to achieve robustness to the selection of the initial prior, it is necessary to also investigate policies that are independent of the initial prior, which may still depend on the process history (actions and observations). We refer to these policies as \emph{prior-independent}, and let the set of all such policies be denoted by \(\Pi_h\).

In traditional robust optimization, one would choose a policy assuming that Nature, being an antagonistic character, picks the worst-case initial belief vector \(b(0)\) for a chosen policy. Hence, the

---

\(^6\)This is indeed a general criticism to Bayesianism and goes well beyond the queueing setting of this paper.
traditional \textit{minimax} robust objective can be defined by first considering the worst-case cost under a policy $\pi_h \in \Pi_h$:

$$R^{\pi_h}(X) = \max_{b \in B} V^{\pi_h}(X, b).$$

The cost under the minimax robust objective is then $R(X) = \inf_{\pi_h \in \Pi_h} R^{\pi_h}(X)$. In this setting, the controller assumes that Nature will pick the transition parameters that result in the maximum cost for any given policy, and chooses a policy that minimizes the cost of this worst-case outcome.

In sharp contrast to this type of robustness, which typically yields overly pessimistic control policies, is the overly optimistic \textit{minimin} objective defined by:

$$N^{\pi_h}(X) = \min_{b \in B} V^{\pi_h}(X, b),$$

and $N(X) = \inf_{\pi_h \in \Pi_h} N^{\pi_h}(X)$, under which the controller chooses a policy assuming Nature picks the transition parameters resulting in the best-case cost for any given policy. In what follows, we first show that both minimax and minimin optimal policies are within the well-known class of \textit{c\mu} policies. Thus, they (1) are fully myopic, and (2) have very simple forms.

**Proposition 1 (Minimax/Minimin \textit{c\mu} Optimal Policies).** At any state $(X, b)$, optimal policies to the minimax and minimin objectives serve classes $\arg \max_{a \in A(X)} \left( \min_{j \in J_a} c_a \mu_{a,j} \right)$ and $\arg \max_{a \in A(X)} \left( \max_{j \in J_a} c_a \mu_{a,j} \right)$, respectively.

Proposition 1 establishes that optimal policies under both minimax and minimin objectives are myopic priority disciplines (known as the \textit{c\mu} rule) with respect to the smallest and largest transition rates within the uncertainty set for each class, respectively. However, it should be noted that such policies (1) ignore the potential for learning from the system behavior, and (2) only consider the potentially unrealistic extreme best and worst-case scenarios. Hence, they provide control policies which typically perform poorly in real-world applications. To address this deficit, we next investigate how the percentile optimization approach provides a balancing alternative between these two extreme strategies, while incorporating learning about the hidden probabilities associated with the true transition parameters (i.e., service rates).

To this end, for a given $\epsilon \in [0, 1]$, we define the percentile optimization program:

$$Y^{\pi_h}(X, \epsilon) = \inf_{y_\epsilon \in [N^{\pi_h}(X), R^{\pi_h}(X)]} y_\epsilon$$

$$s.t. \quad \mathbb{P}_B \left( V^{\pi_h}(X, B) \leq y_\epsilon \right) \geq 1 - \epsilon,$$

and let $Y(X, \epsilon) = \inf_{\pi_h \in \Pi_h} Y^{\pi_h}(X, \epsilon)$ represent the optimal percentile objective. In (5), we impose that $N^{\pi_h}(X) \leq y_\epsilon \leq R^{\pi_h}(X)$ so that the value of the objective is within the most optimistic and
pessimistic values attainable for any given belief in accordance with the policy, hence enforcing “realizable” expected costs. The probability operator, $P_B$, in (6) is defined with respect to a specified probability density function over the prior belief space, where $B$ is a random variable whose realization is $b$. The percentile optimization program (5)-(6) allows us to find a chance-constrained policy; it emphasizes policy performance over a portion of the belief space. We thus term the policy that is the solution under the optimal percentile objective as $(1 - \epsilon)\%$ chance-constrained policy. Intuitively, the smaller the $\epsilon$, the more protection from poor parameter settings (representing a more pessimistic controller) since the proportion of the belief space that performs worse than $y_\epsilon$ becomes smaller.

It is important to note that the percentile objective satisfies attributes that reflect itself as a bridge between non-robust and robust objectives; expressing a manager’s optimism level is a core ambition of this type of robustness. For instance, the chance-constrained policy reduces to the minimax and minimin policies when $\epsilon$ is 0 and 1, respectively.

**Proposition 2 (Percentile/Minimax/Minimin Relationship).** The percentile objective, minimax, and minimin policies share the following relation:

1. If $\epsilon = 0$ and $P_B(B = b) > 0$ for all $b \in B$, then the optimal policy and cost under both minimax and percentile objectives are the same.

2. If $\epsilon = 1$, then the optimal policy and cost under the minimin and percentile objectives are the same.

The additional condition $P_B(B = b) > 0$ for all $b \in B$ in part (i) is necessary, since $P_B$ with zeros allows percentile objective to “ignore” certain portions of the belief space while still satisfying constraint (6). For example, if $P_B$ is the degenerate distribution with respect to a point $b$, $Y(X,0) = V(X,b)$.

### 4. Structure of Optimal Policies under the Percentile Objective

Analyzing program (5)-(6) is inherently complex both analytically and computationally. However, we are able to demonstrate that the optimal policy in our non-robust formulation becomes a priority-based policy similar to the $c\mu$ rule as the observation rate increases. We take advantage of this fact to characterize chance-constrained policies, and analyze their structure. In Section 6, we then develop an easy-to-use heuristic based on our earlier insights about the structure of the optimal policy of the percentile framework.

---

7 One may criticize the use of the percentile objective due to the potential misspecifications of $P_B$; however, it should be noted that this is a second-order distribution, and perturbations in $P_B$ result in very similar convex floating bodies, which is the geometric structure investigated in Section 4 that generates our optimal robust policies.
Importantly, we find that the solution to the percentile optimization program (5)-(6) is linked to solving the non-robust problem. Hence, we first turn our attention to the solution of the dynamic program (4). As the observation rate increases, tending toward continuous observations, the non-robust problem can be transferred to a multi-armed bandit (MAB) problem by noting that under any action, only the belief about transition parameters and number of customers in the served class (the “arms” of a MAB) change (for further discussion, see Lemma 3 in Online Appendix B). MAB problems are typically solved by indexing policies related to the expected savings in cost experienced through exclusively serving one class over time.

