Health Care Management Science (2023) 26:692–718
https://doi.org/10.1007/s10729-023-09649-0

Surgical scheduling via optimization and machine learning with long-tailed data

Yuan Shi[1] · Saied Mahdian[2] · Jose Blanchet[2] · Peter Glynn[2] · Andrew Y. Shin[2,3] · David Scheinker[2,3]

Received: 28 February 2022 / Accepted: 7 June 2023 / Published online: 4 September 2023

Abstract
Using data from cardiovascular surgery patients with long and highly variable post-surgical lengths of stay (LOS), we develop a modeling framework to reduce recovery unit congestion. We estimate the LOS and its probability distribution using machine learning models, schedule procedures on a rolling basis using a variety of optimization models, and estimate performance with simulation. The machine learning models achieved only modest LOS prediction accuracy, despite access to a very rich set of patient characteristics. Compared to the current paper-based system used in the hospital, most optimization models failed to reduce congestion without increasing wait times for surgery. A conservative stochastic optimization with sufficient sampling to capture the long tail of the LOS distribution outperformed the current manual process and other stochastic and robust optimization approaches. These results highlight the perils of using oversimplified distributional models of LOS for scheduling procedures and the importance of using optimization methods well-suited to dealing with long-tailed behavior.

Keywords Surgical scheduling · Intensive care unit · Operations research · Optimization · Machine learning · Simulation

Highlights

- Cardiovascular post-surgical lengths of stay (LOS) are critical in optimizing recovery unit congestion, but extended LOS are very difficult to predict despite the use of a wide range of machine learning models and a rich set of patient characteristics.
- Optimization models that rely on machine learning predictions of LOS without accounting for extended LOS did not improve scheduling performance (recovery unit congestion and wait times of patients) relative to current paper-based systems in use.
- We show that a data-driven conservative stochastic optimization approach that accounts for stochasticity in extended LOS can achieve scheduling performance improvements, outperforming other stochastic and robust optimization approaches.
- We apply and evaluate our methodology in the context of a pediatric academic medical center using real medical and operational data.

1 Introduction

For hospital-based surgical care, the capacity of the intensive care unit (ICU) is often a crucial downstream bottleneck. ICU bed shortages are associated with adverse patient outcomes, lost revenue from cancelled procedures, and a variety of detrimental spillover issues for numerous parts of the hospital. When the ICU is at capacity, staff may transfer patients prematurely to the step-down unit or surgical procedures may be cancelled at the last-minute with adverse impact on patient and family experience, hospital reputation, finances and staff morale [5].

While most patients that require an ICU bed require it urgently, elective surgical procedures are scheduled as far as a year in advance. Our primary goal in this work was to develop a scheduling model that would reduce post-surgical bed congestion in practice in the presence of difficult-to-predict, long-tailed LOS data. Our secondary goals were

Yuan Shi and Saied Mahdian contributed equally to this work.

✉ David Scheinker
dscheink@stanford.edu

[1] Massachusetts Institute of Technology, Cambridge, MA 02139, USA

[2] Stanford University, Stanford, CA 94305, USA

[3] Lucile Packard Children's Hospital, Palo Alto, CA 94304, USA

to address the challenges associated with optimization in the presence of long-tailed empirical data. Numerous works have examined how to optimize surgical scheduling in order to reduce recovery-bed congestion. Despite the importance of post-surgical LOS, studies commonly use only synthetic data to evaluate model performance, discarding empirical LOS data after they have been used to fit the parameters of a distribution. We illustrate limitations associated with using synthetic, rather than empirical, data.

1.1 Setting

This work was performed at the heart center of a high surgical volume pediatric academic medical center (PAMC) in the United States. The heart center uses three operating rooms, 26 cardiovascular ICU (CVICU) beds shared by elective and urgent surgical patients, and an acute care unit. In recent years, growing demand for elective surgical services and fixed CVICU capacity has resulted in numerous surgical cancellations. From September 2019 to May 2020, 84% of cardiovascular surgery cancellations happened on the 40% of days with 23 or more patients in total in the CVICU. In particular, 35% of cancellations happen on the 11% of days with more than 10 elective surgical patients in the CVICU (See Fig. 1). This suggests that a significant fraction of cancellations may be prevented if optimized scheduling smooths elective surgical patient CVICU occupancy.

1.2 Overview

Numerous theoretical works using optimization to schedule elective surgical cases have demonstrated substantial reductions to ICU congestion in numerical experiments [20, 22, 27, 35]. However, relatively little work has been implemented or has demonstrated measurable improvements in practice. Most existing positive results rely on the use of synthetic data, e.g., a log-normal distribution fit to approximate empirical data from the institution studied, to represent procedure duration and post-operative length of stay. Such common practice, combined with the lack of real-life implementations of these algorithms, leaves many real world challenges of surgical scheduling unaddressed and unidentified. Meanwhile, many works have examined empirical data in order to predict the length of stay (LOS) of patients in the ICU (see, for example, [8, 21, 32]), but little has been done in utilizing such LOS predictions for surgical scheduling.

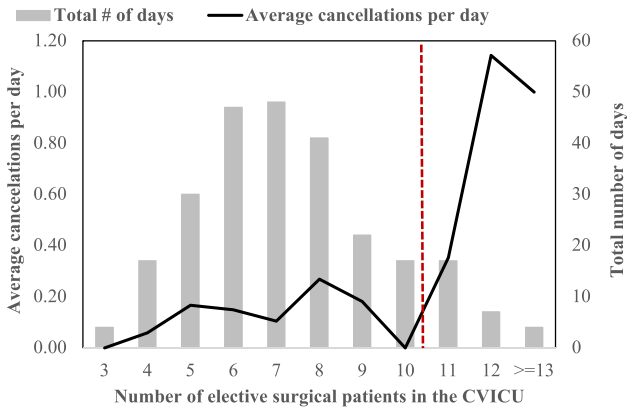


Fig. 1 High ICU occupancy by elective patients is associated with high rate of surgical cancellations due to limited bed capacity

We examine reducing ICU congestion by optimizing the scheduling of complex pediatric cardiovascular surgical procedures, using real data from the hospital for patient flow and post-operation ICU LOS. Our approach combines machine learning for LOS forecasting, optimization for surgical scheduling, and simulation for performance evaluation.

To motivate the problem, we first establish the theoretical upper bound of system performance by running a deterministic optimization formulation with known *a priori* information on post-operation LOS. The results demonstrate significant reductions in CVICU congestion and patient wait times, compared to the institution's current manual and primarily paper-based process. We then develop predictive models using machine learning for LOS, based on patient data available at the time the procedure is scheduled. When known LOSs are replaced with point-predictions based on machine learning, deterministic optimization is not sufficient to reduce congestion without increasing wait times. This is despite access to a very rich set of patient characteristics for LOS prediction. The key bottleneck is identified to be the lack of accurate predictions of LOS, especially for a small group of patients with very long LOS. This has a disproportionately large impact on optimization performance due to the long tail of the distribution of LOS. We observe that patients with similar characteristics may vary drastically in realized LOS, and patient populations display temporal non-stationarity, both suggesting inherent unpredictability of LOS at the time of scheduling. This inspired the development of stochastic and robust optimization algorithms that incorporate LOS probability distributions instead of point estimates.

We develop a novel data-driven modeling framework for scheduling cardiovascular surgery under uncertain LOS. The framework combines machine-learning predicted LOS probability distributions, rolling information update, and optimization methods. Under this framework, three scheduling algorithms are formulated: two based on stochastic optimization, Standard-RSO and Conservative-RSO, and one robust optimization formulation, RRO. The optimal schedules given by each algorithm are evaluated through simulation using historical patient arrivals and LOS.

Machine learning models are used to obtain both point-estimates of LOS and its probability distribution characterized by predictive errors. Standard-RSO, which considers

both under- and over-estimation errors, fails to reduce ICU congestion versus the status quo. In contrast, both Conservative-RSO and RRO, with carefully tuned parameters, managed to reduce ICU congestion without increasing average patient wait times. We attribute the promising performances of the latter two algorithms to their targeted design focusing on addressing common under-estimation for prolonged LOS by machine learning predictions. Moreover, Conservative-RSO achieves better average and worst-case performance compared to RRO in reducing ICU congestion.

The main contributions of our work are two fold. First, the negative results associated with deterministic optimization as well as Standard-RSO offer important lessons that are generally applicable to others seeking to optimize surgical scheduling for complex patients. We highlight the importance of evaluating model performance using empirical data or simulation that fully captures the long-tailed nature of patient LOS. In addition, as hospital operations and medical practices are constantly changing, non-stationarity in LOS should be taken into account when developing simulating and evaluating data-driven models. Our negative results, revealed through simulation with empirical data, provide clear evidence against extrapolating good performance on synthetic data using standard distributional assumptions (e.g. lognormal distribution) into real world scenarios. These findings may also explain the dearth of studies reporting successful implementation and measured improvements of similar approaches.

Our second contribution is to propose a promising algorithm, Conservative-RSO, that is specifically designed to address the challenge of long-tailed ICU LOS in the surgical scheduling problem. Compared to Standard-RSO, Conservative-RSO demonstrates that good performance can be achieved by incorporating careful design choices without increasing computational complexity; the comparison with RRO further suggests that computationally complicated models do not always translate into better performance in practice. We believe the promising performance of Conservative-RSO helps identify a direction of future research on data-driven optimization methods to address one of the most challenging obstacles to efficient hospital operations.

The remainder of the paper is structured as follows. Section 2 provides a review of related literature on surgical scheduling optimization and LOS predictions. Section 3 presents our modeling framework under deterministic LOS estimations, introducing formulations for both offline optimization and rolling optimization. These deterministic models are used to establish performance upper-bounds, given accurate predictions of patient LOS. Section 4 presents the development of machine learning models for LOS predictions and discusses the poor performance and challenges of using deterministic optimization with machine learning prediction. Section 5 presents our modeling framework using predicted LOS distributions, and introduces the three algorithms, Standard-RSO, Conservative-RSO and RRO. The performances of these algorithms under numerical experiments are presented in Section 6. Section 7 discusses insights from the design and performance of algorithms, as well as their limitations and potential extensions. Finally, Section 8 provides concluding remarks on implications of the work, best practices in applying schedule optimization in practice, and directions for future research.

Gurobi 9.0.3 [12] and Python were used to implement and solve all mixed integer optimization algorithms developed in the paper. All optimization problems were solved using the compute facility of Stanford Research Computing Center and Stanford University. On this shared facility, each optimization problem was solved with two CPUs and with either 8GB or 16GB of memory for each CPU.

2 Related work

2.1 Surgery scheduling

Extensive research has been carried out on scheduling of patients to improve operating room (OR) performance. Most of the literature on elective surgery scheduling focuses on OR room capacities without considering the capacity of the subsequent recovery units. The variability of surgical procedure durations is usually considered as the primary source of stochasticity and as the primary challenge to surgical scheduling. We refer to [25, 36] for a comprehensive review and focus on research that studies the surgical scheduling problem under the constraint of limited downstream capacity.

Deterministic formulations of surgical schedule optimization under limited downstream capacity have been examined in [9, 11, 13, 24]. Hsu et al., [13] considered optimizing surgical scheduling with limited capacity at the Post Anesthesia Care Unit (PACU) to minimize the number of nurses needed for the PACU. The case duration and the recovery times in the PACU are treated as deterministic. Similarly, [24] considered the surgical scheduling problem with deterministic recovery times in the PACU and the ICU for all patients under the assumption of perfect information. Guinet and Chaabane [11] also investigates surgery scheduling with downstream PACU capacity constraints. It proposes a two step optimization procedure where patients are first assigned to ORs and dates. In the second step surgery times are assigned to patients. More recently, [9] developed a combined machine learning and optimization approach to reduce congestion in the PACU, where the recovery time in the PACU is estimated using machine learning predictions. Fairley et al., [9] reports good results based on simulations using empirical data, but does

not report the results of implementation. Most of the existing deterministic formulations focus on the PACU capacity as the key downstream bottleneck with limited discussions on the ICU capacity. Compared to the recovery time at the PACU (which is typically in the order of hours), ICU LOS may be multiple days or months and involves greater variance than single-day PACU stays. The substantial difference in variability between CVICU long-tailed LOS and PACU recovery time are why the approach in [9] is not applicable in the present setting.

Given the multi-period nature and significant uncertainties involved in ICU capacity planning, a number of mathematical formulations have been proposed in recent years. Min and Yih [20] provided a well-known benchmark instance of surgery scheduling with ICU capacity constraints using a stochastic mixed integer programming model. The work uses sample average approximation and assumes the LOS in the ICU to be random with known distributions (arbitrary with finite support). Zhang et al., [35] proposes a two level time horizon for surgery scheduling. In the first level, patients are selected from a waitlist to be scheduled with a timeframe (e.g. a week) using approximate dynamic programming. In the second level, selected patients are scheduled using a sample average approximation method similar to [20]. Other stochastic programming approaches built on the work in [20]; we refer to [27] for a review of these formulations.

Besides stochastic optimization models, [22, 27] develop optimization methods for robust performance of surgery scheduling under worst-case realizations of LOS or LOS distributions. Most relevant to our work is [22]. The authors formulated a two-stage robust optimization approach to reduce congestion in downstream capacities, and developed solution techniques which we adapt and apply to our setting. Shehadeh and Padman [27] proposes an alternative approach towards uncertainties using distributionally robust optimization.

All of the above-mentioned papers on stochastic or robust optimization evaluated model performance using synthetic data based on strong distributional assumptions, and none has been implemented in practice. In contrast, we use real LOS data in evaluating our model performance.

In addition, [10] considered a surgery scheduling problem focusing on minimizing downstream costs including overcapacity costs at the ICU. Models for this problem proposed in [10] and others such as [1] are concerned with the *tactical* problem of allocating OR block of times to surgical specialties to optimize patient flow into the downstream units. In contrast, our work and others' mentioned above focus on the *operational* problem of assigning individual patients to surgical time blocks.

To our knowledge, we are the first to study a combined machine learning and optimization approach for schedule optimization to reduce congestion at the CVICU: an environment that includes non-stationary, long-tail LOS behavior. We are also the first to propose a stochastic formulation for surgical scheduling that is specifically designed to address significant variability in the tail of the LOS distribution. We use real world post-operation LOS data for LOS prediction, model parameter tuning and performance evaluation, reflecting the difficulties of working in a setting with non-stationary operations and patient volumes and long-tailed LOS distributions.

Lastly, to ensure that our model can be effectively applied in real-world settings, we utilize a rolling schedule optimization approach that involves sequential decision-making as new information on patient LOS and arrivals become available. Similar rolling horizon policies have been studied in many healthcare service scheduling settings, for example, in [2, 3, 14, 26]. In comparison to existing work, we further integrate such policies with our LOS prediction procedure: uncertainties in LOS are updated overtime during patients' stay in the ICU. Such integrated rolling schedule optimization approach improves predictive accuracy overtime and addresses the dynamic nature of ICU patient scheduling for real-world applications.

2.2 Post-surgery LOS prediction

In developing our machine learning models for LOS prediction, we refer to a separate line of literature focusing on predictions of post-surgery (in particular cardiac surgeries) LOS in the intensive care units.

Most work suggests that post-surgery LOS prediction at admission time - either using predictive modeling or by expert opinion - is challenging especially in the case of prolonged ICU LOS.

One common way to predict prolonged ICU LOS is through binary classification, such as in [8, 32]. However, binary classification does not provide the level of granularity required for optimizing surgical scheduling in our context.

