Lab 4: Smoothing Inpatient Load in Hospitals

Pair Names: ______________________

1 Motivation

Today's lab explores a simplified version of a problem addressed by a recent collaborative project between MIT Business School faculty and Massachusetts General Hospital (MGH), a large hospital in Boston.

This work is part of a growing interest in how large-scale data about health-care operations can be harnessed to help reduce inefficiencies (and associated costs) while providing high-level care.

The collaboration was composed of optimization researchers, hospital administrators, and an anesthesiologist: the team aimed to improve the processing of patients through the hospital (it had been observed that certain bottlenecks in the system were wasting valuable hospital resources as well as producing long waiting times for patients to enter inpatient care after their surgery in the operating room was complete).

This lab will:

- *Continue to develop our Integer-Program Modeling Skills.*
- *Examine practical examples of the "Integrality Gap" in a larger example.*
- *Comment on a common theme of true consulting projects: often "cultural constraints" must be accommodated in a way that restricts the space of feasible solutions (thus adversely affecting the objective). Providing a quantitative measure of just how adverse these effects are to objectives that are widely embraced across the organization can provide valuable insight to stake-holders.*

Download the article from Moodle and read the Introduction and Results Sections.

Make sure that you understand the figures which explain what is going wrong with the previous approach to scheduling surgeries.

2 A Simplified Model

We'll consider the case of a small hospital that is faced with requests to schedule 100 unique elective surgeries (aka, non-emergency surgeries) over the next 10 days.

Throughout this lab, for your own reference, pay attention to the size of the problem (number of integer variables and constraints) and take note of the information AMPL outputs about the number of iterations, etc.

Each surgery request requires 3 hospital resources: operating room (OR) time in hours, some number of days of inpatient recovery in the hospital (in the form of a hospital bed), and some number of hours a day of nursing per day during the period of inpatient recovery. For today, we will assume that each of these quantities is known deterministically. The hospital has a limited number of each of these resources each day, and would like to complete as many surgeries as possible during the next 10 days (where completion means: the surgery is performed, the patient recovers and then leaves the hospital by the tenth day).

Each row of the following table contains data about the resources required for one of the 100 surgeries that may be scheduled.

Surgery	hours in OR	Days in hospital bed	hours of nursing per day
1	2	4	3
2	4	2	2
3	1.5	6	6
.	.	.	.
.	.	.	.
.	.	.	.
100	5	2	2

The data describing the 100 surgeries and the hospital's capacity constraints is given in the file `lab4size100.dat`

3 Formulating an IP

- What explicit decisions do we have to make to specify a solution in this problem? Create integer decision-variables to describe these decisions.

- What implicit decisions are forced by the first set of decision variables that you wrote down? *Hint: You need to keep track of who is occupying inpatient beds on each day...*

- Write down constraints that describe the relationship between the explicit and implicit variables. Think carefully about how to use the data you have to index the expressions in these constraints.

  Hint: Before you try to write notation for this, think about a way to express verbally what you want to force. Based on when a surgery is scheduled...what do you know about bed use?

- Each surgery is scheduled at most once! Write a constraint for this:

- Write down constraints that describe the hospital's limited resources during each of the ten days:

- Make sure that any patient scheduled for surgery during the 10 days leaves the hospital by the end of the tenth day:

- What is the objective?

Now download the model file `lab4.mod`.

Read through the model file: make sure that you recognize the constraints that you have formulated. There are a few tricks used to avoid errors in indexing.

- What is the maximum number of surgeries can be completed during the 10-day period?

- Remove the integrality constraints from the decision variables and solve the LP. What is the maximum number of "fractional-surgeries" that can be accomplished? *Discuss with another student (no need to write): practically speaking, what do you think might be happening in this fractional solution? Why is it so much bigger?*

- Choose one of the parameters that describes hospital resources (`Caporhours`, `Capbedsperday`, `Capnursehours`) to vary. Draw by hand a graph that shows the value of the IP and the value of the LP for $1/2$ of the current parameter value, $1/1$ of the current parameter value, $3/2$ of the current parameter value, $2/1$ of the current parameter value (you've already computed one of these pair of points!). **Describe what you see happening in your graph.**

- Return to the original parameter values. Assume that the numbering of the surgery requests is actually a priority list due to a classic notion of fairness: **"first-come, first-served."** For example, considering two surgery requests i and j with $i < j$: if j is scheduled on day 7, then i must be scheduled on one of the days 1,2,3,4,5,6,7.

  How could you incorporate this notion of fairness into your IP and AMPL model?

- With the additional "first-come first-served" constraints: what is the maximum number of surgeries can be completed during the 10-day period?

- **A Relaxed Notion of Fairness:** Because of the large impact on the objective of the "first-come, first-served" constraints, our consulting team has decided to propose a relaxed notion of fairness which might be described as: **"first-come, served."** For example, considering two surgery requests i and j with $i < j$: if j is scheduled on day 7, then i must be scheduled sometime during the 10-day period. We might say that for $i < j$: if j is scheduled, i is scheduled (and no more than 10 days later).

  How could you incorporate this notion of fairness into your IP and AMPL model?