To take advantage of the above-mentioned connection, we term the myopic policy that serves the class \( a \in \mathcal{A}(X) \) with largest value of \( c_a \mathbb{E}[\mu_a|b] \) the “\( Ec\mu \)” policy. Thus, we denote \( \pi^{c\mu} \) as the \( Ec\mu \) policy, if it serves \( \arg\max_{a \in \mathcal{A}(X(t))} c_a \mathbb{E}[\mu_a|b(t)] \). This policy can be viewed as an extension of the traditional \( c\mu \) policy (often seen in the literature surrounding control of multi-class queueing systems) for queueing systems with ambiguous parameters. The expectation operator in this policy dynamically combines all the possible \( c\mu \) values for each class based on the belief at time \( t \). In the following theorem we show that the \( Ec\mu \) policy is asymptotically optimal for the non-robust problem as the observation rate increases.

**Theorem 1 (Ec\( \mu \) Asymptotically Optimality).** The \( Ec\mu \) policy \( \pi^{c\mu} \) is asymptotically optimal for the non-robust problem: \( \lim_{\psi \to \infty} V^{\pi^{c\mu}}(X, b) = V(X, b) \) for all \( X \in \mathbb{Z}_+^n \) and \( b \in \mathcal{B} \).
only rated on the basis of its contribution to the expected service rate, which combines all potential $c\mu$ values for that class based on $b(t)$.

Another important property of the $E\mu$ policy shown in the following proposition is that under mild conditions, $V^{E\mu}(X, b)$ is piecewise-linear over the belief space (excluding beliefs near edges and faces of $B$).^8

**Proposition 3 (Piecewise-Linearit y of the Approximate Non-Robust Value Function).**

Let $B'$ be any closed subset of $B$ such that for any $b \in B', b_{i,j} > 0$ for all $i \in \mathcal{N}, j \in \mathcal{J}_i$. If $\min_{j \in \mathcal{J}_i} c_i \mu_{i,j} \neq \min_{j \in \mathcal{J}_k} c_k \mu_{k,j}$ for any distinct pair $i,k \in \mathcal{N}$, then $V^{E\mu}(X, b)$ is piecewise-linear on $B'$.

This result is related to two facts: (i) for any given initial prior $b \in B'$ (and $X \in \mathbb{Z}^+_n$), the $E\mu$ policy is unique, unless $b$ lies on the break-points of the piecewise-linear function $V^{E\mu}(X, b)$ (see Lemma 7 and 3 in Online Appendix B), and (ii) policies can be evaluated as linear functions of the belief in any POMDP. Therefore, with respect to closed, non-zero portions of the belief space, the value function $V^{E\mu}(X, b)$ is differentiable (except at breakpoints). As we will show in Theorem 2, the differentiability of the value function strongly enhances the relationship between optimal policies of the non-robust problem and those under the robust percentile optimization program (5)-(6). Thus, the fact that we have identified an asymptotically optimal policy that exhibits this property enables us to solve the robust percentile optimization program in an efficient way. This is an important insight to our search for robust chance-constrained policies especially since, as Zhang (2010) states, there are no known conditions over which a POMDP value function is differentiable on its entire belief space.

To the purpose of finding robust chance-constrained policies, we introduce the following set of policies. Fix the initial $X$, and let $\mathcal{K}_b = \{\pi^1_b, \pi^2_b, \ldots, \pi^k_b\}$ be any finite set of optimal policies to the non-robust problem when the initial prior is $b$, and $p = (p_1, p_2, \ldots, p_k)$ be an associated distribution such that $\sum_{i=1}^k p_i = 1$. We define a policy $\pi^p_{\mathcal{K}_b}$ to be a randomized policy, if at time 0, an element of $\mathcal{K}_b$, $\pi^i_{\mathcal{K}_b}$, is chosen with probability $p_i$, which will dictate all current and future decisions.\(^9\)

Interestingly, we find that there exists a randomized policy that forms an optimal solution to the robust percentile problem. This means that there exists an optimal robust policy that randomizes between optimal non-robust policies obtained for a single belief point $b \in B$. Furthermore, we shed

---

^8 An infinite horizon POMDP value function is not always guaranteed to be piecewise-linear (see, e.g. White and Harrington (1980)).

^9 For these randomized policies, we disallow policies that are not picked at time zero for the purpose of targeting specific contours of the value function.
light on conditions (associated with the differentiability of the POMDP value function defined in equation (4) with respect to the belief space) that reduces the solution such that \( K_{b} \) is composed of a single policy, i.e., conditions under which a deterministic non-robust policy is optimal even for the robust percentile problem.

**Theorem 2 (Chance-Constrained Policy).** For any given \( \epsilon \geq 0 \), there exists a \( b^{*} \in B \) and a distribution \( p^{*} \) forming a randomized policy \( \pi_{K_{b^{*}}}^{p^{*}} \) that is optimal under the percentile optimization program (5)-(6)\(^{10}\): \( Y^{\pi_{K_{b^{*}}}^{p^{*}}}(X,\epsilon) = Y(X,\epsilon) = V(X,b^{*}) \). Furthermore, if \( V^{\pi_{b}}(X,b) \) is differentiable at \( b^{*} \), then \( K_{b^{*}} \) consists of a single policy, and hence, \( \pi_{K_{b^{*}}}^{p^{*}} \) is deterministic.

We refer to \( \pi_{K_{b^{*}}}^{p^{*}} \) as a randomized policy with belief \( b^{*} \) and distribution \( p^{*} \) (a solution to the percentile objective via Theorem 2). The above result significantly reduces the complexity of the search for optimal robust policies. Importantly, it implies that we can combine policies associated with the function \( V(X,b^{*}) \) to find chance-constrained policies. In this way, we no longer need to look at the general space of policies, but rather can focus on the class of non-robust optimal policies. Moreover, Proposition 3 shows that the differentiability condition of Theorem 2 can be met by a surface that converges to the value function. If \( b^{*} \) lies on a linear segment of the value function that is not a breakpoint, \( K_{b^{*}} \) can be composed of a single policy yielding a deterministic chance-constrained policy. Furthermore, in what follows we show there are other ways to reduce the problem to a single policy. Hence, one may not be concerned with characterizing \( p^{*} \).

However, Theorem 2 still leaves us with an important question: what belief, \( b^{*} \), should be used to form the chance-constrained policy \( \pi_{K_{b^{*}}}^{p^{*}} \) for a given percentile problem? If such a \( b^{*} \) is characterized, then the solution to the percentile problem can easily be found by a randomization of non-robust policies associated with \( b^{*} \).

The answer to the above question turns out to be closely related to the geometrical concept of the **convex floating body** first discussed by Dupin (1822). In particular, we utilize the notion of the convex floating body studied by Schutt and Werner (1990) to characterize \( b^{*} \).