For regression-based models, [33] builds and evaluates multivariate regression models using data from 246 hospitals for heart failure patients. The model achieves a modest R^2 value of 4.8%, where only 1.2% of variation is explained by patient characteristics. Similarly, [6] shows through univariate regression that only 12% of the variation could be explained by patient characteristics and general hospital characteristics in aggregate for patients with a primary diagnosis of acute myocardial infarction. More complicated LOS regression models using machine-learning for cardiac surgery patients are artificial neural networks developed in [17, 29] and adaptive neuro-fuzzy systems explored in [19]. Although machine-learning based models generally result in a higher R^2 value, predictive accuracy for patients with prolonged LOS remains low. For instance, the neural network in [29] achieves an overall accuracy of over 60% but is unable

to predict any LOS above 15 days, and the model in [17] with $R^2 = 0.41$ underestimates the LOS for almost all patients with actual LOS of longer than 100 hours.

More generally, [21, 31] demonstrate that prolonged LOS predictions are challenging even for experienced physicians. In addition, [16] suggests that information gathered at admission did not have a significant impact on the identification of patients with prolonged ICU LOS. The variables that had the greatest impact on prolonged ICU LOS were those measured on day 5 of ICU stays. Although LOS prediction using patient features collected during and after surgery tends to achieve a higher level of accuracy [16, 28, 34], most of these data are not available when a procedures is being scheduled weeks to months in advance.

In our attempt to develop a predictive model for post-operation LOS, we identified the same difficulty as observed by the others in dealing with the long-tailed, non-stationary distribution of LOS for complex cardiovascular surgeries. This challenge motivated our design of a stochastic optimization formulation that specifically addresses the unpredictability and the consistent underestimation of prolonged LOS by predictive models. We discuss the formulation in detail in Section 5.

3 Deterministic formulations for schedule optimization

We first study the problem of surgical scheduling where the LOS in the ICU are treated as deterministic. We develop two optimization models - offline and rolling deterministic optimization - given operational constraints at the PAMC and some *a priori* information on patient LOS.

3.1 Offline deterministic optimization

We formulate the offline surgery scheduling problem as a mixed integer program (MIP) with the objective of smoothing out CVICU elective census overtime to reduce cancellations without creating excessive delays.

Define set $D = \{1, 2, \ldots, N_d\}$ as the set of available dates in the time period of interest, and define set $P = \{1, 2, \ldots, N_p\}$ as the set of all elective surgical patients that are to be scheduled within the time period. We define binary decision variables,

$$x = \{x_{d,p} : d \in D, p \in P\},$$

where $x_{d,p} = 1$ if patient p is scheduled for surgery on day d and otherwise $x_{d,p} = 0$.

For the offline problem, it is assumed that both patient arrivals, P, and the LOS of each patient, l_p, are known in advance. Given x and $\{l_p : p \in P\}$, the ICU overflow variable, u_d for all $d \in D$, counts the number of elective patients on day d that exceed ICU capacity, c.

The offline deterministic optimization problem is formulated as below.

$$\min_x \quad \sum_{p\in P}\sum_{d=1}^{N_d}(d - d_p^{min})^+ x_{d,p} + \beta \sum_{d=1}^{N_d} f(u_d) \tag{1a}$$

$$\sum_{d\in D} x_{d,p} = 1 \quad \forall p \in P \tag{1b}$$

$$y_{d,p} = \sum_{d'=\max(d-l_p+1,1)}^{d} x_{d',p}, \quad \forall p \in P, d \in D \tag{1c}$$

$$\sum_{p\in P} y_{d,p} \le c + u_d \quad \forall d \in D \tag{1d}$$

$$x \in Q^{op} \tag{1e}$$

$$y_{d,p}, x_{d,p} \in \{0, 1\}, \quad u_d \ge 0 \tag{1f}$$

We use the notation $(\cdot)^+ = \max(\cdot, 0)$.

In the offline problem, the first term of the objective function 1a represents the total wait time for all patients. Here, d_p^{min} denotes the earliest date by which patient p will be ready for his surgery. The second term, $\beta \sum_{d\in D} f(u_d)$, is the total cost associated with ICU overflow, weighted by constant β. We set the function $f(\cdot)$ to be a convex, piece-wise linear function of u_d such that it approximates the quadratic cost function, u_d^2. The purpose of the convex cost function is to impose higher penalty for greater overflow in order to smooth out and minimize large peaks in ICU occupancy. This also reflects the highly non-linear increase in the possibility of cancellation as elective ICU occupancy increases, as demonstrated in Fig. 1. We provide the exact MIP formulation of $f(\cdot)$ in Appendix A.

Constraint Eq. 1b enforces the assignment of every patient to exactly one date of surgery. The second constraint Eq. 1c calculates if a patient needs an ICU bed on a given day based on her assignment and LOS. The third constraint 1d calculates the ICU overflow, u_d, on each day given ICU bed capacity, c.

For the last constraint 1e, we use Q^{op} to represent any remaining institution-specific operational constraints. Scheduling for complex procedures is commonly constrained by single surgeon-patient matches, surgeon and OR room availability, which may differ depending on the context. For our formulation, Q^{op} is tailored to the context of the PAMC as explained below.

Patient availability. When scheduling patients for surgery, there is often a clinically determined upper bound on patient wait time and also a corresponding lower bound that accounts

for surgical delays due to patient availability, travel arrangement and necessary insurance or health checks. We incorporate this constraint by imposing windows of availability for individual patients as hard constraints in our formulation, similar to [11]. In practice, the window of availability of patients can be estimated case by case upon arrival and hence used as inputs for optimization.

For the purpose of simulation, since a patient's actual window of availability at the time of arrival is not recorded in historical census and surgery data from the PAMC, we use the following heuristics to obtain an estimation. The earliest available date of patient p for surgery, denoted as $d_p^{min} \in D$, is set to be date that is half of the lead time prior to her originally scheduled surgery date. The latest available date, denoted as $d_p^{max} \in D$, is set to be 90 days after the actual surgery date. This heuristic ensures that the set of feasible scheduling solutions is not too restricted, and any resultant increase in wait time will be penalized by the first term in the objective function.

Surgeon and OR Room availability. Each procedure requires one OR room and takes a pre-specified amount of time for the patient's assigned surgeon to complete. We incorporate the following constraints imposed at the institution and observed in the data. First, a maximum of 45 hours of surgery in OR is available on every weekday, where each surgeon performs operations for no longer than 15 hours every day[1]. During implementation, we also incorporate surgeon-specific day-offs in the formulation. For example, one of the surgeons does not perform surgeries on Mondays.

While many past work focused their attention on the stochastic nature of surgical durations (see, for example, [22]), we do not consider uncertainty in surgery duration in this work. In our setting and in similar highly specialized surgical settings, there is a dedicated OR room and OR team with a charge to accommodate procedures that run well beyond standard operating hours. Uncertainty in surgical duration is thus not a bottleneck in our setting; surgical duration is treated as deterministic and uses the actual length of procedure in optimization[2].

Complex surgeries. On top of the regular capacity constraints above, special arrangements are often required for complex, full-day procedures. At the PAMC, Pulmonary Artery Reconstruction (PAR) surgery is a type of highly complex procedure that requires special treatment in the algorithm. The surgeons only perform PAR surgeries on certain weekdays ('PAR days') and not the others. If a PAR surgery is scheduled for a surgeon on any day, no other surgery can be performed by the same surgeon on the same day. In addition, the surgeons need to take at least a one-day break in between PAR surgeries, i.e., PAR surgeries cannot be scheduled on two consecutive days for the same surgeon.

Full mathematical formulations of Q^{op} for our context and implementation details are provided in Appendix A.

Here, the weight for ICU overflow, β, and ICU bed capacity, c are parameters to be tuned. For the value of β, we select a value among

$$\beta \in \{5, 10, 25, 50, 100, 200\}$$

so that the model achieves the the most reduction on ICU overflow without excessively lengthening the average and median patient wait times compared to the original surgical schedule. Meanwhile, we determine the value of c based on the minimum achievable ICU occupancy upper bound given true historical on LOS, i.e.,

$$c = \min_{x} \max_{d \in D} \sum_{p \in P} y_{d,p} \tag{2}$$

s.t. Constraints (1b), (1c), (1e), and (1f)

Solving optimization problem 2 yields $c = 8$ at the PAMC, which is used for all numerical experiments for the rest of the paper.

3.2 Rolling deterministic optimization (RDO)

In practice, the arrival and LOS of incoming patients are unknown and performing offline optimization over a one-year horizon is not practical. We thus develop a rolling-horizon alternative to the offline formulation, where patients are scheduled in batches with estimated LOS at the time of scheduling. Meanwhile, we dynamically observe the realization of LOS for patients who have undergone surgery in the previous period and update LOS estimates for these patients.

We define a sequence of scheduling days, $s_b \in D$, for $b \in \{1, 2, \ldots, B\}$ in chronological order. On each scheduling day s_b, scheduling is performed for all patients who arrived between s_{b-1} and $s_b - 1$. We use P_b to denote the batch of patients who are scheduled on day s_b. These patients may be scheduled for surgery any time during the scheduling horizon, $s_b, s_b + 1, \ldots, s_b^{max}$, where $s_b^{max} \in D$ is the latest date that any patient in P_b can be scheduled for surgery, i.e., $s_b^{max} = \max_{p \in P_b} d_p^{max}$. Then, the scheduling horizon is rolled forward one period at the next scheduling day, for patients who arrived between s_b and $s_{b+1} - 1$. Figure 2 illustrate the timeline of such rolling schedule optimization. To simplify implementation, we set $s_b^{max} = N_d$ for all b, where

[1] Surgeries that took longer than 15 hours in reality were treated as 15 hours for feasibility.

[2] Alternatively, we also estimated each patients' surgical duration using the average surgical duration for each procedure type in the empirical data. This approach produces similar results to that using the actual surgical durations.

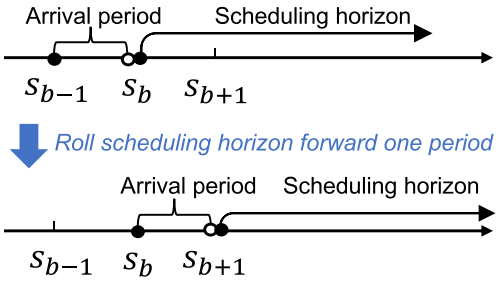


Fig. 2 Timeline visualization for rolling schedule optimization

N_d is sufficiently large to cover the scheduling horizon for all batches.[3]

Note that scheduling decisions for patients in P_b are also influenced by past patients, $P_b^{past} = \bigcup_{k=1}^{b-1} P_k$ who are still in the system. This includes all patients who arrived before day s_{b-1} who are currently staying in the ICU on day s_b, as well as those who have their surgery scheduled for days within the current scheduling horizon. Surgical and ICU capacities are adjusted accordingly to account for these past scheduling decisions.

To introduce the mathematical formulation, we use the same set of notations as the offline problem. Note that the binary decision variables $x_{d,p}$, $y_{d,p}$ are now defined for all $d \in D$ and $p \in P_b \cup P_b^{past}$. Since scheduling decisions for patients in P_b^{past} are already made, we let constants $\tilde{x} = \{\tilde{x}_{d,p} : d \in D, p \in P_b^{past}\}$ denote previous scheduling decisions. This constraint implicitly updates surgical and ICU capacities available for incoming patients by accounting for patients who have been previously scheduled. The deterministic batch optimization problem (BOP) on scheduling day s_b is formulated as below.

$$\min_{x} \quad \sum_{p\in P_b} \sum_{d=s_b}^{N_d} (d - d_p^{min})^+ x_{d,p} + \beta \sum_{d=s_b}^{N_d} f(u_d) \tag{3a}$$

$$x_{d,p} = \tilde{x}_{d,p} \quad \forall p \in P_b^{past}, d \in D \tag{3b}$$

$$\sum_{d=s_b}^{N_d} x_{d,p} = 1, \quad \sum_{d=1}^{s_b-1} x_{d,p} = 0 \quad \forall p \in P_b \tag{3c}$$

$$y_{d,p} = \sum_{d'=\max(d-l_p+1,1)}^{d} x_{d',p} \quad \forall p \in P_b \cup P_b^{past}, d \in D \tag{3d}$$

$$\sum_{p\in P_b\cup P_b^{past}} y_{d,p} \le c + u_d \quad \forall d \in D \tag{3e}$$

$$x \in Q^{op} \tag{3f}$$

$$y_{d,p}, x_{d,p} \in \{0,1\}, \quad u_d \ge 0 \tag{3g}$$

The objective function 3a mirrors that of the offline formulation in objective function 1a. Constraint Eq. 3b requires that patients scheduled previously are not re-scheduled for a different date. Constraint Eq. 3c requires that every patient in P_b is scheduled for one surgery day no earlier than s_b. Constraints 3d-3g mirror constraints 1c-1f in the offline problem. Note that constraints 3e and 3f are influenced by both patients in P_b and those in P_b^{past}. This is because the latter may also undergo surgery and/or need ICU beds after the scheduling date s_b, thus competing for ICU resources and other resources in Q^{op} (e.g. surgeon and OR room availability) with patients in P_b. Full mathematical formulation with our context-specific Q^{op} and implementation details are provided in Appendix A.1.

Surgical schedule optimization can thus be operationalized in practice by solving deterministic BOP sequentially for $b = 1, 2, \ldots, B$, with an appropriately chosen sequence of scheduling days, $\{s_b\}$ and estimates of patient LOS, $\{l_p : p \in P_b \cup P_b^{past}\}$. For patients in P_b, estimations of l_p can be made based on patient features at the time of arrival (see Section 4). For patients in P_b^{past} who have received their surgeries, l_p estimates can be progressively updated as uncertainties in LOS realize. We formalize this information update procedure below.

Information Update Procedure for deterministic BOP. At the start of each batch b, the value of l_p for all $p \in P_b^{past}$ is updated as follows.

(a) If the patient has undergone surgery and has been discharged from ICU by scheduling day s_b, her realized LOS is observed and we update l_p to be the true LOS.
(b) If the patient is in the ICU on day s_b having stayed for m days, then l_p is updated based on partially realized LOS and any additional post-operations information.
(c) If the procedure of the patient is scheduled after s_b, there is no change to l_p.

For simulation purpose only, we use a simplified heuristic for patients under (b). We assume the true LOS can be evaluated with certainty for those soon to be discharged (within 5 days), and thus update l_p to be the true LOS for those patients. For the remaining patients, we set $l_p \leftarrow \max\{m + 10, l_p\}$. In practice, l_p can be re-evaluated on a case-by-case basis for each patient based on any new information available.

On every scheduling day, patient arrivals P_b and any realization of additional LOS information for P_b^{past} are observed.

[3] Note that we still enforce the constraint that each patient is scheduled no later than d_p^{max} through Q^{op}.

An optimal solution of deterministic BOP is then obtained and implemented for all patients in P_b. The existing schedule and required surgical resources are updated accordingly. We describe this rolling optimization process in Algorithm 1.

Algorithm 1 Rolling Deterministic Optimization (RDO)

Initialize with $b = 1$, $P_b^{past} = \emptyset$, $\tilde{x} = \emptyset$.
for $b = 1$ **to** B **do**
  1. Information Update. Patient arrival P_b is realized between s_{b-1} and $s_b - 1$; obtain LOS estimates, $\{l_p : p \in P_b\}$.
  for $p \in P_b^{past}$ **do**
    Update LOS estimate l_p using the *Information Update Procedure.*
  end for
  2. Schedule Optimization. Solve deterministic BOP with s_b, P_b, P_b^{past}, $\tilde{x}$, l_p $\forall p \in P_b \cup P_b^{past}$; implement the optimal solution x^* for all $p \in P_b$.
  3. Schedule and Capacity Update.
  $P_{b+1}^{past} \leftarrow P_b^{past} \cup P_b$
  $\tilde{x} \leftarrow \tilde{x} \cup x^*$
end for

In practice, the sequence of s_b is an additional design choice that can be set based on hospital practices and nature of different procedures. In all our experiments (both the deterministic formulation and stochastic or robust formulations in later sections), we set scheduling days to be the start of every month, and schedule patients in monthly batches given our context. For complex, elective cardiovascular procedures, there is usually a significant lead time between patient referral and when a patient undergoes surgery. For our patient cohort, the median lead time is 59 days and the average lead time is 79 days. Therefore, monthly scheduling is chosen as an example and proof of concept given the context: although patients need to wait for being scheduled, the effect of this additional wait time is limited in this case as the total wait time is being minimized under our optimization algorithms. In Appendix C, we also re-run our numerical experiments for biweekly scheduling days and show that the resultant insights are similar. For other types of procedures that require faster scheduling, s_b can be adjusted accordingly such that scheduling can happen more frequently in smaller batches. We discuss the implications of this design further in Section 7.

3.3 Model performance under perfect information

Before introducing machine learning approaches for estimation of LOS, we first evaluate the performance of our models in the real world setting of the PAMC, using true historical LOS as model inputs, $\{l_p\}$. The purpose of this exercise is to identify the upper bound of the achievable level of operational performance versus the status quo, which sets the optimal performance benchmark for remaining numerical experiments.

We evaluate the performance of our scheduling model by considering the 596 elective cardiovascular surgeries that were carried out at the PAMC from September 2018 to March 2020. Patient arrivals are modeled using deterministic, historical arrivals. An optimal schedule was generated using the optimization models given the true LOS of each patient, and the resultant daily CVICU census was simulated based on the optimal surgical schedule and the actual LOS.

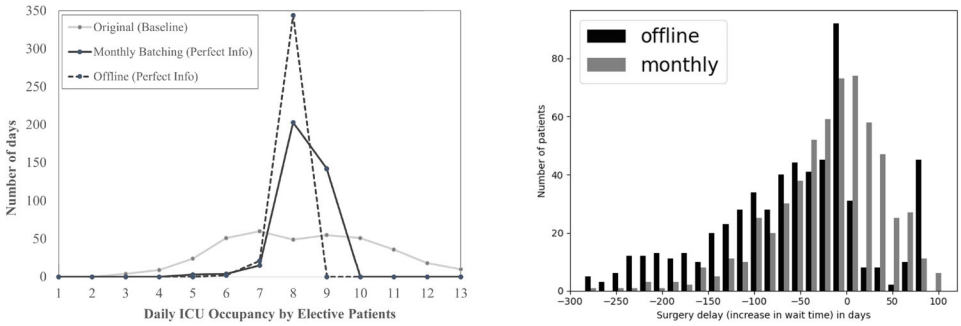
The performance of the optimization models is assessed using two metrics: 1) percentage reduction in the number of high ICU occupancy days (10 or more elective patients) in the ICU versus the historical status quo, and 2) median and average reduction in patient wait times compared to the original schedule. Good performance corresponds to a measurable reduction in the number of high ICU occupancy days without increasing patient wait times.

When evaluating our models, we treat the first six months of the time period (Sept 2018 - March 2019) as a warm-up period[4] and focus on model performance in the one-year period from March 2019 to March 2020.

The performance of offline and rolling deterministic optimization is visualized in Figs. 3a and 3b. In both figures, we use $\beta = 100$ for offline deterministic optimization and $\beta = 25$ for rolling deterministic optimization. With perfect information on LOS, both models are able to eliminate the number of days with 10 or more elective patients in the ICU (Fig. 3a). Meanwhile, the offline model schedules 83.8% of the patients no later than the status quo (Fig. 3b), and the median and average reductions in wait time are 51.0 days and 66.1 days respectively. Rolling scheduling in monthly batches results in slightly longer wait times as the patients are not scheduled until the start of the following month. Still, 60.4% of the patients are scheduled no later than the statues quo (Fig. 3b)), and the median and average reductions in wait time are 10.0 and 21.94 days respectively. Overall, both models achieve significant reduction in both high occupancy days in the ICU and patient wait times.

The above results show that it is theoretically possible to eliminate the number of days with more than 10 elective patients in the ICU given perfect information on LOS at the time of scheduling. Such performance improvement can be achieved along with a reduction in average and median wait times.

[4] 86 out of the 596 surgeries performed from September 2018 to March 2020 are associated with patient arrivals prior to September 2018. For these patients, the earliest possible surgery dates are adjusted so that their procedures are always scheduled on or after September 1st 2018. This modified constraint is an artifact of the fixed optimization time window and is tighter than what is realistically feasible. The first six months is treated as the warm up period for this reason.



(a) Daily ICU occupancy under offline and monthly optimization

(b) Change in wait time versus the status quo

Fig. 3 Schedule optimization eliminates days with 10 or more elective patients in the ICU given perfect information in LOS (left). Both offline and rolling monthly scheduling schedule the majority of the patients no later than their original surgery date (right)

Having established the performance upper bound under perfect information, we next proceed to consider incorporating machine learning predictions of LOS in schedule optimization, and the impact of inaccurate predictions on deterministic optimization models.

4 Predicting LOS with machine learning

Having established the performance upper bound of schedule optimization using accurate LOS predictions in the previous section, we now present the development of machine learning models for LOS prediction in this section, as well as the schedule optimization results using the predicted LOS.

We discuss both regression and classification models for LOS prediction. 5-fold cross validation is used for hyperparameter tuning and model selection during the training process. The model with the highest cross-validation score is selected for validation on the test set. The sklearn library [23] in Python was used to train and implement the selected models.

The predicted LOS is then used as input for RDO. Scheduling optimization is only performed on the test set and not on the training set. This simulates the real world setting where the incoming patients are prospectively scheduled based on LOS predictions at the time of admission. We compare the schedule optimization result with the status quo.

4.1 The dataset

Our data set consists of medical records, surgery data and CVICU census data for 2,352 elective cardiovascular surgical patients at the PAMC from 2014 to 2020. It is split into a training set with 1,738 patients from January 2014 to mid-June 2018, and a test set with 614 surgical patients from mid-June 2018 to May 2020.

The patient features are chosen based on clinical knowledge of the experienced surgeons at the unit, as detailed below.

Categorical Features:

- Surgeon: The assigned surgeon for the surgery
- Procedure: The type of surgical procedure
- Month: The month when the surgery is performed
- Genetic Disorder: The type of genetic disorder the patient has prior to the surgery, if any
- Ventilator Status: whether the patient is on a ventilator prior to the surgery
- Respiratory Status: the type of respiratory diseases the patient has, if any, prior to the surgery

Continuous Features:

- Age: The age of the patient at the time of the surgery
- Weight: The weight of the patient at the time of surgery
- Height: The height of the patient at the time of surgery

The missing values in the data are imputed using the mean for continuous features and the mode for categorical features from the training set.

To compare our data quality with the existing literature, we first train a binary classification model to identify patients with risks of prolonged stays of more than 5 or 10 days, making up 28% and 12.8% of the training set respectively. Candidate machine learning models considered include logistic regression, random forest and gradient boosting machine. Gradient boosting machine is selected based on cross-validation AUC.

We refer to Ettema et al. (2010) [8] for performance benchmarks. The authors of [8] reviewed and validated 14 models for the prediction of prolonged LOS in the ICU after cardiac surgery using data from 11,395 surgeries, and found that the area under the curve (AUC) scores of the best performing models range from 0.71 to 0.75. Our model achieves an AUC score of 0.77 for stays more than 5 days and 0.73 for stays more than 10 days on the test set, which is inline with that of the best-performing models identified in [8] (See Table 1).

Table 1 Model performance in binary classification vs benchmarks

Model	AUC Score
>5 days	0.77
>10 days	0.73
Benchmarks	0.71-0.75[8]

Although a direct comparison with existing literature is difficult due to different patient populations and different definitions of prolonged stay, our model and features can achieve high levels of distinguishing performance compared to the benchmarks.

4.2 Model development

We now train machine learning models for LOS with the objective of using the predictions for surgical scheduling.

We first consider classification models that classify LOS into ordinal buckets: 0-1 day, 2-5 days, 6-10 days and $>$ 10 days. For each LOS bucket, we define two metrics to evaluate model performance. 'Accuracy 1' is calculated as the exact number of true predictions in the bucket divided by the actual number of patients in the bucket. 'Accuracy 2' extends the definition of true prediction to include the scenario where the patient is predicted to be in the LOS bucket adjacent to his or her true bucket.

Candidate models include gradient boosting machine (GBM), random forest and ordinal regression models. GBM achieves the best cross-validated accuracy score on LOS classification with multiple LOS buckets. The model achieves an overall accuracy of 53%, and 88% of the predicted LOS either fall in the true bucket or the adjacent buckets. The most significant predictors according to the impurity-based feature importance are procedure types (0.47), weight (0.22), height (0.17) and the surgeon (0.07).

A breakdown of the predictive accuracy on each buckets of the test set is presented in Table 2. While predictive accuracy for the first three patient groups with LOS $\leq$ 10 days is relatively high based on Accuracy 2, predictive accuracy drops sharply for the group of patients with longer than 10 days of LOS, where the majority of the patients' LOS is being under-estimated. Similar behavior is observed under other choices of bucketing, such as using finer buckets for the LOS$>$ 10 patient group.

We use the predicted buckets of LOS are as inputs for RDO in Algorithm 1. Given each patient's predicted LOS bucket, the upper bound of the first three buckets is used as the LOS point estimates l_p, and $l_p = 30$ is used for the $>$ 10 bucket because 95% of the patients left the CVICU in less than or equal to 30 days. Although the machine learning model uses month of surgery from historical data, this feature was studied and had little effect on LOS prediction. Optimization performance on ICU occupancy and wait times is shown in red in Fig. 4, in comparison with performance upper bounds in grey assuming the classification algorithm achieves 100% accuracy. Here, we consider a range of weights on the cost of ICU overflow relative to the cost of total wait times: $\beta \in \{200, 100, 50, 25, 10, 5\}$. With perfect predictions of each patient's LOS bucket (grey bars), the optimization model significantly reduces the number of high ICU occupancy days without pushing back surgeries when $\beta = 10$ and 5. In contrast, the machine learning predicted outcomes show only slight improvement in both objectives for $\beta = 50$ (red bars). The trade off between the two objectives becomes significant as β decreases. When the weight on the cost of ICU overflow is sufficiently high, optimization reduces the number of high ICU occupancy days at the cost of increasing patient wait times and vice versa.

Table 2 Model performance (GBM) on LOS classification using the coarse LOS buckets

Bucket (% of Test Set)	Accuracy 1	Accuracy 2
0-1 days (26%)	0.33	0.96
2-5 days (41%)	0.89	0.97
6-10 days (15%)	0.11	1.00
$>$ 10 days (18%)	0.33	0.43
Overall	0.53	0.88

Compared to classification models, generating point-forecast for individual patients using regression models is even more challenging especially for those with prolonged LOS (see, for example, [30]). We follow the same procedure to select and train a variety of regression models including OLS regression, Lasso regression, gradient boosting machine, quantile regression and more. However, when the generated point-predictions are combined with deterministic optimization, there is still no model that achieves performance improvement on both metrics of interest. Similar to the classification model above, the main difficulty comes from significant under-estimation of prolonged LOS. Figure 5 plots the density distribution of relative predictive errors, defined as the ratio between true LOS and predicted LOS, for training and testing set. The distribution is heavily right-skewed with a long-tail of very large relative errors (i.e. true LOS $>>$ predicted LOS). We explore this issue further in the following section.

4.3 Identifying the challenges

Results in Section 4.2 show that the major challenge in LOS prediction and achieving measurable improvements through optimization is predicting prolonged LOS at the time of admission. This is consistent with what many past studies have observed. For example, neither linear regression or arti-

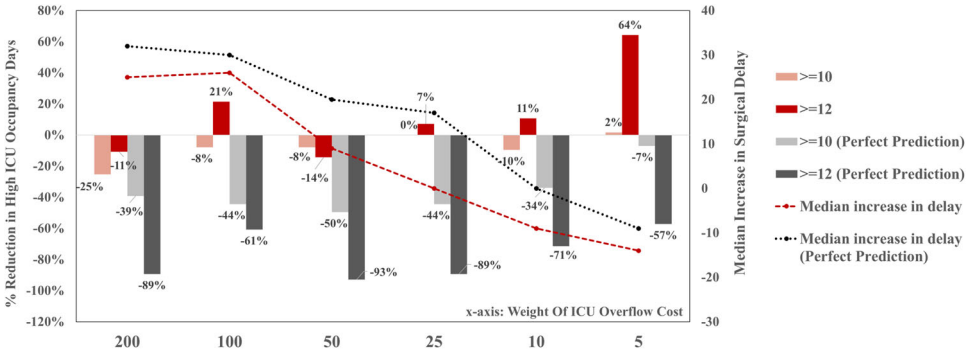


Fig. 4 RDO performance comparison using 100% accurate classification predictions (grey) and actual machine learning predictions (red)

ficial neural network developed in [30] is able to predict patient LOS of greater than 15 days. The neural network proposed in [17] is unable to predict patient LOS of greater than 150 hours on the validation set. Yang et al., [34] concluded that the predictive accuracy of most existing prediction techniques is expected to be inferior in the tail of the distribution to that in the middle due to imbalanced data. Furthermore, [16] shows that after 5 days in the ICU, the most significant predictors of the remaining LOS are features collected on day 5 instead of those at the time of admission.

Poor predictions of prolonged LOS have a significant impact on optimization performance, especially in the context of cardiovascular surgeries where LOS in the ICU tends to have a long tail distribution. As shown in Fig. 6b, 5.7% of the patients with LOS of greater than 30 days in the ICU makes up 42% of the total CVICU daily census count in the test set aggregated from 2018 to 2020. This small group of patients thus has a disproportionately large influence on ICU resource use and operational performance.

A closer examination of our data also reveals the non-stationary nature of hospital operations, another challenge to developing and evaluating predictive models in real life. A comparison between Figs. 6b and 6a highlights the shift in patient population in terms of LOS distributions from 2014-2018 to 2018-2020 at the PAMC. The most notable change

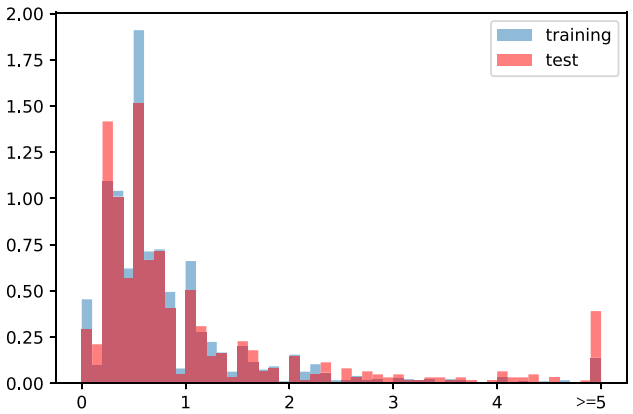


Fig. 5 Relative predictive errors of the GBM regression model, $\frac{\text{true LOS}}{\text{predicted LOS}}$. The distribution is right-skewed

is an increase in cases with extended LOS in the CVICU, driven by the hospital's decision to admit a larger fraction of complex cases in recent years. Such shifts in hospital operations make prospective predictions of LOS based on past data even less reliable. We split the training and test set in chronological order instead of randomly to reflect the impact of such inherent non-stationarity on model development and performance in practice.