**Definition 1 (Convex Floating Body).** Let \( W_{\epsilon} = \{(w,w) \in \mathbb{R}^m \times \\mathbb{R} : P_{B}(Bw^T \geq w) \leq \epsilon \} \) be the set of all half spaces that “cut off” \( \epsilon \) or less volume of the belief space \( B \) with respect to \( P_{B} \). The half space defined by each element of \( W_{\epsilon} \) therefore divides \( B \) into two pieces, one

\(^{10}\) For notational convenience, we suppress the dependency of \( p^{*} \) and \( b^{*} \) on \( \epsilon \).
of which holds $\epsilon$ or less volume (probability). An $\epsilon$-based convex floating body on $B$ is then
$L_{\epsilon} = \bigcap_{(w,w) \in W} \{ b \in B : bw^{T} \leq w \}$. We let $\delta L_{\epsilon}$ be the boundary of $L_{\epsilon}$.

Based on the above definition, a convex floating body is the region left from hyperplanes “cutting off” a specified volume ($\epsilon$) from an object. For every $b \in \delta L_{\epsilon}$, there exists a hyperplane that divides $B$ into two pieces, one which has volume less than or equal to $\epsilon$. For example, Figure 2 illustrates the convex floating body of a sphere with uniform density, which is either the empty set or another sphere. We study convex floating bodies with respect to the density measure $\mathbb{P}_{B}$ on the belief space of our priors to characterize $b^{*}$, and thereby find optimal chance-constrained policies as discussed in Theorem 2.

The literature surrounding the convex floating body is typically involved with comparing the volume of the convex floating body to its associated object. However, we are interested in its boundary $\delta L_{\epsilon}$, since it is on this surface that we can establish important links to the search for a chance-constrained policy $\pi_{K_{b^{*}}}$ of Theorem 2. For the purposes of characterizing $b^{*}$ and $p^{*}$, it is important that the convex floating body is non-empty. Fortunately, Fresen (2013) states that when $\mathbb{P}_{B}$ is a log-concave probability distribution, the convex floating body exists so long as $\epsilon \leq e^{-1}$. Hence, for many robust applications which tend toward pessimism (where $\epsilon$ is small), under common distributions, the convex floating body is guaranteed to exist. If the convex floating body is nonempty, we find that such a $b^{*}$ is found at the largest value of the non-robust problem on the boundary of the convex floating body.

Proposition 4 (Characterizing $K_{b^{*}}$). If $L_{\epsilon}$ is nonempty, then $b^{*} = \arg\max_{b \in \delta L_{\epsilon}} V (X, b)$, where $b^{*}$ satisfies $Y (X, \epsilon) = V (X, b^{*})$.

Interestingly, Proposition 4 relates percentile optimization to a minimax objective: one can search for a worst-case belief within a specified set. Since $V (X, b)$ is concave in $b$ (by the convexity results of Sondik (1971) and Smallwood and Sondik (1973)), this means that if $\delta L_{\epsilon}$ is easily characterized, we can apply gradient-based optimization to solve the problem rather than evaluating the entire surface which is computationally intractable. Although Theorem 2 states $K_{b^{*}}$ is a singleton when the value function is differentiable at $b^{*}$, the differentiability is not always guaranteed. To this end, in the proof of Proposition 4 (see Online Appendix B), we characterize $p^{*}$: we find that the distribution $p^{*}$ such that the contour $\{ b \in B | V^{p^{*}}_{K_{b^{*}}} (X, b) = V (X, b^{*}) \}$ is a subgradient hyperplane to $L_{\epsilon}$.

We note that if $L_{\epsilon}$ is nonempty, $\delta L_{\epsilon}$ always exists since closed, convex, and compact sets are equal to the convex hull of their boundary.
In general, since non-robust policies are only partially characterized (they converge to $\text{E}c\mu$ policies asymptotically), it is important to connect the $\text{E}c\mu$ policies to the percentile optimization objective. The following corollary is similar to Proposition 4 and shows that there exists a finite randomization of $\text{E}c\mu$ policies that are asymptotically optimal to the percentile objective.

**Corollary 1 (Robust $\text{E}c\mu$ Optimality).** If $\mathcal{L}_\epsilon$ is nonempty, then there exists a policy $\pi$ that is a finite randomization of $\text{E}c\mu$ policies such that

$$Y^\pi(X, \epsilon) - Y(X, \epsilon) \leq V^{\pi^{\epsilon\mu}}(X, \hat{b}) - V(X, b^\star),$$

where $\hat{b} = \arg\max_{b \in \delta \mathcal{L}_\epsilon} V^{\pi^{\epsilon\mu}}(X, b)$.

This corollary holds despite the fact that $V^{\pi^{\epsilon\mu}}(X, b)$ is not guaranteed to be concave in $b$. In fact, if it is concave in $b$, the randomized policy $\pi$ can be directly built from non-robust policies. However, if $V^{\pi^{\epsilon\mu}}(X, b)$ is not concave in $b$, we can still form the appropriate randomized policy satisfying Corollary 1 via a randomization of policies that satisfy minimax solutions within the set of $\text{E}c\mu$ policies on the boundary of the convex floating body, namely $\min_{b \in \mathcal{B}} \max_{b^2 \in \delta \mathcal{L}_\epsilon} V^{\pi^{\epsilon\mu}}(X, b^2)$.

With respect to optimal solutions to the percentile objective, additional results can further confine $\mathcal{K}_{b^\star}$ (of Theorem 2) by noting that $b^\star$ must lie near the extreme belief state with worst-case transition parameters. We denote this “worst-case” belief state by $b_0$, and note that it is composed of components

$$b_{0,i,j}^j = \begin{cases} 
1 : \text{ if } \mu_{i,j} = \min_{k \in J_i} \mu_{i,k}, \\
0 : \text{ otherwise.} 
\end{cases}$$

(7)

It can be shown (see the proof of Proposition 5) that for any prior-independent policy, $b_0$ is the worst-case (most expensive) belief state for the system. To further characterize $b^\star$, we define the concept of visibility (seen in geometry literature but repurposed for our needs).
Definition 2 (Visibility). A belief point \( b \in L_\epsilon \) is said to be visible from a reference belief \( b^1 \in B \) if \( \{ b^2 \in B : b^2 = \eta b^1 + (1 - \eta)b^1, \eta \in [0,1] \} \cap L_\epsilon = b \).

As demonstrated in Figure 3, a belief \( b \) in the convex floating body is visible from a reference belief \( b^1 \) if, on the line segment connecting these points, only \( b \) lies within the convex floating body. Of note, this implies that if the reference belief point \( b^1 \) is distinct from \( b \) and \( b \) is visible from \( b^1 \), then \( b \) must lie in the boundary (\( b \in \delta L_\epsilon \)). However, not every point on the boundary \( \delta L_\epsilon \) is visible from a reference point \( b^1 \). In the following Proposition, we show that the belief \( b^* \) (introduced in Theorem 2) must be visible from the worst-case belief state \( b_0 \) (with components defined in expression (7)).

Proposition 5 (Visibility of \( b^* \)). If \( L_\epsilon \) is nonempty, then there exists a \( b^* \) visible from the worst-case belief \( b_0 \).

Proposition 5 significantly helps us find \( b^* \) (of Theorem 2): we only need to search part of \( \delta L_\epsilon \) which is visible from \( b_0 \). Proposition 5 also can facilitate establishing effective heuristics which circumvent the calculation of the non-robust problem. For instance, Figure 4 demonstrates the implications of Proposition 5 for a uniform type \( P_B \): \( b^* \) lies somewhere on the dashed line.

The set \( L_\epsilon \) (and hence \( \delta L_\epsilon \)) typically needs to be estimated by a polytope since most distributions result in convex floating bodies with no easy closed-form representation. However, upper and lower bounds to the percentile objective can be found by optimizing over sets (in the sense of Proposition 4) that contain or are contained by \( L_\epsilon \) which converge to \( Y(X, \epsilon) \) as the sets converge to \( L_\epsilon \). The details of this are expressed in the proof of Lemma 9 in Online Appendix B. Additionally, with certain \( P_B \), the problem of estimating \( L_\epsilon \) may be altogether circumvented. This is specifically the case when \( P_B \) has the form of a spherical-type distribution defined below.

Definition 3 (Spherical Distribution). We say \( P_B \) is a spherical distribution centered at \( b_1 \), if, for any \( \epsilon \in \mathbb{R}^+ \), \( L_\epsilon = \{ b_2 \in B : \| b_2 - b_1 \| \leq d \} \) for some \( d \in \mathbb{R}^+ \), where \( \| \cdot \| \) is the \( l^2 \)-norm.

Remark 1. In cases with spherical distributions, searching for \( b^* \) is simplified even in large dimensional spaces, since we have the expression for \( \delta L_\epsilon \) and bounds based on the visibility from \( b_0 \). In such a scenario, the problem is reduced to searching for the maximum of a concave function on a sphere.

Another distribution that displays attractive attributes with respect to its convex floating body is a special case when \( P_B \) is uniform, where we can exactly specify \( L_\epsilon \). If \( n = 2, m_1 = 2, m_2 = 2 \), and
Figure 3  Belief points $b^2$ and $b^3$ are not visible from reference belief $b^1$, whereas $b^4$ is visible from reference belief $b^1$ ($n = 2, m_1, m_2 = 2$).

\[ P_B \text{ is uniform, a small modification of a result by Calgar (2010) shows that } \delta \mathcal{L}_\epsilon \text{ is given by a curve defined in four quadrants as:} \]

\[
b_{2,1} = \begin{cases} 
\frac{b_{1,1} - 1 + 0.5\epsilon}{b_{1,1} - 1} : & 0.5 \leq b_{1,1} \leq 1 - \epsilon, \quad 0.5 \leq b_{2,1} \leq 1 - \epsilon \\
\frac{b_{1,1} - 0.5\epsilon}{b_{1,1}} : & \epsilon \leq b_{1,1} \leq 0.5, \quad 0.5 \leq b_{2,1} \leq 1 - \epsilon \\
\frac{0.5\epsilon}{b_{1,1} - 1} : & \epsilon \leq b_{1,1} \leq 0.5, \quad \epsilon \leq b_{2,1} \leq 0.5 \\
\frac{0.5\epsilon}{b_{1,1} - 1} : & 0.5 \leq b_{1,1} \leq 1 - \epsilon, \quad \epsilon \leq b_{2,1} \leq 0.5
\end{cases}
\]

for $0 < \epsilon < 0.5$ as shown in Figure 4. We use this uniform case and various spherical cases in Section 7 to show that since our approach includes learning, it provides robustness to the specification of $P_B$. Therefore, even though exact closed-form representations of $\mathcal{L}_\epsilon$ in general are rare, polytope or spherical approximations are sufficient for the purposes of percentile optimization in our framework.

5. Asymptotically Tight Bounds

Although we have characterized the optimal policies of the non-robust and percentile problems, evaluating the non-robust value function $V(X, b)$ is still a computationally complex problem (see, e.g., Littman et al. (1998), Mundhenk et al. (2000), and Papadimitriou and Tsitsiklis (1987) for an in-depth discussion regarding the complexity of POMDP programs). If the value function $V(X, b)$ and the convex floating body’s boundary $\delta \mathcal{L}_\epsilon$ are known, the solution to the percentile optimization is easily characterizable (Theorem 2, Proposition 4, and Proposition 5). Therefore, we provide computationally tractable bounds to the non-robust problem that can be evaluated in closed-form to facilitate the computability of chance-constrained policies.

The bounds we form are based on the performance of (1) queues under no model ambiguity with fixed rate parameters equal to $E[\mu_i|b]$, and (2) following a particular server allocation priority rule based on the initial parameter belief. These imply that our bounds rely only on the valuation of
Figur e 4 Convex floating bodies $\mathcal{L}_\epsilon$ for $\epsilon = 0.05, 0.15, 0.25$ with $n = 2, m_1, m_2 = 2$, and uniform $\mathbb{P}_B$. To be visible from $b_0$, belief $b^*$ associated with Proposition 4 must lie on the dashed lines assuming $\mu_{1,1} < \mu_{1,2}$ and $\mu_{2,1} < \mu_{2,2}$ (in accordance with Proposition 5).

fixed priority-based policies that do not change with dynamic observations, significantly reducing the computational complexity of the problem.

For a given belief $\hat{b} \in \mathcal{B}$, consider a counterpart system identical to our original setting with the exception of the ambiguity sets being $\mathcal{M}_i = \{E[\mu_i|\hat{b}]\}$ (analogous to the original ambiguity sets $\mathcal{M}_i$). That is, the counterpart queueing system has fully known service rates based on an expectation of belief $\hat{b}$. Obviously, the optimal policy for this system is the traditional $c\mu$ rule, since all of its parameters are fully known. Let $\pi_{\hat{b}}$ denote this $c\mu$ rule and $V^{{\pi}_{\hat{b}}}(X, \hat{b})$ be the associated cost of the counterpart system under $\pi_{\hat{b}}$. It is important to emphasize that $\pi_{\hat{b}}$ exhaustively serves class $\arg\max_{a \in A(X)} c_a E[\mu_a|\hat{b}]$ until no customer of that class remains in the system, and acts only as a function of the queue state, not of belief, even when $\pi_{\hat{b}}$ is implemented in the original system. When $\pi_{\hat{b}}$ is implemented in the original system, we denote the total cost by $V^{{\pi}_{\hat{b}}}(X, \hat{b})$. Using the counterpart system’s cost and its associated policy, we can bound the non-robust cost (which is needed to calculate the robust cost; see Theorem 2 and Proposition 4) using the following proposition.