5 Data-driven surgical scheduling with machine learning under uncertain LOS

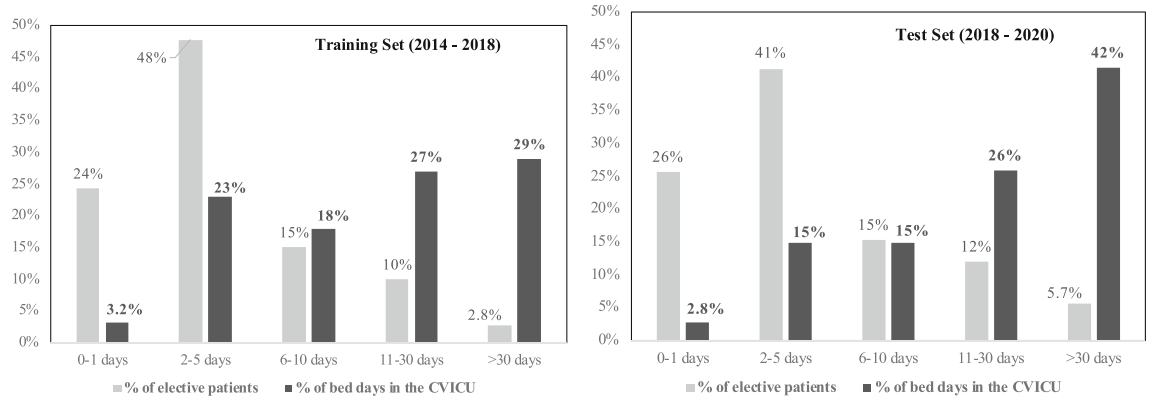
While combining machine learning with a deterministic formulation of optimization can provide significant performance improvement in theory, we have shown that the unpredictability of the long-tailed LOS along with other challenges can lead to poor performance in practice. A natural next step is to consider optimization methods that directly incorporate the uncertainties in LOS.

In this section, we propose a data-driven surgical scheduling framework, designed to address the identified challenges in the previous sections. Our framework combines optimization under uncertainty with machine-learning predicted distribution of LOS, and rolling information update.

Under this general framework, we develop three algorithms: Standard-RSO, Conservative-RSO and RRO. Standard-RSO and Conservative-RSO apply stochastic optimization using the predicted LOS distributions to minimize the expected cost, where the latter further adjusts the predicted LOS distribution to target under-estimations for prolonged LOS. RRO uses robust optimization to produce solutions that are robust against under-estimations for prolonged LOS.

5.1 Estimating LOS probability distribution using machine learning

Instead of relying on deterministic predictions of LOS for optimization, we now use machine learning models to predict



(a) Training set

(b) Test set

Fig. 6 The fraction of patients with extended LOS (> 30 days) doubled in the test set (right) compared to the train set (left). 5.7% of the patients with LOS of > 30 days make up 42% of the CVICU census count from 2018 to 2020. Training set is from January 2014 to mid-June 2018. Test set is from mid-June 2018 to May 2020

the probability distribution of LOS. We start by generating point-wise prediction of LOS for each patient using conventional machine learning models discussed in Section 4. For each patient p, let v_p be the value of LOS point prediction from the machine learning model for the true LOS, L_p, and define the relative prediction error

$$R_p = L_p / v_p. \tag{4}$$

L_p and R_p are random variables and v_p is known. We assume that the random variables R_p's are independent and identically distributed across patients from a distributions $\mathcal{S}$, i.e.,

$$R_p \overset{\text{iid}}{\sim} \mathcal{S} \quad \forall p \in P.$$

Note that $R_p > 1$ represents an under-prediction of LOS and $R_p < 1$ means an over-prediction. We choose to model *relative* prediction errors so that the absolute prediction errors, $|L_p - v_p|$, tend to be larger for patients with prolonged realized LOS. This is in accordance with our finding in Section 4 that prediction is more challenging for prolonged LOS.

The distribution $\mathcal{S}$ is approximated using the empirical distribution of predictive errors of the machine learning model during training and validation. Figure 5 provides an example of this distribution.

Given estimated $\mathcal{S}$, the distribution of L_p for individual patient p is estimated using

$$L_p = round(v_p \cdot R_p) \quad R_p \sim \hat{\mathcal{S}}.$$

5.2 Rolling stochastic optimization (RSO)

Building on the rolling deterministic optimization framework introduced in Section 3.2, we first explore stochastic optimization approaches that utilize estimated probability distributions of patient LOS.

Let the random variable L_p denote the LOS of each patient p, and a *LOS realization trace*,

$$\omega = \{l_p : p \in P_b \cup P_b^{past}\},$$

denote a sequence of realized LOS for all patients who have arrived at the system at the start of period b. We use μ_b to denote the discrete probability distribution of ω over the set of all possible traces for batch b, denoted as Ω_b. We assume that each patient's LOS is independent of other patients, and hence μ_b is the joint distribution of independent random variables L_p, for all $p \in P_b \cup P_b^{past}$.

Using the same set of notations as deterministic BOP introduced in Section 3.2, stochastic optimization formulations aim to minimize the expected cost taken over the distribution μ_b:

$$\min_x \sum_{p \in P_b} \sum_{d=s_b}^{N_d} (d - d_p^{min})^+ x_{d,p} + \beta \cdot \mathbb{E}_{\omega \sim \mu_b} \sum_{d=s_b}^{N_d} f(u_d(x, \omega)) \tag{5}$$

Objective function 5 parallels the objective function 3a of the deterministic BOP. The distinction is that the ICU overflow variable, u_d is now a random variable that is a function of x and ω.

Following the approach used in [20], we approximate the objective function 5 using Sample Average Approximation (SAA). Specifically, we randomly sample N_Ω traces of ω from Ω_b, according to probability distribution μ_b. Each trace of ω is sampled by independently sampling l_p from the estimated distribution of L_p for each patient, $p \in P_b \cup P_b^{past}$.

Denote sampled traces as $\{\omega^{(n)} : n = 1, \ldots, N_\Omega\}$ and sampled LOS $\{l_p^{(n)} : n = 1, \ldots, N_\Omega, p \in P_b \cup P_b^{past}\}$. For each sampled trace, we introduce auxiliary binary variables $y_{d,p}^{(n)}$ that indicate if patient p needs an ICU bed on day d given $l_p^{(n)}$. Similarly, we use $u_d^{(n)}$ to denote ICU overflow on day d given $\omega^{(n)}$. The stochastic BOP with SAA is formulated below.

$$\min_x \sum_{p\in P_b} \sum_{d=s_b}^{N_d} (d - d_p^{min})^+ x_{d,p} + \frac{\beta}{N_\Omega} \sum_{n=1}^{N_\Omega} \sum_{d=s_b}^{N_d} f(u_d^{(n)}) \quad (6a)$$

$$x_{d,p} = \tilde{x}_{d,p} \quad \forall p \in P_b^{past}, d \in D \quad (6b)$$

$$\sum_{d=s_b}^{N_d} x_{d,p} = 1, \quad \sum_{d=1}^{s_b-1} x_{d,p} = 0 \quad \forall p \in P_b \quad (6c)$$

$$y_{d,p}^{(n)} = \sum_{d'=\max(d-l_p^{(n)}+1,1)}^{d} x_{d',p} \quad \forall p \in P_b \cup P_b^{past}, \forall d \in D, n = 1, \ldots, N_\Omega \quad (6d)$$

$$\sum_{p\in P_b\cup P_b^{past}} y_{d,p}^{(n)} \le c + u_d^{(n)} \quad \forall d \in D, n = 1, \ldots, N_\Omega \quad (6e)$$

$$x \in Q^{op} \quad (6f)$$

$$y_{d,p}^{(n)}, x_{d,p} \in \{0, 1\}, \quad u_d \ge 0 \quad (6g)$$

The objective function 6a formulates the SAA approximation of the objective function 5. The constraints 6d and 6e formulate the stochastic parallel to constraints 3d and 3e in deterministic BOP, creating a replica of variables $y_{d,p}$ and $u_{d,p}$ for every sample trace $\omega^{(n)}$. The other constraints remain identical to their deterministic counterpart. Note that we focus on surgical scheduling problems where different realizations of ω only affect daily ICU occupancy and overflow, and do not affect feasibility of a schedule, x. In practice, ICU overflow can be accommodated by setting up temporary ICU beds or utilizing spare resources from other ICUs. Feasibility is thus determined exclusively by operational constraints such as OR room capacity, surgeon and patient availability, which remain unchanged.

We combine stochastic BOP with the rolling optimization framework introduced previously for RDO. Under this framework, stochastic BOP is solved sequentially for a sequence of schedule days, $\{s_b : b = 1, \ldots, B\}$. Meanwhile, the probability distributions of LOS L_p are progressively updated for every patients as uncertainty realizes. We introduce the information update procedure for stochastic BOP below.

Information Update Procedure for Stochastic BOP. At the start of each batch b, the distribution of L_p for all $p \in P_b^{past}$ is updated as follows.

(a) If the patient has undergone surgery and has been discharged from ICU by scheduling day s_b, her realized LOS is observed and we update L_p to be a constant equal to the true LOS.
(b) If the patient is in the ICU on day s_b having stayed for m days, then we update the distribution of L_p using the conditional distribution of $L_p | L_p \ge m$. To obtain the resultant distribution, we update the distribution of prediction errors, $R_p \sim \mathcal{S}$ by conditioning on $round(v_p \cdot R_p) \ge m$ while fixing v_p.
(c) If the procedure of the patient is scheduled after s_b, there is no change to the distribution of L_p.

We develop two rolling stochastic optimization algorithms, Standard-RSO and Conservative-RSO. Both algorithms adopt the same framework combining stochastic optimization, machine-learning predicted distribution and rolling information update.

The two algorithms differ from each other on how LOS realization traces ω are sampled for SAA. In Standard-RSO, realizations of $l_p^{(n)}$ are generated by sampling the relative prediction error $r_p^{(n)}$ directly from $\mathcal{S}$ and setting

$$l_p^{(n)} = round(v_p \cdot r_p^{(n)}).$$

In contrast, in Conservative-RSO, we generate *conservative* realizations of $l_p^{(n)}$ by rounding any $r_p^{(n)} < 1$ drawn from $\mathcal{S}$ to one, i.e.,

$$l_p^{(n)} = round(v_p \cdot \max\{r_p^{(n)}, 1\}).$$

In other words, Standard-RSO draws LOS realizations from a distribution derived from two-way predictive errors, $\mathcal{S}$; Conservative-RSO draws LOS realizations from an adjusted distribution of $\mathcal{S}$, where only one-way errors from LOS *under-predictions* (i.e. $r_p > 1$) are retained, and any over-prediction errors $r_p < 1$ are rounded up to 1. Conservative-RSO is designed such that the algorithm focuses on stochasticity arising from difficult-to-predict extended LOS for cardiovascular surgeries, where machine learning models consistently yield under-predictions.

We formalize the two rolling stochastic optimization algorithms in Algorithms 2 and 3 below.

When implementing Standard-RSO and Conservative-RSO, we set $N_\Omega = 10$ to obtain our numerical results. Large values of N_Ω can lead to very large numbers of variables and

Algorithm 2 Standard-RSO

Initialize with $b = 1$, $P_b^{past} = \emptyset$, $\tilde{x} = \emptyset$.
for $b = 1$ **to** B **do**
  1. Information Update. Patient arrival P_b is realized between s_{b-1} and $s_b - 1$; obtain point-prediction of LOS, $\{v_p : p \in P_b\}$.
  for $p \in P_b^{past}$ **do**
    Update LOS distribution L_p and $\mathcal{S}$ using the *Information Update Procedure* for stochastic BOP.
  end for
  2. SAA Sampling
  for $n = 1, 2, \ldots, N_\Omega$ **do**
    Sample the LOS realization trace, $\omega^{(n)} = \{l_p^{(n)} : p \in P_b \cup P_b^{past}\}$ by randomly sampling relative prediction error $r_p^{(n)}$ from $\mathcal{S}$ and setting $l_p^{(n)} = round(r_p^{(n)} \cdot v_p)$ for all p. In the case where $L_p = l_p$ is a constant, set $l_p^{(n)} = l_p$.
  end for
  3. Schedule Optimization. Solve stochastic BOP with SAA given s_b, P_b, P_b^{past}, $\tilde{x}$, $\{\omega^{(n)} : n = 1, 2, \ldots, N_\Omega\}$; implement the optimal solution x^* for all $p \in P_b$.
  4. Schedule and Capacity Update.
  $P_{b+1}^{past} \leftarrow P_b^{past} \cup P_b$
  $\tilde{x} \leftarrow \tilde{x} \cup x^*$
end for

Algorithm 3 Conservative-RSO

Initialize with $b = 1$, $P_b^{past} = \emptyset$, $\tilde{x} = \emptyset$.
for $b = 1$ **to** B **do**
  1. Information Update. Patient arrival P_b is realized between s_{b-1} and $s_b - 1$; obtain point-prediction of LOS, $\{v_p : p \in P_b\}$.
  for $p \in P_b^{past}$ **do**
    Update LOS distribution L_p and $\mathcal{S}$ using the *Information Update Procedure* for stochastic BOP.
  end for
  2. Conservative SAA Sampling
  for $n = 1, 2, \ldots, N_\Omega$ **do**
    Sample the LOS realization trace, $\omega^{(n)} = \{l_p^{(n)} : p \in P_b \cup P_b^{past}\}$ by randomly sampling relative prediction error $r_p^{(n)}$ from $\mathcal{S}$ and setting $l_p^{(n)} = round(v_p \cdot \max\{r_p^{(n)}, 1\})$ for all p. In the case where $L_p = l_p$ is a constant, set $l_p^{(n)} = l_p$.
  end for
  3. Schedule Optimization. Solve stochastic BOP with SAA given s_b, P_b, P_b^{past}, $\tilde{x}$, $\{\omega^{(n)} : n = 1, 2, \ldots, N_\Omega\}$; implement the optimal solution x^* for all $p \in P_b$.
  4. Schedule and Capacity Update.
  $P_{b+1}^{past} \leftarrow P_b^{past} \cup P_b$
  $\tilde{x} \leftarrow \tilde{x} \cup x^*$
end for

constraints under constraints 6d and 6e. We thus limit the value of N_Ω due to computational constraints. We discuss the computational limitations further in Section 7.

5.3 Rolling robust optimization (RRO)

Robust optimization formulations provide an alternative to stochastic optimization for scheduling under uncertainty. Robust optimization aims to minimize the *worst-case* cost defined over an uncertainty set, $\mathcal{U}_b$, which is a chosen subset of all possible realized traces $\omega = \{l_p : p \in P_b \cup P_b^{past}\}$. Using the same notations as before, the formulation of robust BOP for a pre-defined uncertainty set $\mathcal{U}_b$ can be written as follows.

$$\min_x \quad \sum_{p \in P_b} \sum_{d=s_b}^{N_d} (d - d_p^{min})^+ x_{d,p} + \beta \cdot \theta(x) \tag{7a}$$

s.t.

$$x_{d,p} = \tilde{x}_{d,p} \quad \forall p \in P_b^{past}, d \in D \tag{7b}$$

$$\sum_{d=s_b}^{N_d} x_{d,p} = 1, \quad \sum_{d=1}^{s_b-1} x_{d,p} = 0 \quad \forall p \in P_b \tag{7c}$$

$$x \in Q^{op}, x_{d,p} \in \{0, 1\} \tag{7d}$$

where $\theta(x)$ is the worst-case cost associated with ICU overflow for all possible LOS realization traces in the uncertainty set,

$$\theta(x) = \max_{\{l_p : p \in P_b \cup P_b^{past}\} \in \mathcal{U}_b} \min_{y,u} \sum_{d=s_b}^{N_d} f(u_d) \tag{7e}$$

s.t.

$$y_{d,p} = \sum_{d'=\max(d-l_p+1,1)}^{d} x_{d',p} \ \forall p \in P_b \cup P_b^{past}, d \in D \tag{7f}$$

$$\sum_{p \in P_b \cup P_b^{past}} y_{d,p} \le c + u_d \quad \forall d \in D \tag{7g}$$

$$y_{d,p} \in \{0, 1\}, u_d \ge 0. \tag{7h}$$

Apart from the uncertainty set $\mathcal{U}_b$, the remaining constraints of robust BOP are similar to those of deterministic BOP in optimization problem 3. It is worth noting that in Eq. 7f, $x_{d,p}$ are now constant model parameters passed on from the outer minimization problem, and l_p are now decision variables to be optimized.