**Proposition 6 (Asymptotically Tight Bounds).** For any state $(X, \hat{b})$, the non-robust cost $V(X, \hat{b})$ is bounded by

$$V^{{\pi}_{\hat{b}}}(X, \hat{b}) \leq V(X, \hat{b}) \leq V^{{\pi}_{\hat{b}}}(X, \hat{b}).$$

Furthermore:

(i) The gap between the upper and lower bound costs decrease to zero as queue length $X_i$ increases to infinity, where $i = \arg\max_{a \in A(X)} c_a E[\mu_a|\hat{b}]$.

(ii) The gap between the upper and lower bound costs monotonically decrease to zero as $\text{Var}[\mu_i|\hat{b}]$ decrease to zero (for all $i \in \mathcal{N}$).
Both the upper and lower bounds of Proposition 6 are easily calculable via closed-form expressions (see Online Appendix B). Furthermore, under the conditions above, the lower and upper bounds become arbitrarily close approximations adding computational tractability to the problem as well as analytical insight to the relationship between our non-robust and traditional $c\mu$ policies. In particular, part (ii) of Proposition 6 supports the intuition that gathering more data on unknown service parameters can provide more accurate bound information. Part (i) of Proposition 6 provides conditions under which the myopic, non-learning policy’s cost converges to that of the optimal policy.

**Remark 2.** Since the percentile objective relies on the computation of the non-robust problem, the bound results can be easily applied to the percentile formulation as well. For instance, one can refine the search for $\arg \max_{b \in \delta L_{\epsilon}} V(X, b)$ as in Proposition 4: if the upper bound for a $b \in \delta L_{\epsilon}$ is less than the lower bound for $b' \in \delta L_{\epsilon}$, $b$ must not be the belief point $b^\ast$. Since most infinite-horizon POMDPs are calculated by finite-horizon approximations, a second application of the bounds is to use them as the terminal cost used in the finite-horizon dynamic program. That is, when evaluating the finite-horizon approximation, one can replace $V_0(X, b)$ by lower and upper bounds $\hat{V}^{b_\ast}(X, \hat{b})$ and $V^{\tilde{b}}(X, b)$, respectively. This can provide very tight bounds on the POMDP, since after a certain number of “learning periods,” where the POMDP is explicitly evaluated, the controller has collected enough information to have confidence in the true transition parameters.

### 6. An Analytically-Ro oted Heuristic Policy

Chance-constrained policies are inherently difficult to calculate, even given the analytical results established in the previous section. To circumvent complexity arising from (1) evaluating a POMDP over a belief space with high dimensionality, (2) finding the exact shape of the convex floating body, and (3) calculating optimal indices of the non-robust problem, we now introduce an effective heuristic policy. This heuristic policy operates by simply choosing the $E c\mu$ policy associated with the belief point on the convex floating body’s boundary $\delta L_{\epsilon}$ that minimizes the distance from $b_0$ (the worst-case parameter settings for each class characterized in (7)). This is typically an easy-to-perform task, especially in the cases of uniform and spherical type distributions on the belief space. Moreover, as we will show in Section 7, this heuristic performs extremely well both on randomly generated data and real-world data collected from a leading U.S. hospital.

We term the $E c\mu$ policy with expectation taken based on belief point $\arg \min_{b \in \delta L_{\epsilon}} \| b_0 - b \|$, where $\| \cdot \|$ is the $l^2$-norm, as the $(1 - \epsilon)% E c\mu$ heuristic policy. This heuristic policy takes advantage
of three main structural results of the chance-constrained policy (that we established in the previous section), while providing a much simpler version of it:

1. It assumes that the true optimal policies of the non-robust problem are $E_c \mu$, a fact supported by Theorem 1 which shows the asymptotic relationship of the optimal policies to $E_c \mu$.

2. It locates belief $\arg\min_{b \in \delta L} \| b_0 - b \|$ to be near $b^*$ (of Theorem 2) based on Proposition 4. The worst-case (most expensive) belief state is $b_0$, and through the proof of Proposition 5 (see Online Appendix B) the value function is non-increasing in $\lambda$ with respect to belief $\lambda b + (1 - \lambda) b_0$ for $\lambda \in [0, 1]$. Thus, $\arg\max_{b \in \delta L} V(X, b)$ is expected to be near $b_0$.\(^{12}\)

3. It takes advantage of the fact that $\arg\min_{b \in \delta L} \| b_0 - b \|$ satisfies Proposition 5 (since this belief is visible from $b_0$).

7. Numerical Experiments

We perform various numerical experiments in order to (1) identify the advantages of chance-constrained policies in a variety of environments under model ambiguity, (2) demonstrate the sensitivities of the underlying queueing models, (3) study the effectiveness of the proposed $E_c \mu$ heuristic in mimicking the optimal chance-constrained policies, and (4) demonstrate the implications of our results in real-world applications. To pursue these goals, we present our analyses in four parts: we (1) investigate how our policies perform over a large parameter suite but in a relatively small queueing system, (2) evaluate our proposed heuristic alongside percentile, minimax, and minimin policies in a larger system, (3) demonstrate the gap between the $E_c \mu$ and optimal (non-robust) policies, and (4) apply the $E_c \mu$ heuristic to a hospital Emergency Department setting using real-world data, and discuss its significant implications on improving the current patient flow policies.

To better understand the relative performance of our proposed robust policies, we start by considering a large parameter suite including over 1,000 parameter settings in an $n = 2, m_1 = 2$, and $m_2 = 2$ setting with four different $P_B$ distributions at their 95% chance-constrained policy. We name these $P_B$ distributions $f_1, f_2, f_3$ and $f_4$ respectively: $f_1, f_2, f_3$ are truncated multivariate normal distributions with means $\mu_1 = (0.5, 0.5), \mu_2 = (0.4, 0.4), \mu_3 = (0.6, 0.6)$ and covariance matrices $\Sigma_1 = \begin{pmatrix} 1.5 & 0.9 \\ 0.9 & 1.5 \end{pmatrix}$, $\Sigma_2 = \begin{pmatrix} 0.5 & 0.0 \\ 0.0 & 0.5 \end{pmatrix}$, and $\Sigma_3 = \begin{pmatrix} 0.5 & 0.0 \\ 0.0 & 0.5 \end{pmatrix}$ respectively. Finally, $f_4$ is the uniform distribution.