To formulate the uncertainty set $\mathcal{U}_b$ for scheduling day s_b, let l_p^{min} and l_p^{max} denote the lower and upper bound of LOS for patient p. and A_b the subset of patients where $l_p^{max} - l_p^{min} > 0$, i.e., $A_b = \{p \in P_b \cup P_b^{past} : l_p^{max} - l_p^{min} > 0\}$. The uncertainty set, $\mathcal{U}_b$ is defined to be the set of possible realizations of $\omega \in \Omega_b$ that satisfy the following two constraints:

$$l_p^{min} \le l_p \le l_p^{max} \quad \forall p \in P_b \cup P_b^{past}, \tag{8}$$

and, for some chosen constant $0 \leq \eta \leq 1$,

$$\sum_{p \in A_b} \left[\frac{l_p - l_p^{min}}{l_p^{max} - l_p^{min}} \right] \leq \eta \cdot |A_b|. \tag{9}$$

Constraint 9 enforces a *budget of uncertainty*, $\eta \cdot |A_b|$, which limits the total possible normalized deviations (i.e., extended ICU days) from the lower bound l_p^{min}. Following the terminology in robust optimization literature (see, e.g., [4]), we refer to η as the *uncertainty budget*, and tune the value of η among $\{0.5, 0.75, 1.0\}$ based on performance of numerical experiments.

Instead of using stylized assumptions to set values for l_p^{min} and l_p^{max} and construct the uncertainty set (see, e.g., [22]), we tailor the values of l_p^{min} and l_p^{max} for individual patients using machine-learning predicted LOS distributions. For all incoming patients, $p \in P_b$, we set l_p^{min} to be the point-prediction given by the machine learning model, v_p. The LOS upper-bound l_p^{max}, on the other hand, is determined using the $100(1-\alpha)$ percentile of relative predicted error under the estimated distribution, $\mathcal{S}$:

$$l_p^{max} = round(v_p \cdot r_{max}), \quad r_{max} = F_{\mathcal{S}}^{-1}(1-\alpha),$$

where F_S^{-1} denotes the inverse cumulative distribution function of $\mathcal{S}$. For our numerical experiments, we present results with $\alpha = 0.2$ (i.e., the 80th percentile), chosen among $\{0.1, 0.15, 0.2, 0.25\}$ based on simulated robust optimization performance for our data set.

As uncertainties of LOS realize overtime for previously scheduled patients, P_b^{past}, we update the distributions of LOS and parameters used for the uncertainty set dynamically as follows.

Information Update Procedure for Robust BOP. At the start of each batch b, update the distributions of L_p for all $p \in P_b^{past}$ using the update procedure for stochastic BOP introduced in Section 5. Given the updated distributions, l_p^{min}, l_p^{max} are updated as follows.

(a) If the patient has undergone surgery and has been discharged from ICU by scheduling day s_b, update $l_p^{min} = l_p^{max}$ to be equal to the true LOS.
(b) If the patient is in the ICU on day s_b having stayed for m days, update $l_p^{min} = \max\{m, v_p\}$, and $l_p^{max} = round(v_p \cdot r_{max})$, where $r_{max} = F_{\mathcal{S}}^{-1}(1-\alpha)$ using the updated $\mathcal{S}$.
(c) If the procedure of the patient is scheduled after s_b, there is no change to l_p^{min}, l_p^{max}.

We develop an adapted column and constraint generation approach (AC&CG) to solve robust BOP in optimization problem 7 with the above definition of $\mathcal{U}_b$. The method of AC&CG for surgical scheduling with ICU capacity constraints was first proposed in [22], which was designed for linear cost functions $f(u_d)$ only. Our AC&CG approach extends the algorithm in [22] to accommodate convex, piecewise linear formulations of $f(u_d)$.

At a high level, the AC&CG approach involves iteratively solving a *Main problem* and a *Recourse problem*. Every iteration t, the Main problem solves for a *temporary* optimal scheduling decision, x^{t*}, that minimizes the worst-case cost over a *subset* of the uncertainty set, $\Omega^t \subset \mathcal{U}_b$. The optimal objective value provides a lower bound to that of the original BOP. The Recourse problem then finds a worst-case LOS realization trace, $\omega^{t*} \in \mathcal{U}_b$, that maximizes the cost of ICU overflow under the temporary scheduling decision, x^{t*}. The optimal objective value provides an upper bound to that of the original BOP. When the upper and lower bounds obtained are equal, the temporary scheduling decision x^{t*} obtained from the Main problem is also optimal to robust BOP, and the algorithm terminates. Otherwise, we update the set Ω^t to include ω^{t*} obtained from the Recourse problem, $\Omega^{t+1} \leftarrow \Omega^t \cup \{\omega^{t*}\}$, and re-solve the updated the Main problem with Ω^{t+1}.

We formalize the above AC&CG algorithm for solving robust BOP in Algorithm 5 in Appendix B. Note that each iteration of AC&CG adds a LOS realization trace, ω^{t*} to the set Ω^t. This process adds a significant number of new variables and related constraints to the Main problem.[5] As the number of iterations increases, the size of the Main problem can therefore grow quickly. When implementing AC&CG, we thus terminate the algorithm after at most 10 iterations due to computational constraints. In Appendix C, we show that the optimally gaps, $\frac{UB-LB}{LB}$, are reduced to under 1% for almost all batches after 10 iterations. We discuss the computational challenges further in Section 7.

The full rolling robust optimization algorithm (RRO) combining information update and AC&CG is presented in Algorithm 4.

6 Numerical experiments and results

In this section, we evaluate the three algorithms introduced in Section 5 based on numerical simulations with real-world data. Using the dataset and procedures described in Section 4, we develop a regression model for LOS prediction using GBM on historical training data from 2014 to 2018. We use the machine learning model to generate LOS predictions for all patients in the test set, $\{v_1, v_2, \ldots, v_{|P|}\}$. The empirical distribution of predicative errors on both the training set and

[5] See constraints 23d and 23e in Appendix B.

Algorithm 4 Rolling Robust Optimization (RRO)

1: Initialize with $b = 1$, $P_b^{past} = \emptyset$, $\tilde{x} = \emptyset$.
2: **for** $b = 1$ **to** B **do**
3: **1. Information Update.** Patient arrival
4: P_b is realized between s_{b-1} and $s_b - 1$; evaluate l_p^{min}, l_p^{max} for all $p \in P_b$.
5: **for** $p \in P_b^{past}$ **do**
6: Update LOS distributions and l_p^{min}, l_p^{max} using the *Information Update Procedure* for robust BOP.
7: **end for**
8: **2. Schedule Optimization.** Solve robust BOP with AC&CG given s_b, P_b, P_b^{past}, $\tilde{x}$, l_p^{min}, l_p^{max} $\forall P \in P_b \cup P_b^{past}$; implement solution x^* for all $p \in P_b$.
9: **3. Schedule and Capacity Update.**
10: $P_{b+1}^{past} \leftarrow P_b^{past} \cup P_b$
11: $\tilde{x} \leftarrow \tilde{x} \cup x^*$
12: **end for**

the test set[6], are used as the estimated distribution $\mathcal{S}$. The LOS predictions and $\mathcal{S}$ are then used as inputs to Algorithms 2, 3 and 4 to generate optimal surgical schedules. Performances of the optimal schedules are simulated and evaluated using historical patient arrivals and actual LOS data in the test set from September 2018 to March 2020.

Our numerical results indicate that both Conservative-RSO and RRO outperform the status quo, while Standard-SP fails to achieve consistent performance improvement. Moreover, Conservative-RSO achieves the best overall performance despite its relative computational simplicity compared to RRO. The results highlight the importance of tailoring algorithm for long-tailed distributions and reveals shortcomings of complex algorithms in practical settings. The best-performing algorithm, Conservative-RSO, provides a promising direction for designing efficient CVICU scheduling algorithms.

6.1 Performance on historical true LOS

We first simulate the performance of Standard RSO, Conservative-RSO and RRO on historical patient arrivals.

Each run of either stochastic optimization algorithms may produce a different scheduling policy, because the objective function involves random sampling. For this reason, both Standard-RSO and Conservative-RSO are run 90 times each on the testing period for each values of $\beta \in \{1, 5, 10, 20\}$, where greater values of β mean more weight on the cost of ICU overflow compared to wait time.

In contrast, the solution of RRO is deterministic for the same set of model parameters. When evaluating RRO, we thus only run the algorithm once for each set of parameters. For RRO, we tune both values of $\beta \in \{1, 5, 10, 20\}$ and the uncertainty budget, $\eta \in \{0.5, 0.75, 1.0\}$. Greater values of η mean a larger uncertainty set containing longer LOS realizations.

Performance of each optimal policy is simulated using the historical true LOS trace, i.e., the actual realized values of LOS for each patient in the testing period. The change in wait times and the number of high ICU occupancy days in the ICU for each experiments are evaluated in comparison to those of the original schedule (i.e. the status quo). Good performance corresponds to an improvement on both metrics.

Performance of the Standard-RSO, Conservative-RSO and RRO are presented in Figs. 8, 7, and 9 respectively. Each run of experiments is plotted using the average or median change in wait time (the y coordinate) and the simulated reduction in high ICU occupancy days in the ICU (the x coordinate). The quadrant shaded in blue in each plot indicates performance improvement on both metrics. Different values of β indicated by color, and different values of η (for RRO only) are indicated by marker type.

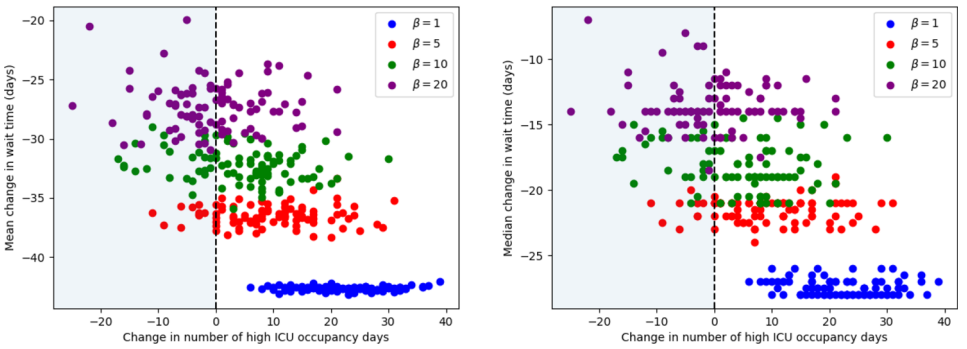
Standard-RSO shows poor performance in reducing ICU overflow regardless of the value of β,[7] as illustrated in Figs. 7a and 7b. Despite considering the stochasticity in LOS, Standard-RSO tends to aggressively schedule most patients much earlier and is unable to effectively handle the trade off between ICU overflow and patient wait times.

In contrast, Conservative-RSO demonstrates consistent improvement of performance in both metrics compared to the status quo in Fig. 8a in terms of the average change in wait time and the reduction in high ICU occupancy days for $\beta \in \{5, 10\}$. Relatively weaker performance is observed when the median change in wait times is examined in Fig. 8b. This is likely because the formulation of the objective function specifically minimizes the sum of wait times instead of the median. As expected, higher values of β lead to less high ICU occupancy days but longer wait times.

RRO also achieves good performance shown in Fig. 9. In particular, performance improvement on both metrics parameter are achieved under parameter combinations ($\eta = 1, \beta = 10$) and ($\eta = 0.75, \beta \in \{5, 10, 20\}$). Higher values of β and uncertainty budget η tend to result in less high ICU occupancy days but longer wait times, in line with our expectation.

[6] In practice, prediction errors of the test set will not be available at the time of scheduling. The test errors are included in constructing $\mathcal{S}$ so that, if the LOS of an incoming patient in the test set surpasses all cases in the training set, conditional sampling described in the previous section (see Step 2(b) of Method 1 still works as intended. Although this is not ideal, the scheduling process should not benefit much from it and it helps simplify our simulations. In practice, if the LOS of a patient surpasses all previous cases, one approach is to consult health providers.

[7] For consistency, we present performance for $\beta \in \{1, 5, 10, 20\}$ for all algorithms. It is worth noting that increasing the value of β for Standard-RSO does not lead to meaningful improvement in performance. One example is provided in Fig. 13 of Appendix C.



(a) Mean change in wait times vs. ICU congestion (b) Median change in wait times vs. ICU congestion

Fig. 7 Performance trade-off of Standard-RSO between patient wait times and ICU congestion using different values of β, compared to the status quo. ICU congestion is measured using the number of high ICU occupancy days with at least 10 elective patients in the ICU ($u_d \geq 2$)

Although Conservative-RSO and RRO are more conservative in its ICU occupancy forecasts and lead to longer patient wait times, there is enough slack in the original system so that both are able to effectively reduce ICU congestion without excessively pushing back surgeries compared to the original schedule. In addition, the contrast in performance between Standard-RSO and Conservative-RSO further reflects that, with careful design choices targeting the challenges specific to the problem, performance improvement can be achieved in practical settings without increasing complexity of the algorithm.

6.2 Performance on bootstrapped traces of LOS

In order to obtain confidence intervals for the relative performances of the three methods, we now use bootstrapping [7] to generate multiple *evaluation traces* of LOS realization for all patient arrivals in the test data.

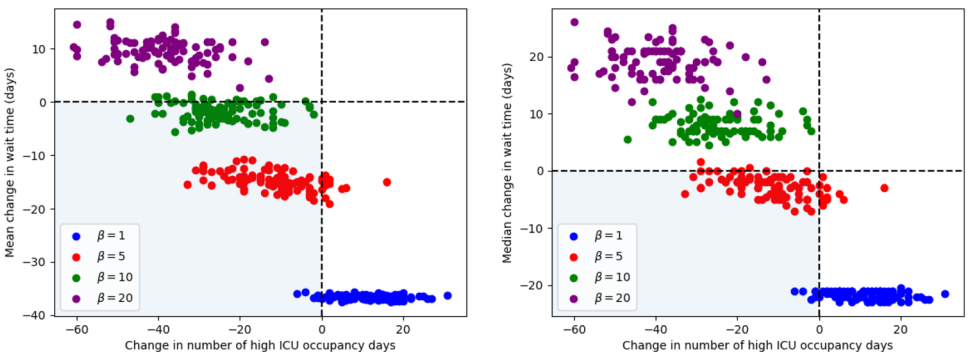
Let $\{v_1, v_2, \ldots, v_{|P|}\}$ be the historical sequence of predicted LOS values for all patients to be scheduled during the test period, $P = \bigcup_{b=1}^{B} P_b$. The procedure of evaluating each method on bootstrapped evaluation traces is as follows.

Performance evaluation with bootstrapping. Repeat the following steps to sample N_E evaluation traces $\omega'^{(i)} = \{l_p'^{(i)} : p \in P\}$ for $i = 1, 2, \ldots, N_E$:

- Step 1: Generate a bootstrapped patient arrival sequence. The historical arrival sequence is defined by a list of patient arrival times, $\{t_1, t_2, \ldots, t_{|P|}\}$ with predicted LOS $\{v_1, v_2, \ldots, v_{|P|}\}$. We uniformly sample with replacement to obtain a new sequence of arrival with predicted LOS $\{v_1'^{(i)}, v_2'^{(i)}, \ldots, v_{|P|}'^{(i)}\}$ while fixing the arrival times $\{t_1, t_2, \ldots, t_{|P|}\}$.
- Step 2. Generate predicted LOS distributions for the new arrival sequence using $\{v_1'^{(i)}, v_2'^{(i)}, \ldots, v_{|P|}'^{(i)}\}$ and $\mathcal{S}$.
  - Use the new arrival sequence and its predicted LOS distributions as common inputs for Standard and Conservative RSO, and RRO.
  - Use the same arrival sequence and set $l_p = v_p'^{(i)} \forall p$ as inputs for the deterministic formulation, RDO.

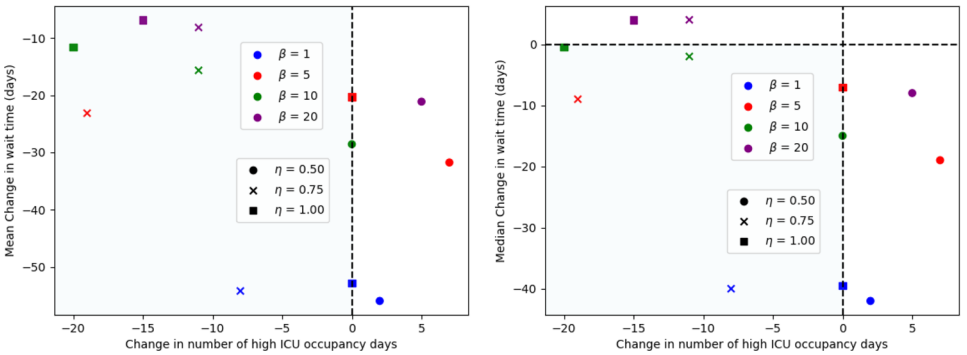
  Each algorithm produces one optimal schedule x^* for the sequence.
- Step 3: Generate a trace of realized LOS for the new arrival sequence by randomly sampling from the predicted LOS distributions. Specifically, for patient arrivals at $\{t_1, t_2, \ldots, t_{|P|}\}$, sample the sequence of prediction errors $\{r_1'^{(i)}, r_2'^{(i)}, \ldots, r_{|P|}'^{(i)}\}$ independently from $\mathcal{S}$ and obtain the trace of realized LOS,

$$\omega'^{(i)} = \{l_1'^{(i)}, l_2'^{(i)}, \ldots, l_{|P|}'^{(i)}\}$$



(a) Mean change in wait times vs. ICU congestion (b) Median change in wait times vs. ICU congestion

Fig. 8 Performance trade-off of Conservative-RSO between patient wait times and ICU congestion using different values of β, compared to the status quo. ICU congestion is measured using the number of high ICU occupancy days with at least 10 elective patients in the ICU ($u_d \geq 2$



(a) Mean change in wait times vs. ICU congestion (b) Median change in wait times vs. ICU congestion

Fig. 9 Performance trade-off of RRO between patient wait times and ICU congestion using different values of β, η compared to the status quo. ICU congestion is measured using the number of high ICU occupancy days with at least 10 elective patients in the ICU ($u_d \geq 2$)

by setting $l_p'^{(i)} = round(v_p'^{(i)} \cdot r_p'^{(i)})$ for all p.

- Step 4. Simulate the performance of the optimal schedule by each algorithm under the LOS realization scenario described by $\omega'^{(i)}$. Compare relative performance on both metrics. Performance of RDO is used as a benchmark for comparison.

Similar to Section 6.1, Step 1-4 are repeated for 100 iterations ($N_E = 100$) on each value of β for each algorithm. For RRO, $\eta = 0.75$ is used because of its good performance on the historical trace shown in Fig. 9. We compare the change in wait times relative to the original schedule, and compare the number of high ICU occupancy days of the three algorithms relative to that of deterministic optimization, RDO.

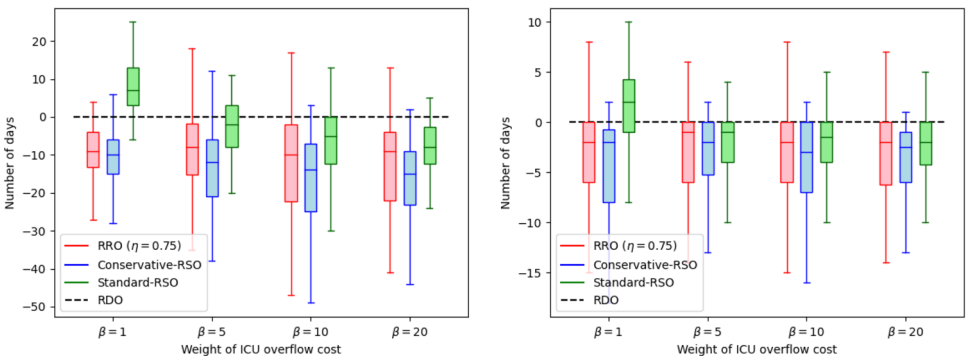
Figure 10 presents the performance of Standard-RSO, Conservative-RSO and RRO in reducing ICU congestion on bootstrapped patient arrivals in comparison to the performance benchmark set by RDO.

Standard-RSO manages to achieve some overflow reduction for $\beta \geq 10$. When β is small, however, Standard-RSO tends to under-perform compared to the deterministic benchmark. In contrast, the interquartile ranges of Conservative-RSO (blue) and RRO (red) both show more significant reduction in the number of high ICU occupancy days compared to RDO for all values of chosen β.

The *median* of changes in patient wait time compared to the status quo are presented in Fig. 11. Standard-RSO demonstrates similar behaviors to its deterministic counterpart, with the tendency to aggressively schedule patients earlier. In contrast, Conservative-RSO strategically delays procedures so that ICU overflow can be effectively avoided. For $\beta \in \{1, 5, 10\}$, the Conservative-RSO is able to consistently reduce ICU overflow without pushing back a majority of procedures compared to the original calendar. RRO, on the other hand, achieves comparable performance to Conservative-RSO and is able to reduce ICU overflow without pushing back procedures for all values of β.

Comparing Conservative-RSO and RRO, both algorithms utilize conservative LOS estimates to address under-predictions for long-tailed LOS. Both achieves promising reductions in ICU congestion without excessive increase in wait times. However, Conservative-RSO demonstrates better average and worst-case performance (upper caps of the box plots) in ICU overflow for all selected values of β.

The relative poorer performance of RRO in reducing ICU overflow may be a result of its sensitivity towards the choice of other parameters, such as the uncertainty budget η, the choice of l_p^{min} and the percentile $1-\alpha$ used for setting l_p^{max}. During our effort to calibrate the uncertainty set, the performance of RRO is observed to be drastically different under



(a) Change in high ICU occupancy days, $O_d \geq 10$ (b) Change in high ICU occupancy days, $O_d \geq 12$

Fig. 10 Difference in the number of high ICU occupancy days using RRO, Conservative-RSO and Standard-RSO relative to the deterministic benchmark by RDO, for different values of β. High ICU occupancy days in (a) are days with at least 10 elective patients in the ICU, and those in (b) are days with at least 12 elective patients. Each box plot shows the median, the interquartile range, minimum and maximum of 100 experiments

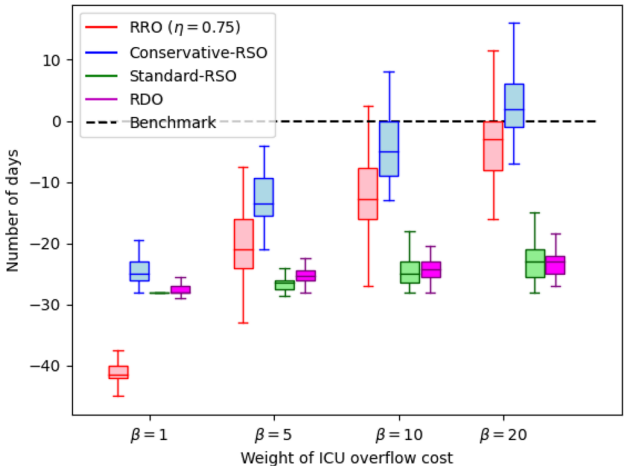


Fig. 11 Median change in wait times relative to the status-quo using RRO, Conservative-RSO, Standard-RSO and RDO respectively for different values of β

different parameter choices. While there remains room for potential performance improvement with better parameter tuning, finding the optimal combination of all parameters can be a very difficult task in practice given the large number of possibilities and the long computation time required for each run of the AC&CG algorithm. This reveals a shortcoming of using robust optimization approaches that involve a large number of parameters, and demonstrates that increasing algorithmic complexity does not always leads to better performance in practice. To improve practicality of robust optimization, further research is needed to develop new formulations less sensitive to parameter choices.

Lastly, we note that there is still a notable performance gap between our best-performing model, Conservative-RSO, and the theoretical upper bound established in Section 3 with rolling deterministic optimization under perfect information. This implies significant room for further development of more efficient data-driven optimization methods, with Conservative-RSO serving as a stepping tone towards a promising direction.

6.3 Summary of numerical results and additional approaches

We examined a variety of optimization formulations in conjunction with data-driven LOS predictions. All approaches are developed under similar frameworks combining machine learning, rolling information update and optimization methods, but they differ significantly in their simulation performances, strengths and weaknesses. We provide a summary of these algorithms in Table 3.

Numerous alternative formulations of each of the these approaches were examined:

- For stochastic optimization, estimating the distribution of true LOS L_p using the historical empirical distribution of LOS within the same procedure type, instead of sampling prediction errors.
- For robust optimization, using an uncertainty budget that scales with $\sqrt{|A_b|}$ instead of $|A_b|$ (see constraint 24f), and various values of $1 - \alpha$ for the LOS upper bound, including $\alpha \in \{0.1, 0.15, 0.2, 0.25, 0.3\}$.
- Various alternative optimization formulations, such as linear $f(u_d)$, penalizing under-utilization of OR blocks or minimizing maximum wait times, etc.
- Various alternative machine learning models. In particular, We used H2O AutoML [18] to search for other promising models from its wide range of built-in machine learning and deep learning algorithms.

None of the above alternatives result in consistent performance improvements compared to results presented in this section.

Table 3 Comparison of different algorithms

Algorithm	Performance	Challenges
RDO	Does not reduce ICU congestion without increasing patient wait times	Weak predictive accuracy for prolonged LOS
Standard-RSO	Does not reduce ICU congestion without increasing patient wait times	Insufficient sampling on rare occurrences of prolonged LOS
Conservative-RSO	Promising reductions in both ICU congestion and wait time	Can be overly conservative without precise LOS prediction
RRO	Promising reductions in both ICU congestion and wait time	Slow and difficult to tune

7 Insights and discussions

Among all the practical formulations considered, Conservative-RSO and RRO both achieved reduction in both ICU congestion and patient wait times. The key driver of their promising performance is the focus on under-estimates of long-tailed, extended LOS of cardiovascular surgery patients, through the design of either SAA sampling of LOS realizations, or the uncertainty set of LOS. Good performance also relies on precise choices of the trade-off coefficient β, and several other model parameters in the case of robust optimization. To achieve the desired optimization performance in practice, all model parameters need to be carefully tuned to balance reduced ICU congestion with longer wait times.

On each scheduling day, possible LOS outcomes for patients currently in the ICU are updated based on how long they have been in the ICU. Patients that have been in the ICU longer than expected have their projected probability distribution of LOS is updated accordingly. This information update procedure further corrects for potential under-estimates of prolonged LOS at the time of scheduling and improves optimization performance.

The fact that the optimization outcomes are highly sensitive to these nuanced design choices underscores the operational challenges of surgical scheduling with long-tailed LOS. Any solution will need to be tailored and fine-tuned to suit the context of specific institutions in order to reach the desired outcome.

Although our proposed scheduling framework has demonstrate promising performance, there remains room for potential extensions and future work to address some of its limitations.

First, under our rolling optimization framework, patients who arrived in-between two scheduling days need to wait until the next scheduling day to be scheduled. In our simulations, we have chosen the scheduling days s_b to be one month apart for elective procedures based on our context, as explained at the end of Section 3.2. However, for other types of procedures, scheduling may need to happen on a daily bases. While our modeling framework naturally extends to the case of daily scheduling (by setting s_b as consecutive days), managing ICU overflow in this case can become more difficult, since scheduling is performed with less patient arrival information. This is a limitation of our framework to be addressed in future works. For instance, authors of [15] propose an alternative rolling horizon framework for surgical scheduling that takes into account uncertain future patient arrivals. With a forward-looking, rolling arrival horizon, scheduling decisions can be made more frequently for realized patient arrivals, in anticipation of stochastic future arrivals. Incorporating designs of this kind with our current framework has the potential to improve practicality and performance for procedures that require faster scheduling.

Second, a challenge of both Conservative-RSO and RRO that has not been addressed is their computational limitations. In our numerical experiments, we have limited the number of sampled traces, N_Ω for SAA to 10 and the number of iterations of AC&CG to 11. As mentioned in Section 5.2 and 5.3, the memory and run time required to solve the optimal schedule for each batch of patients can grow quickly when N_Ω and the number of AC&CG iterations increase. In the case of SAA for Standard and Conservative-RSO, it typically took 4 to 8 hours to run each numerical experiment for $N_\Omega = 10$ with 19 batches with two CPUs and 16GB memory per CPU. For $N_\Omega = 30$, at least 21 hours were required to run one such numerical experiment with four CPUs and 16GB of memory per CPU. Meanwhile, AC&CG for RRO requires even longer run time: it took typically 7 to 12 hours to run each experiment for 11 iterations with two CPUs and 16GB memory per CPU. The authors of [22] also point out that the proposed algorithm may be computationally inefficient for larger uncertainty sets. Developing a better understanding of how the choice of N_Ω and the maximum number of AC&CG iterations impacts optimization performance is a meaningful subject of further research.

Finally, our models focus on uncertainties in LOS and ICU capacity, treating surgical duration and the corresponding OR capacity as deterministic while respecting the constraints at the institution studied (see Section 3.1). In environments where OR capacity is also a bottleneck or overtime is considered as part of the objective (see, e.g., [20, 22]), developing data-driven models that incorporate both uncertainty sources will be useful.

8 Concluding remarks

Using seven years of data from an academic medical center, we developed machine learning models to predict post-surgical length of stay, optimization models to schedule procedures to minimize post-surgical bed congestion, and simulated the results of the use of these models on the most recent 18 months of held-out empirical test data. We established an upper bound on performance with an offline optimization model of historical data with access to actual LOS.

A conservative stochastic program with sufficient sampling of the LOS distribution to capture tail behavior managed to achieve the best overall performance in reducing ICU

congestion without increasing wait times for surgery. Compared to the hospital's current paper-based system, the deterministic and standard stochastic optimization approaches, along with numerous variants of machine learning and optimization formulations, failed to improve performance. The failure of most models to improve over the status quo, illustrate the importance of using empirical data, rather than synthetic parametric data, especially for long-tailed distributions. Several lessons for dealing with such data are offered by the contrast between the models that did and did not improve performance.

The prevalent null results highlight the challenge of the unpredictability and non-stationarity of prolonged LOS in practical settings. This challenge is rarely addressed in previous works, most of which follow the common practice of using synthetic LOS data generated based on strong distributional assumptions. For example, in [20, 27], the distribution of the LOS in the ICU is assumed to have a mean and standard deviation of no more than 3.5 days. In the robust optimization approach discussed in [22], the worst-case deviation of the uncertainty set is assumed to be 4 days, which the authors admit can be far from reality. Despite the positive results obtained in simulation, the actual performance of these optimization models can be drastically different in practice in the presence of significant LOS estimation errors.