We include two non-learning robust policies (minimin and minimax) as benchmarks for the performance of our proposed robust policies and compare the policies by evaluating their total cost when each model (i.e., parameter configuration) is equally likely. That is, we assume that the true (but unknown) prior of our system is $b = (0.5, 0.5, 0.5, 0.5)$, and we evaluate the total cost under

\(^{12}\)This does not imply that $\arg\max_{b \in \delta L} V(X, b) = \arg\min_{b \in \delta L} \| b_0 - b \|$. $V(X, b)$ is only assured to be non-increasing on line segments connected to $b_0$. 
95% chance-constrained, minimax, and minimin policies. Furthermore, we assume \( c_1 = c_2 \). In every problem instance, we assume \( \mu_{2,1} < \mu_{1,1} \) and \( \mu_{1,2} < \mu_{2,2} \) so that the policy is not uniform throughout the belief space, which provides incentive for gaining additional knowledge. Further detail on this parameter suite is presented in Online Appendix A.1.

We start by investigating whether or not our proposed robust policies overcome the deficiencies we observe for non-robust policies (Observations 7, 8, and 9 established in Online Appendix A.2). Our results presented in Online Appendix A.5 provide a positive response. We next compare our proposed policies with other non-learning robust policies (minimax and minimin). In Table 1, we present the results of this comparison expressed by the average (among all models) optimality gap percentage under various policies. The optimality gap percentage for policy \( \pi \) at \( \mathbf{b} \) is defined as

\[
\frac{V^*(\mathbf{X}, \mathbf{b}) - V(\mathbf{X}, \mathbf{b})}{V(\mathbf{X}, \mathbf{b})} \%.
\]

From Table 1, we observe that on average, our proposed chance-constrained policies perform much better than the other non-learning policies. Since there is equal chance of every parameter configuration, non-learning policies serve the wrong class for a realized set of parameters 50% of the time, which results in poor performance.

Comparing the chance-constraint policies under \( f_1, f_2, f_3, \) and \( f_4 \) in Table 1 reveals yet another interesting insight: they exhibit similar performance. The reason behind this is three-fold: (1) as a property of Proposition 5, since we used 95% chance-constrained policies, each \( \mathbf{b}^* \) tends to be near \( \mathbf{b}_0 \), (2) even though the distributions \( f_1, f_2, f_3, \) and \( f_4 \) are different (e.g., they have differing covariance structures and are centered at different beliefs), their convex floating bodies are quite similar, and (3) the chance-constrained policies we propose exhibit learning. Hence, we can make the following:

**Observation 1 (Sensitivity).** The performance of chance-constrained policies is not sensitive to the choice of \( \mathbb{P}_\mathbf{b} \).

In Section 6, we introduced the \( \text{E}c\mu \) heuristic as an easy-to-implement policy that mimics the performance of robust optimal chance-constrained policies. To demonstrate the validity of the first assumption underlying this heuristic – that the optimal policies of the non-robust problem are \( \text{E}c\mu \) – in Figure 5 we depict the percent optimality gap of the \( \text{E}c\mu \) heuristic policy by comparing its cost to that of the optimal non-robust policies in a situation where \( \psi \) is small. Since we know that \( \text{E}c\mu \) becomes optimal as \( \psi \) becomes large (Theorem 1), this poses a "worst-case" scenario for the performance of the \( \text{E}c\mu \) policies. From Figure 5, we can make the following:
Observation 2 confirms that the myopic $E_c\mu$ policy provides us with a good approximation of the optimal POMDP value function (as we would expect given its asymptotic relationship to the chance-constrained policy; see Theorem 1). However, using such a rule to find the explicit surface of the POMDP value function is computationally challenging, even though the $E_c\mu$ policy is simple. This is because policy evaluation (even when a policy is known) in POMDPs is PSPACE complete (see, e.g., Mundhenk et al. (2000)). Hence, the ideal task of searching for the max of the convex floating body as in Proposition 4, even with the help of Proposition 5, is not feasible in any large problem instance. Furthermore, often times the shape of $L_\epsilon$ is difficult to determine explicitly as is the case even in the simple uniform distributions in more than two dimensions, which further complicates our search. Instead, as an approximation of $b^*$ of Theorem 2, our heuristic uses the $b \in \delta L_\epsilon$ that is closest to $b_0$, which is typically a much easier point to characterize, and is supported by Propositions 4 and 5.

To gain deeper insights into the performance of our heuristic, we simulate systems with $m_1 = m_2 = m_3 = 3$ with uniform $P_B$ in the largest inscribed sphere of the belief space. To also evaluate the robustness of our proposed heuristic vis-a-vis the optimal percentile policy as well as minimin and minimax policies, we use $C\text{Var}(q)$, which is the average cost within the most costly $q\%$ of our simulated runs. Therefore, if $S = \{s_1, \ldots, s_r\}$ is the set of the costs from a simulation of $r$ runs ordered from most costly to least costly, then

$$C\text{Var} (q) = \frac{\sum_{i=1}^{\lceil (1-q)(r-1)+1 \rceil} s_i}{\lceil (1-q)(r-1)+1 \rceil}.$$  

This statistic may roughly be seen as a function that increases in pessimism, since we use fewer low cost data points in the expectation as $q$ increases.\textsuperscript{13}

\textsuperscript{13} For instance, one would expect the minimax policy to perform well in comparison to other policies at $C\text{Var}(1)$.

\begin{table}[h]
\centering
\begin{tabular}{cccccc}
\hline
\textbf{X} & \textbf{Minimax} & \textbf{Minimin} & \multicolumn{2}{c}{\textbf{95\% Chance Constrained}} & \textbf{Chance Constrained} \\
\hline
(2, 2) & 3.17 & 15.51 & 1.84 & 2.07 & 2.2 & 1.97 \\
(2, 5) & 2.52 & 13.58 & 0.85 & 0.81 & 0.86 & 0.86 \\
(2, 10) & 1.35 & 8.21 & 0.52 & 0.51 & 0.54 & 0.49 \\
(5, 2) & 4.48 & 8.73 & 2.65 & 2.74 & 2.33 & 2.21 \\
(5, 5) & 5.01 & 10.3 & 0.85 & 0.81 & 0.75 & 0.79 \\
(5, 10) & 4.14 & 4.05 & 1.76 & 1.93 & 1.32 & 1.35 \\
(10, 2) & 4.49 & 5.15 & 0.53 & 0.51 & 0.54 & 0.55 \\
(10, 5) & 5.01 & 10.3 & 0.85 & 0.81 & 0.75 & 0.79 \\
(10, 10) & 4.34 & 5.15 & 0.33 & 0.33 & 0.35 & 0.35 \\
\hline
\textbf{Av.} & 3.76 & 8.76 & 1.10 & 1.14 & 1.04 & 1.02 \\
\end{tabular}
\caption{Performance of various robust policies over the test suite ($n = 2, m_1 = 2, m_2 = 2$).}
\end{table}
Figure 5  The optimality gap (%) of Ecµ policy when evaluated on the central prior \( \bar{\theta} (\mu_{1,1} = 0.6, \mu_{1,2} = 0.7, \mu_{2,1} = 0.5, \mu_{2,2} = 0.8) \).