We proposed a novel formulation of stochastic optimization, Conservative-RSO, which demonstrated most consistent improvements with empirical data despite of its relative algorithmic simplicity. Our results show that Conservative-RSO is a promising first step towards addressing the unpredictability and long-tailed behavior of LOS to improve surgical scheduling, and they provide meaningful directions for future work. First, much work is needed to explore and evaluate alternative designs of data-driven optimization formulations, calibrated and evaluated using empirical data of LOS. For instance, one may want to explore alternative uncertainty sets for robust optimization, or ambiguity sets for distributionally robust optimization (see, e.g., [27]). When empirical data are not available, any distributional assumptions should incorporate the long-tailed, non-stationary nature of LOS distributions to minimize the bias present in using standard distributional assumptions. Furthermore, learning over long-tailed distributions has not been fully examined in prior research. We highlight its central importance in tackling the general problem of surgical scheduling with constrained downstream capacities. Obtaining accurate and precise LOS predictions is one way to push the Pareto frontier of optimization closer to the theoretical performance upper bound using full information. Settings where patient characteristics available at the time of scheduling fail to explain the majority of variation in LOS offer exciting challenges to be tackled by innovative approaches that combine the power of prediction, optimization, and information update.

Appendix A: Mathematical formulations for $f(u_d)$, Q^{op}

Solving the quadratic objective, $f(u_d) = u_d^2$ of the mixed-integer program can be slow. To reduce the runtime required to solve the problem, we implement the quadratic term in the objective function, $f(u_d) = u_d$, using a piece-wise linear approximation,

$$f(u_d) = e_1 u_d^{(1)} + e_2 u_d^{(2)} + e_3 u_d^{(3)} + e_4 u_d^{(4)} + e_5 u_d^{(5)}, \quad (10)$$

and add additional constraints

$$\sum_{p \in P_b \cup P_b^{past}} y_{d,p} \leq c + m - 1 + u_d^{(m)} \quad \forall d, m$$

$$u_d^{(m)} \geq 0 \quad \forall d, m$$

In words, $u_d^{(m)}$ counts ICU overflow above $c + m - 1$ for $m = 1, \ldots, 5$. Here, e_m are constant coefficients for the piecewise linear function.

In our formulation, we set $e_1 = 1, e_2 = e_3 = e_4 = e_5 = 2$. Since u_d only takes integer values, this coefficient choice leads to $f(u_d) \equiv u_d^2$ for any $u_d \leq 5$. The piece-wise linear approximation is 20 times faster than the quadratic form in our numerical experiments.

The above approximation is used in all our algorithms. The formulations for stochastic and robust algorithms are analogous (e.g. copies of $u_d^{(m)}$ are created for different LOS realizations), and we thus omit the details here.

Next, we introduce the full mathematical formulation of Q^{op} in constraint 1e for offline optimization below.

Sets

- $D = \{1, 2, \ldots, N_d\}$: index for days
- $P = \{1, 2, \ldots, N_p\}$: index for patients
- $P^{par} \subset P$: set of PAR patients
- K: set of surgeons

Parameters

- c: number of ICU beds reserved for elective patients, set to 8
- q_p: operation duration for $p \in P$.
- l_p: post-op length of stay in CVICU for $p \in P$
- $h_{d,k}$: number of hours surgeon k is available to perform surgery on day d, for $s \in S$. $h_{d,k} = 15$ if k operates on day d, 0 otherwise.
- $par\_day_{d,k}$: indicator variable for PAR days. $par\_day_{d,k} = 1$ if surgeon k can perform PAR surgeries on day d, 0 otherwise.

- $original\_date_p \in D$: the original date that a surgery is scheduled for for $p \in P$
- $actual\_date_p \in D$: the actual date that a surgery takes place for $p \in P$; can be different from $original\_date_p$ if the surgery was rescheduled.
- $arrived\_on_p$: the arrival date for $p \in P$, i.e., when a patient is first being scheduled for surgery; can be negative (i.e. outside the set D) if the surgery was scheduled before September 2018
- $lead_p$: the lead time of a surgery, i.e.

$$original\_date_p - arrived\_on_p$$

- $m_{k,p}$: binary, 1 if patient p is assigned to surgeon k and 0 otherwise

Decision Variables

- u_d: integer, the number of additional elective patients in the ICU on day d above c
- $x_{d,p}$: binary, $x_{d,p} = 1$ if patient p is scheduled to have her surgery on day d; otherwise 0
- d_p: the date when the surgery of patient p is scheduled by the model
- $y_{d,p}$: binary, $y_{d,p} = 1$ if patient p stays in the CVICU on day d; otherwise 0
- $z_{d,p}$: continuous, number of hours that the surgery of p lasts on day d. $z_{d,p} = q_p$ if $x_{d,p} = 1$, otherwise 0

Indicator Function

- Assumed feasible window where patient p is available for surgery:

$$g(d, p) = \mathbb{1}\{d_p^{min} \le d < d_p^{max}\}$$

where $d_p^{max} = \min(actual\_date_p + 90, N_d)$,

$$d_p^{min} = \begin{cases} \max[1, original\_date_p - \frac{lead_p}{2}], & lead_p > 20 \\ \max[1, arrived\_on_p], & lead_p \le 20 \end{cases} \tag{11}$$

Next we describe the optimization constraints 12-16 that are equivalent to $x \in Q^{op}$.

Constraint 12 ensures each patient is scheduled exactly once within that patient's window of availability.

$$\sum_{d \in D} x_{d,p} \cdot g(d, p) = 1, \quad \forall p \in P \tag{12}$$

Constraints 13-16 incorporate daily availability of surgeons. Constraint Eq. 13 includes number of hours a surgeon is available on each day for (non PAR) surgeries.

$$z_{d,p} = x_{d,p} q_p, \quad \forall d \in D, p \in P \tag{13}$$

$$\sum_{p \in P} z_{d,p} m_{k,p} \le h_{d,k}, \quad \forall d \in D, k \in K \tag{14}$$

Constraint 15 captures that PAR surgeries can only be done on pre-specified days.

$$\sum_{p \in P^{par}} x_{d,p} m_{k,p} \le par\_day_{d,k}, \quad \forall d \in D, k \in K \tag{15}$$

Constraint 16 ensures each surgeon is not scheduled for PAR surgeries in subsequent days.

$$\sum_{p \in P^{par}} x_{d,p} m_{k,p} + \sum_{p \in P^{par}} x_{d+1,p} m_{k,p} \le 1, \quad \forall d \in D \setminus \{N_d\}, k \in K \tag{16}$$

Appendix A.1: Q^{op} for batch optimization problem

When solving BOP for rolling scheduling, Q^{op} is adjusted accordingly. In addition to the Sets and Parameters specified in Section A, we use the following sets and parameters.

Sets

- $P_b \subset P$: index of batch patients of period b
- $P_b^{past} = \bigcup_{k=1}^{b-1} P_k$: set of patients prior to period b
- $P_b^{par} = P^{par} \cap P_b$

Parameters

- s_b: start date/day of batch b
- $d_{p,b}^{min} = \max(d_p^{min}, s_b)$
- $x_{d,p}^*$: solutions obtained from previous periods

Since the definition of $d_{p,b}^{min}$ may restrict the original time window of availability for some patients, we also adjust the last available date, $d_p^{\max}$, to at least 90 days after the start of the period, i.e.,

$$d_{p,b}^{max} = \max(d_p^{max}, \min(s_b + 90, N_d)) \tag{17}$$

This adjustment allows flexible scheduling as described in Section 3.1. In practice, the definition of $d_{p,b}^{max}$ will not include N_d; we included it for our simulation runs. Although the latter adjustment could potentially increase wait time, it is necessary in ensuring that the set of feasible scheduling solutions is not too restricted, and any resultant increase in wait time

will be penalized by the objective function.

Indicator Function

- Adjusted feasibility window for each patient.

$$g_b(d, p) = \mathbb{1}\{d_{p,b}^{min} \le d < d_{p,b}^{max}\}$$

Constraints 18-22 give the equivalent formulation of $x \in Q^{op}$ in all deterministic, stochastic and robust BOPs.

$$\sum_{d \in D} x_{d,p} g_b(d, p) = 1, \quad \forall p \in P_b \tag{18}$$

$$z_{d,p} = x_{d,p} q_p, \quad \forall d \in D, p \in P_b \tag{19}$$

$$\sum_{p \in P_b} z_{d,p} m_{s,p} \le h_{d,s}, \quad \forall d \in D, s \in S \tag{20}$$

$$\sum_{p \in P_b^{par}} x_{d,p} m_{s,p} \le par\_day_{d,s}, \quad \forall d \in D, s \in S \tag{21}$$

$$\sum_{p \in P_b^{par}} x_{d,p} m_{s,p} + \sum_{p \in P_b^{par}} x_{d+1,p} m_{s,p} \le 1, \quad \forall d \in D \setminus \{N_d\}, s \in S. \tag{22}$$

Appendix B: Solving AC&CG for robust BOP

In the following, we introduce the AC&CG algorithm used for solving the robust BOP.

1. Main problem. We start with a subset of traces in the uncertainty set, $\Omega^t = \{\omega^{(n)} : n = 1, \ldots, |\Omega^t|\} \subseteq \mathcal{U}_b$. Instead of minimizing the worst-case cost over the entire uncertainty set $\mathcal{U}$, the Main problem of AC&CG minimizes the worst-case cost over the subset, Ω^t:

$$\min_{x} \quad \sum_{p \in P_b} \sum_{d=s_b}^{N_d} (d - d_p^{min})^+ x_{d,p} + \beta \cdot \theta \tag{23a}$$

s.t.

$$x_{d,p} = \tilde{x}_{d,p} \quad \forall p \in P_b^{past}, d \in D \tag{23b}$$

$$\sum_{d=s_b}^{N_d} x_{d,p} = 1, \quad \sum_{d=1}^{s_b - 1} x_{d,p} = 0 \quad \forall p \in P_b \tag{23c}$$

$$y^{(n)} = \sum_{d'=\max(d - l_p^{(n)} + 1, 1)} x_{d',p} \quad \forall p \in P_b \cup P_b^{past} \quad \forall d \in D, n = 1, \ldots, |\Omega^t| \tag{23d}$$

$$\sum_{p \in P_b \cup P_b^{past}} y_{d,p}^{(n)} \le c + u_d^{(n)} \quad \forall d \in D, n = 1, \ldots, |\Omega^t| \tag{23e}$$

$$x \in Q^{op}, x_{d,p} \in \{0, 1\} \tag{23f}$$

$$\theta \ge \sum_{d=s_b}^{N_d} f(u_d^{(n)}) \quad \forall n = 1, \ldots, |\Omega^t|. \tag{23g}$$

Constraint 23g sets the variable θ to be equal to the maximum cost of overflow among all traces of LOS realization, $\omega^{(n)}$. The objective 23a thus minimizes the weighted sum of total wait time and the maximum cost of overflow for all $\omega^{(n)} \in \Omega^t$. The remaining constraints mirror those for BSOP-SAA in optimization problem 6, only replacing the set of sampled traces with set Ω^t. The optimal objective value of the Main problem provides a *lower bound* to the optimal objective value of robust BSOP in optimization problem 7.

2. Recourse problem. At each iteration t, after the Main problem is solved with Ω^t, an optimal solution x^{t*} is obtained. The Recourse problem aims to find a trace $\omega^{t*} \in \mathcal{U}_b$ that maximizes the cost of ICU overflow under the given schedule x^{t*}. Since x^{t*} and ω^{t*} are feasible under the original BOP, solving the Recourse problem finds an *upper bound* of the objective value of robust BOP.

$$\max_{\omega = \{l_p : p \in P_b \cup P_b^{past}\}} \quad \min_{y,u} \sum_{d=s_b}^{N_d} f(u_d) \tag{24a}$$

s.t.

$$y_{d,p} = \sum_{d'=\max(d - l_p + 1, 1)}^{d} x_{d',p}^{t*} \quad \forall p \in P_b \cup P_b^{past}, d \in D \tag{24b}$$

$$\sum_{p \in P_b \cup P_b^{past}} y_{d,p} \le c + u_d \quad \forall d \in D \tag{24c}$$

$$y_{d,p} \in \{0, 1\}, u_d \ge 0. \tag{24d}$$

$$l_p^{min} \le l_p \le l_p^{max} \quad \forall p \in P_b \cup P_b^{past} \tag{24e}$$

$$\sum_{p \in A_b} \left[\frac{l_p - l_p^{min}}{l_p^{max} - l_p^{min}} \right] \le \eta \cdot |A_b|. \tag{24f}$$

This formulation follows from expressions 7e-7h, and constraints 24e and 24f use the definition of $\mathcal{U}_b$ in expressions 8 and 9.

One key difficulty of solving the Recourse problem in its current form is that the decision variable l_p appears in the boundary of the summation in constraint Eq. 24b. We follow the reformulation approach in [22] and extend it to convex, piece-wise linear forms of $f(u_d)$. The reformulated Recourse problem is provided in Appendix B.1. We present the full AC&CG algorithm in Algorithm 5.

Algorithm 5 AC&CG for Robust BOP

1: Initialize with $x^* = \mathbf{0}$, $t = 1$, $LB = -\infty$, $UB = \infty$, $\Omega^1 = \{\omega^{(1)}\}$, where $\omega^{(1)} = \{l_p^{min} : p \in P_b \cup P_b^{past}\}$.
2: **while** $t \le T$ **and** $UB - LB > \epsilon$ **do**
3: **1. Main Problem.**
4: Solve the Main problem in optimization problem 23 with Ω^t; Obtain optimal solution x^{t*} with objective value μ^{t*}.
5: Update $x^* \leftarrow x^{t*}$, $LB \leftarrow \mu^{t*}$.
6: **2. Recourse Problem.**
7: Solve the recourse problem in optimization problem 24 with x^*; Obtain optimal solution ω^{t*} with objective value v^{t*}.
8: Update $UB \leftarrow \min\{UB, v^{t*}\}$.
9: **if** $UB - LB > \epsilon$ **then**
10: $\Omega^{t+1} \leftarrow \Omega^t \cup \{\omega^{t*}\}$
11: $t \leftarrow t + 1$
12: **end if**
13: **end while**
14: **return** x^*

Appendix B.1: Solving the recourse problem

The Recourse problem, denoted as $Q(x^{t*})$, involves constraints including l_p decision variables in the boundary of summations. Here, we introduce our MIP reformulation that is readily solvable by Gurobi. We refer readers to [22] for more details and proofs of its validity.

As in [22] we define the variables $v_{d,p}, w_{d,p} \in \{0, 1\}$ for $d \in D$, $p \in P$. The variable $v_{d,p}$ is 1 only if patient p is admitted in the ICU by d. Given the temporary solution x^{t*}, $v_{d,p}$ are constant parameters determined by x^{t*}. The decision variable $w_{d,p}$ is 1 only if patient p leaves the ICU by day d. So, $y_{d,p} = v_{d,p} - w_{d,p}$.

The inner minimization problem of $Q(x^{t*})$ is

$$\min_{u \ge 0} \sum_{d \in D} e_1 u_d^{(1)} + e_2 u_d^{(2)} + e_3 u_d^{(3)} + e_4 u_d^{(4)} + e_5 u_d^{(5)}$$

$$\sum_{p \in P_b \cup P_b^{past}} (v_{d,p} - w_{d,p}) \le c + m - 1 + u_d^{(m)} \quad \forall d, m$$

where $e_1 = 1$, $e_2 = e_3 = e_4 = e_5 = 2$.

We apply strong duality to reformulate the inner minimization as a maximization problem and also substitute for the definition of uncertainty set $\mathcal{U}_b$. Let d_p denote the date of scheduled procedure for patient p according to x^{t*}, i.e., $d_p = \sum_{d \in D} d \cdot x_{d,p}^{t*}$. $Q(x^{t*})$ is reformulated below with decision variables $\lambda_d^{(m)}$ and $w_{d,p}$.

$$\max_{\lambda, w} \sum_{d \in D} \sum_{m=1}^{5} \left[\sum_{p \in P_b \cup P_b^{past}} (v_{d,p} - w_{d,p}) - c - m + 1) \right] \lambda_d^{(m)}$$

$$d_p = \sum_{d \in D} d \cdot x_{d,p}^{t*} \quad \forall p \in P_b \cup P_b^{past}$$

$$w_{d,p} \ge 1, \quad p \in P_b \cup P_b^{past}, d = d_p + l_p^{max}, \ldots, T$$

$$w_{d,p} \le 0, \quad p \in P_b \cup P_b^{past}, d = 0, \ldots, d_p + l_p^{min} - 1$$

$$w_{d,p} \le w_{d+1,p} \quad \forall d \in D, p \in P_b \cup P_b^{past}$$

$$\sum_{d \in D} (v_{d,p} - w_{d,p}) = l_p, \quad \forall p \in P_b \cup P_b^{past} \text{ where } p \notin A_b$$

$$\sum_{p \in A_b} \left[\frac{\sum_{d \in D} (v_{d,p} - w_{d,p}) - l_p^{min}}{l_p^{max} - l_p^{min}} \right] \le \eta \cdot |A_b|.$$

$$0 \le \lambda_d^{(m)} \le e_m \quad \forall d, m$$

$$w_{d,p} \in \{0, 1\}, \quad \forall d, p$$

For the optimal solution, we must have $\lambda_d^{(m)} \in \{0, e_m\}$.

The formulation above involves a bilinear term, $w_{d,p}\lambda_d^{(m)}$. Since $w_{d,p} \in \{0, 1\}$, we can reformulate the problem by using $q_{d,p}^{(m)} = w_{d,p}\lambda_d^{(m)}$ for all d, p, m. The final reformulation of the Recourse problem is the following.

$$\begin{aligned} Q(x^{t*}) = \max_{\lambda, w} & \sum_{d \in D} \sum_{m=1}^{5} \sum_{p \in P_b \cup P_b^{past}} v_{d,p}\lambda_d^{(m)} - \\ & \sum_{d \in D} \sum_{m=1}^{5} \sum_{p \in P_b \cup P_b^{past}} q_{d,p}^{(m)} - \\ & \sum_{d \in D} \sum_{m=1}^{5} (c + m - 1)\lambda_d^{(m)} \end{aligned}$$

$$d_p = \sum_{d \in D} d \cdot x_{d,p}^{t*} \quad \forall p \in P_b \cup P_b^{past}$$

$$w_{d,p} \geq 1, \quad p \in P_b \cup P_b^{past}, d = d_p + l_p^{max}, \ldots, T$$

$$w_{d,p} \leq 0, \quad p \in P_b \cup P_b^{past}, d = 0, \ldots, d_p + l_p^{min} - 1$$

$$w_{d,p} \leq w_{d+1,p} \quad \forall d \in D, p \in P_b \cup P_b^{past}$$

$$\sum_{d \in D} (v_{d,p} - w_{d,p}) = l_p, \quad \forall p \in P_b \cup P_b^{past} \text{ where } p \notin A_b$$

$$\sum_{p \in A_b} \left[\frac{\sum_{d \in D} (v_{d,p} - w_{d,p}) - l_p^{min}}{l_p^{max} - l_p^{min}} \right] \leq \eta \cdot |A_b|$$

$$q_{d,p}^{(m)} \geq \lambda_t^{(m)} - e_m (1 - w_{d,p}) \quad \forall d, p, m$$

$$q_{d,p}^{(m)} \leq e_m w_{d,p}, \; q_{d,p}^{(m)} \leq \lambda_d^{(m)} \quad \forall d, p, m$$

$$q_{d,p}^{(m)} \geq 0, \lambda_d^{(m)} \in \{0, e_m\}, w_{d,p} \in \{0, 1\} \quad \forall d, p, m.$$

Appendix C: Additional numerical results

This section presents additional results on the optimality gap of AC&CG and biweekly scheduling.

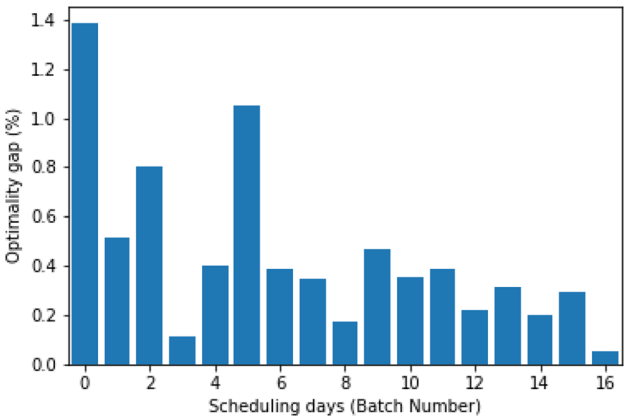
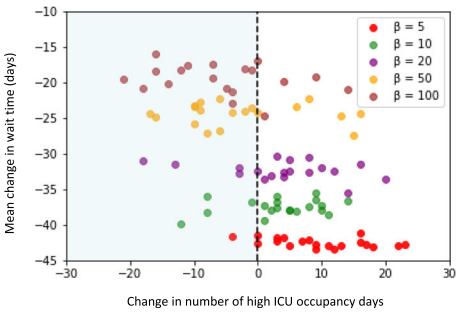
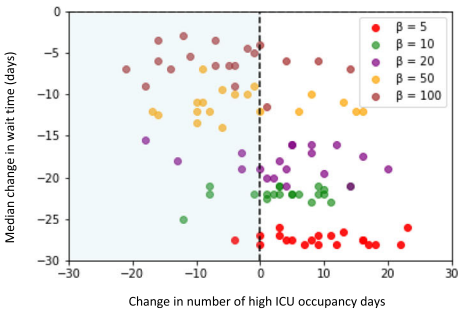


Fig. 12 The optimality gap, $\frac{UB-LB}{LB}$, of most batches are below 1% after 10 iterations of BROP-AG&CG

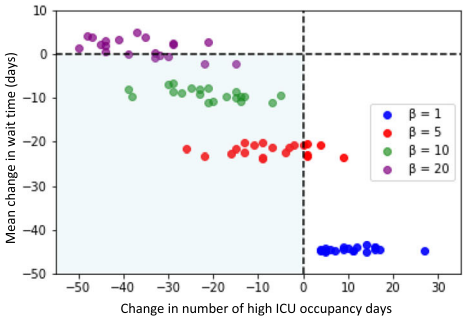


(a) Mean change in wait times vs. ICU congestion

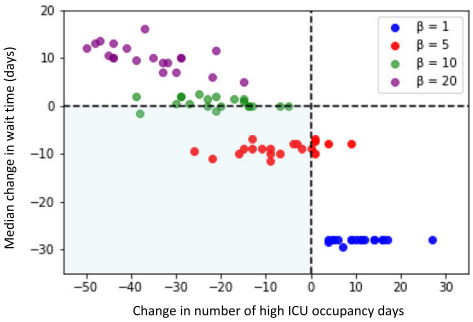


(b) Median change in wait times vs. ICU congestion

Fig. 13 (Biweekly Scheduling) Performance trade-off of Standard-SRO between patient wait times and ICU congestion using different values of β compared to the status quo. We also include greater values of β to show that increasing β further leads to longer wait times but insignificant reduction in ICU congestion

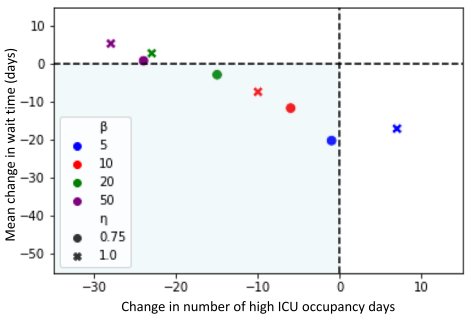


(a) Mean change in wait times vs. ICU congestion

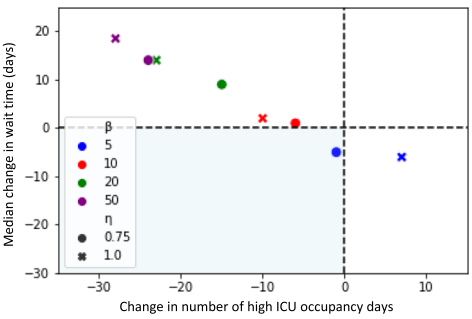


(b) Median change in wait times vs. ICU congestion

Fig. 14 (Biweekly Scheduling) Performance trade-off of Conservative-SRO between patient wait times and ICU congestion using different values of β compared to the status quo



(a) Mean change in wait times vs. ICU congestion



(b) Median change in wait times vs. ICU congestion

Fig. 15 (Biweekly Scheduling) Performance trade-off of RRO between patient wait times and ICU congestion using different values of β, η compared to the status quo

Declarations

Competing interests The authors have no competing interests to declare that are relevant to this paper. J. Blanchet acknowledges support from NSF grant 2118199.

Ethical approval This research uses secondary data collected from standard clinical practice with minimal risks. Ethical approval was not needed. Anonymity and confidentiality of information were assured.

References

1. Beliën J, Demeulemeester E, Cardoen B (2009) A decision support system for cyclic master surgery scheduling with multiple objectives. J Sched 12(2):147–161
2. Benzaid M, Lahrichi N, Rousseau LM (2020) Chemotherapy appointment scheduling and daily outpatient-nurse assignment. Health Care Manag Sci 23:34–50
3. Bertsekas DP, Castanon DA (1999) Rollout algorithms for stochastic scheduling problems. J Heuristics 5:89–108
4. Bertsimas D, Sim M (2004) The price of robustness. Oper Res 52(1):35–53
5. Chan CW, Farias VF, Bambos N, Escobar GJ (2012) Optimizing intensive care unit discharge decisions with patient readmissions. Oper Res 60(6):1323–1341
6. Chen E, Naylor CD (1994) Variation in hospital length of stay for acute myocardial infarction in ontario, canada. Med Care 32(5):420–435
7. Davison AC, Hinkley DV (1997) Bootstrap methods and their application. Cambridge University Press
8. Ettema RG, Peelen LM, Schuurmans MJ, Nierich AP, Kalkman CJ, Moons KG (2010) Prediction models for prolonged intensive care unit stay after cardiac surgery: systematic review and validation study. Circulation 122(7):682–689

9. Fairley M, Scheinker D, Brandeau ML (2019) Improving the efficiency of the operating room environment with an optimization and machine learning model. Health Care Manag Sci 22(4):756–767
10. Fügener A, Hans EW, Kolisch R, Kortbeek N, Vanberkel PT (2014) Master surgery scheduling with consideration of multiple downstream units. Eur J Oper Res 239(1):227–236
11. Guinet A, Chaabane S (2003) Operating theatre planning. Int J Prod Econ 85(1):69–81
12. Gurobi Optimization LLC (2021) Gurobi Optimizer reference manual. URL http://www.gurobi.com
13. Hsu VN, De Matta R, Lee CY (2003) Scheduling patients in an ambulatory surgical center. Nav Res Logist 50(3):218–238
14. Keyvanshokooh E, Shi C, Van Oyen MP (2021) Online advance scheduling with overtime: A primaldual approach. Manuf Serv Oper Manag 23(1):246–266
15. Keyvanshokooh E, Kazemian P, Fattahi M, Van Oyen MP (2022) Coordinated and prioritybased surgical care: An integrated distributionally robust stochastic optimization approach. Prod Oper Manag 31(4):1510–1535
16. Kramer AA, Zimmerman JE (2010) A predictive model for the early identification of patients at risk for a prolonged intensive care unit length of stay. BMC Med Inform Decis Mak 10(1):1–16
17. LaFaro RJ, Pothula S, Kubal KP, Inchiosa ME, Pothula VM, Yuan SC, Maerz DA, Montes L, Oleszkiewicz SM, Yusupov A et al (2015) Neural network prediction of icu length of stay following cardiac surgery based on pre-incision variables. PLoS One 10(12):e0145395
18. LeDell E, Poirier S (2020) H2O AutoML: Scalable automatic machine learning. ICML Workshop on Automated Machine Learning
19. Maharlou H, Kalhori SRN, Shahbazi S, Ravangard R (2018) Predicting length of stay in intensive care units after cardiac surgery: comparison of artificial neural networks and adaptive neuro-fuzzy system. Healthc Inform Res 24(2):109
20. Min D, Yih Y (2010) Scheduling elective surgery under uncertainty and downstream capacity constraints. Eur J Oper Res 206(3):642–652
21. Nassar AP Jr, Caruso P (2016) Icu physicians are unable to accurately predict length of stay at admission: a prospective study. Int J Q Health Care 28(1):99–103
22. Neyshabouri S, Berg BP (2017) Two-stage robust optimization approach to elective surgery and downstream capacity planning. Eur J Oper Res 260(1):21–40
23. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V et al (2011) Scikit-learn: Machine learning in Python. J Mach Lear Res 12:2825–2830
24. Pham DN, Klinkert A (2008) Surgical case scheduling as a generalized job shop scheduling problem. Eur J Oper Res 185(3):1011–1025
25. Rahimi I, Gandomi AH (2021) A comprehensive review and analysis of operating room and surgery scheduling. Arch Comput Methods Eng 28(3):1667–1688
26. Rohleder TR, Klassen KJ (2002) Rolling horizon appointment scheduling: a simulation study. Health care Manag Sci 5(3)
27. Shehadeh KS, Padman R (2021) A distributionally robust optimization approach for stochastic elective surgery scheduling with limited intensive care unit capacity. Eur J Oper Res 290(3):901–913
28. Tabbutt S, Schuette J, Zhang W, Alten J, Donohue J, Gaynor JW, Ghanayem N, Jacobs J, Pasquali SK, Thiagarajan R et al (2019) A novel model demonstrates variation in risk adjusted mortality across pediatric cardiac intensive care units after surgery. Pediatr Crit Care Med: J Soc Crit Care Med World Federation Pediatr Intensive Crit Care Soc 20(2):136
29. Tsai PFJ, Chen PC, Chen YY, Song HY, Lin HM, Lin FM, Huang QP (2016) Length of hospital stay prediction at the admission stage for cardiology patients using artificial neural network. J Healthc Eng 2016
30. Tsai PFJ, Chen PC, Chen YY, Song HY, Lin HM, Lin FM, Huang QP (2016) Length of hospital stay prediction at the admission stage for cardiology patients using artificial neural network. J Healthc Eng 2016
31. Vicente FG, Lomar FP, Mélot C, Vincent JL (2004) Can the experienced icu physician predict icu length of stay and outcome better than less experienced colleagues? Intensive Care Med 30(4):655–659
32. Weissman GE, Hubbard RA, Ungar LH, Harhay MO, Greene CS, Himes BE, Halpern SD (2018) Inclusion of unstructured clinical text improves early prediction of death or prolonged icu stay. Crit Care Med 46(7):1125–1132
33. Whellan DJ, Zhao X, Hernandez AF, Liang L, Peterson ED, Bhatt DL, Heidenreich PA, Schwamm LH, Fonarow GC (2011) Predictors of hospital length of stay in heart failure: findings from get with the guidelines. J Card Fail 17(8):649–656
34. Yang CS, Wei CP, Yuan CC, Schoung JY (2010) Predicting the length of hospital stay of burn patients: Comparisons of prediction accuracy among different clinical stages. Decis Support Syst 50(1):325–335
35. Zhang J, Dridi M, El Moudni A (2019) A two-level optimization model for elective surgery scheduling with downstream capacity constraints. Eur J Oper Res 276(2):602–613
36. Zhu S, Fan W, Yang S, Pei J, Pardalos PM (2019) Operating room planning and surgical case scheduling: a review of literature. J Comb Optim 37(3):757–805