Using a 95% chance-constrained policy, the Ecµ heuristic, minimin, and minimax policies, Figure 6 illustrates performance over 20,000 simulation runs.\(^{14}\) The leftmost subfigures display the raw CVar values. However, we direct our attention to the rightmost figures, which display the percentage gap (of CVars) between the four selected policies and “best” policy at a given \( q \). From Figure 6, we observe the following:

**Observation 3 (Heuristic Performance).** The Ecµ heuristic performs nearly identically to the chance-constrained policy, with a diminishing difference as the system becomes more congested.

We note that percentile optimization is not concerned about the “worst-case” scenarios, and rather optimizes based on a proportion of the belief space. Hence, being a statistic concerned with the tail performance of the distribution of costs, CVar (as compared to the expected cost) provides us with a more accurate representation of the value of robustness that percentile optimization offers. Furthermore, Figure 6 demonstrates that the proposed heuristic captures the essence of the chance-constrained policy in that it lies near the optimal policy, mirroring its performance in each simulated run. Overall, our goal to provide an alternative to the over-conservatism and over-optimism of the minimax and minimin policies seems to be met by our percentile optimization technique: it performs well at each level of the CVar statistic. Thus, we make the following:

**Observation 4 (Chance-Constrained vs. Minimin & Minimax).** The chance-constrained policy, unlike minimax and minimin policies, performs well throughout the spectrum of optimism-pessimism.

Even in cases where the chance-constrained policy is inferior to other policies with regard to the CVar statistic (e.g., the fourth row of Figure 6 with \( X = (10, 10, 10) \), where the minimin policy is seen to perform best with regard to CVar(0)), we can see that fixed priority policies (e.g., those

\(^{14}\) The associated confidence intervals are tight, so we only show the averages.
obtained under the minimin objective) miss out on the advantages of robustness that the chance-constrained policy offers throughout the optimism spectrum. Furthermore, percentile optimization is flexible: by modifying $\epsilon$, we can change our policy’s focus to be more or less optimistic to the point of becoming a minimax and minimin policy itself (Proposition 2). A similar advantage is also gained in the APOMDP framework of Saghaian (2015), where $\alpha$-maximin expected utility ($\alpha$-MEU) preferences are used.


In most hospital Emergency Departments (EDs) in the U.S., patients upon arrival are sorted by means of an urgency-based triage system into one of (typically) five classes known as Emergency Severity Index (ESI) levels. These ESI levels classify patients in descending order of urgency so that a patient of ESI 1, being in dire condition, is immediately treated, whereas patients of levels 4 and 5 are sent to a “fast track” area to be treated. Therefore, the classes served by the main section of the ED (the majority of arrivals) are those with ESI levels 2 and 3 (see, e.g., Saghaian et al. (2012),
Saghaian et al. (2014), and the references therein). We denote ESI 2 and 3 patients by “Urgent” and “Non-Urgent” patients, respectively.

As patients wait to receive treatment their condition may worsen over time and lead to adverse medical events. Sprivulis et al. (2006) and Plunkett et al. (2011) show that higher patient mortality is associated with longer waiting times prior to seeing a physician. Other research (e.g., an extremely large study on data of nearly 14 million patients by Guttmann et al. (2011)) indicate that the Risk of Adverse Events (ROAE) for patients increases with higher waiting times leading to higher mortality and hospital admission rates. Therefore, with the objective of increasing patient safety, we consider the goal of minimizing average ROAE for ED patients, and investigate optimal prioritization policies. To do so, we assume adverse events occur based on a Poisson process with a higher rate for urgent patients, and note that ROAEs in this setting play the role of holding cost parameters in our multi-class queueing model introduced earlier. The same approach is used in Saghaian et al. (2014), where the benefits of further stratifying these levels in terms of a patient’s complexity is discussed. Simple patients are those that experience only a single interaction with the physician, and thus are more quickly treated by the ED (as a server) than complex patients, whose treatment necessitates several interactions with the physician interspersed with various tests (CT scans, MRI, etc.).

Figure 7 (left) illustrates a schematic flow of patients as a multi-class queueing system. To analyze the multi-class queueing system of Figure 7 (right), one needs to estimate various parameters (e.g., service/treatment rates for each class), a task which is subject to inevitable errors. Furthermore, triaged urgency and complexity levels are subject to misclassifications, which further confuses the true parameter settings of the system. Although misclassifications can be included in the analysis when all of the parameters of the system are known, the misclassification probabilities themselves are also hard to quantify. These create model misspecification, and one needs to use robust analyses to hedge against them. However, current ED patient prioritization policies are based on analyses that ignore such ambiguities.

To demonstrate the benefits of our percentile optimization approach, we now focus on two questions: “how should EDs prioritize their patients given that they are faced with parameter ambiguity?” and “how much benefit can they get by taking ambiguities into consideration?” To answer these questions, we first model the ED from a broad perspective with non-stationary Poisson process arrivals and known service rates for all four classes: Urgent Simple (US), Urgent Complex (UC), Non-Urgent Simple (NS), and Non-Urgent Complex (NC) patients. In this way, we model the ED as a single “super-server” that represents the ED resources as a whole, and calibrate our model using
A year of data obtained from a partner hospital. This allows us to answer the patient prioritization question we raised above by noting that the ED queueing model of Figure 7 (right) is essentially a special case of our general model depicted in Figure 1 with $n = 4$.

Patient arrivals in an ED fluctuate throughout a given day, so we model these arrivals with a non-stationary Poisson process with rates changing according to a year of collected ED data as depicted in Figure 8. Furthermore, since patient LOS in our data has a lognormal distribution (as displayed in Figure 9), we fit lognormal service distributions so as to match the LOS of patients for each class of patients. Next, we design our “cloud of models” by perturbing the fitted rate parameters such that for each class $i$ with fitted rate $\hat{\mu}_{i,3}$, we incorporate four additional possible rate parameters so $\hat{\mu}_{i,1} < \hat{\mu}_{i,2} < \hat{\mu}_{i,3} < \hat{\mu}_{i,4} < \hat{\mu}_{i,5}$. Additionally, because patients become fairly stabilized upon seeing a physician, we focus on adverse events in the waiting area of EDs, and assume ROAE drops to zero once the treatment stage begins. Our model is non-preemptive, which is a reflection of physicians’ behavior in EDs: upon initiating treatment to a patient, they rarely pause treatment to serve a different patient. Since there is a possibility that the ROAE for simple patients differs from that of complex patients, we also consider a variety of such “cost” structures in our study.

Though this model allows for dynamic arrivals (unlike our model introduced in Section 3), we can still incorporate chance-constrained policies through the use of our heuristic, and compare its performance to the complexity-based prioritization policy that serves classes US, UC, NS, and NC in descending priority (demonstrated to be optimal for EDs in Saghaian et al. (2014) when ambiguity is ignored), minimax, and minimin policies. To do so, we simply modify the Bayesian belief to also incorporate arrival data. We simulate these policies, and track the non-discounted ROAE by assuming that $P_B$ is uniform. The result of 20,000 simulated days expressed in terms of the CVar statistic is reported in Figures 10 and 11 (see Online Appendix A.7 for three additional ROAE settings and in depth discussions).

A widely discussed topic in the literature surrounding EDs is the “overcrowding” issue (see e.g. Derlet and Richards (2000), Derlet et al. (2001), and Trzeciak and Rivers (2003)) that stems from
Figure 8  Per hour arrival rates of urgent and non-urgent classes based on a year of data collected from a partner hospital.

high arrival rates and limited resources (such as capacity, physicians, equipment, etc). Overcrowding in EDs results in high ROAE that endangers patients. The third row of Figures 10 and 11 demonstrates how policies perform in overcrowded EDs by considering an ambiguity set with smaller service rates (in comparison to the other ambiguity sets). We note that percentile optimization, in comparison with other policies, is especially suited for modeling overcrowded EDs. This is because under heavy congestion, chance-constrained policies learn faster, since more classes are available to serve at any given time. Using our results, we can make the following:

Observation 5 (High Traffic). The proposed percentile optimization approach performs well for prioritizing patients in EDs, especially in highly congested ones (e.g. those in busy research hospitals).

Also, Figures 10 and 11 show that, once again, the chance-constrained policies nearly dominate the entire spectrum of the CVaR statistic due to the fact that they explicitly incorporate both learning and robustness. In systems with high traffic, this learning may occur at an advanced rate, since it has available customers from each class a majority of the time the system is online. This means that while static priority policies continue to serve the “wrong” classes (due to the underlying parameter ambiguity), the chance-constrained policy quickly identifies the optimal $c\mu$ priority using the observed values. This enhances the quality the robustness percentile optimization offers, especially since one is typically more concerned with overcrowded/busy systems (EDs with low traffic have short patient LOS naturally, and are not in significant need for optimization).

Furthermore, our “clearing” system is a model often used to study queues undergoing overcrowded situations. Therefore, a more congested ED is a better fit to our original model, and while considering dynamic arrivals, we can reconfirm all the previous insights generated in the “clearing” environment. This also further confirms the results of Section A.4 (within Online Appendix A), where we show
that most of the main insights gained from the “clearing” system holds for systems with dynamic arrivals.

Of note, in communities with unstable population characteristics, where ED service rates or misclassification probabilities are more ambiguous, ED managers may incorporate percentile optimization to effectively hedge against such ambiguities. Moreover, the more the ambiguity, the better suited percentile optimization is to the system. In our simulations, this is captured through modifying our cloud of models to incorporate larger differences in the fitted parameters (see the first row of Figures 10 and 11 and compare it with the second rows). Therefore, in the presence of instability in patient population characteristics, percentile optimization stands out as a method that protects from negative consequences of focusing only on extreme outcomes, while simultaneously learning from incoming data in a theoretically sound manner. This results in the following:

**Observation 6 (Uncertain Population Characteristics).** *Percentile optimization can significantly help EDs that are placed in areas with unstable or unknown patient population characteristics to better prioritize their patients.*

8. **Conclusion**

Multi-class queues are versatile structures widely used in operations management that see a large variety of applications in both service and manufacturing sectors. In such environments, often exact parameter specification is rife with estimation errors that (if ignored) can cause system managers to implement wrong policies. We identify and implement a novel data-driven percentile optimization framework for use in POMDPs. Our unique method is accomplished by layering chance-constrained optimization on a non-robust learning model. This facilitates the learning of the true system state parameters, and allows the manager to set an optimism level indicating the extent of protection against poor parameter scenarios s/he desires. We characterize the optimal policies to both the
Figure 10 20,000 simulated days in the ED for the complexity-based prioritization, 95% Ecµ heuristic, minimin, and minimax policies, when PB is uniform, and the cloud of models perturbs the fitted service rate $\hat{\mu}_i$ in terms of two-hour time increments with $c = (3.5, 4.0, 1.75, 2.0)$. (Triage levels US, UC, NS, and NC are denoted 1, 2, 3, and 4, respectively.)

non-robust and percentile problems. We find that chance-constrained policies can be established via optimal policies to the non-robust problem. We also show that a policy which we term Ecµ is asymptotically optimal for the non-robust problem.

Since percentile optimization problems are typically computationally difficult, we introduce an analytically-rooted heuristic that can be used to effectively incorporate robustness in large problem
Figure 11 20,000 simulated days in the ED for the complexity-based prioritization, 95% Ecy heuristic, minimin, and minimax policies, when \( \mu \) is uniform, and the cloud of models perturbs the fitted service rate \( \hat{\mu} \) in terms of two-hour time increments with \( c = (4.0, 4.0, 2.0, 2.0) \). (Triage levels US, UC, NS, and NC are denoted 1, 2, 3, and 4, respectively.)

instances. To further improve computational tractability, we find asymptotically tight bounds to the non-robust problem, which can be used to efficiently solve the percentile optimization problem.

Finally, we demonstrate the efficacy of our methods numerically in both stylized and realistic environments. Using real-world data collected from a leading hospital, we observe that our approach provides promising results in improving current patient flow policies, especially for overcrowded EDs,
or those facing unknown patient population characteristics. Since ED managers typically do not fully know the service rate parameters, traditional patient flow policies based on queueing models that assume full service rate knowledge subject patients to higher risk than chance-constrained policies. Our work is the first to take into account the inevitable ambiguities in ED operations, and shed light on the dire consequences of ignoring such ambiguities.

Future research could investigate the effects of multiple servers or service stages in the system. However, each mild addition to the base model presented in our paper may lead to high degrees of complexity for the percentile optimization problem. Thus, expanding the framework can cause intractability, and hence, should be done with careful consideration. However, given the importance of considering inevitable model ambiguities in various applications of multi-class queueing systems, we expect to see more studies in this vein in the near future.

References