- Being careful to comment out the "first-come, first-served" constraints, and with the additional "first-come, served" constraints: what is the maximum number of surgeries can be completed during the 10-day period?

- Write 2-3 sentences interpreting the differences between applying these two notions of fairness.

- In real life, the amount of time required in the operating room is not known deterministically. If we plan as if it is known deterministically, we risk the possibility that a single surgery running over time could cause a cascade that pushes back the scheduled times of many later surgeries. How might you modify your model in order to provide some protection against such cascade events?

4 Back to the Article

Changes in Practice: Skim the Materials and Methods, Results, and Appendix A of the article. In Appendix A you can see the formulation of an Integer Program that is not much more complex than those we've encountered. The results of this consulting project were considered sufficiently compelling that MGH (which has 50+ operating rooms) piloted implementing hospital-wide the changes suggested by the consulting team. This allowed smoothing of the elective-surgery load so that time blocks could be protected for emergency patients during peak times. Here is the Abstract on the implementation study, published recently in the ***Annals of Surgery***:

Annals of Surgery:
Post Author Corrections: January 23, 2015
doi: 10.1097/SLA.0000000000001003
Original Article: PDF Only

Pooled Open Blocks: Shorten Wait Times for Nonelective Surgical Cases.

Zenteno, Ana C. PhD; Carnes, Tim PhD; Levi, Retsef PhD; Daily, Bethany J. MHA; Price, Devon MBA, MSc; Moss, Susan C. MPH; Dunn, Peter F. MD

Abstract

Objective: Assess the impact of the implementation of a data-driven scheduling strategy that aimed to improve the access to care of nonelective surgical patients at Massachusetts General Hospital (MGH).

Background: Between July 2009 and June 2010, MGH experienced increasing throughput challenges in its perioperative environment: approximately 30% of the nonelective patients were waiting more than the prescribed amount of time to get to surgery, hampering access to care and aggravating the lack of inpatient beds.

Methods: This work describes the design and implementation of an "open block" strategy: operating room (OR) blocks were reserved for nonelective patients during regular working hours (prime time) and their management centralized. Discrete event simulation showed that 5 rooms would decrease the percentage of delayed patients from 30% to 2%, assuming that OR availability was the only reason for preoperative delay.

Results: Implementation began in January 2012. We compare metrics for June through December of 2012 against the same months of 2011. The average preoperative wait time of all nonelective surgical patients decreased by 25.5% ($P < 0.001$), even with a volume increase of 9%. The number of bed-days occupied by nonurgent patients before surgery declined by 13.3% whereas the volume increased by 4.5%.

Conclusions: The large-scale application of an open-block strategy significantly improved the flow of nonelective patients at MGH when OR availability was a major reason for delay. Rigorous metrics were developed to evaluate its performance. Strong managerial leadership was crucial to enact the new practices and turn them into organizational change.

Impact of Marginal Improvement: An important theme in many logistics settings is that making even incremental improvements in efficiency can have huge impacts when they are deployed across a large system.

Cultural/Political Constraints: You may wonder why the consulting team limits itself to permuting time blocks scheduled by surgeons rather than creating a tool that schedules each surgery independently as we have done in this lab. This is mentioned cryptically in the 2nd paragraph of the discussion section; surgeons who historically had the authority to schedule specified time blocks autonomously were very resistant to being told when they would conduct each surgery. The system described in the paper optimizes over a space of feasible solutions that represents a compromise between independent surgeons being in charge of scheduling their own time blocks and a major theme in optimization: that with increased coordination comes increased efficiency.

An optimization framework for smoothing surgical bed census via strategic block scheduling

Tim Carnes Devon Price Retsef Levi Peter F. Dunn
Bethany J. Daily Sue Moss

October 2, 2011

1 Introduction

Improving the flow of patients throughout a hospital is a key way to ensure both better and more cost-effective health care. One approach to achieving this goal is increasing the number of resources available. The difficulty to employing this tactic is that it is often cost-prohibitive, which motivates the alternative method of getting better usage out of the existing resources and removing systemic inefficiencies. We derive a method to increase the efficiency of the perioperative system at Massachusetts General Hospital (MGH), but our approach is general enough so that it can be applied at many other hospitals as well.

A patient undergoing a surgical operation will typically visit at least three areas of the hospital: an operating room (OR), then the post anesthesia care unit (PACU), and then a surgical inpatient bed, provided that the patient is not discharged to home directly. In Figure 1 we can see the amount of patients who wait in an OR to get into the PACU, as well as the occupancy of the PACU by hour of day. In both cases we see a peak load in the early afternoon, which corresponds to the system reaching capacity. If there were always inpatient beds available, then the PACU should not become backlogged, and if the PACU did not fill up then there should not be any patients occupying an OR waiting for a PACU bed. This identifies the inpatient beds as a key bottleneck of the perioperative environment, and improving access to beds will result directly in improved flow.

A hospital does not have the luxury of specifying in advance when every patient will be admitted and discharged from the hospital. For example, patients arriving through the emergency department may have to undergo a surgical operation as soon as possible, and the arrival of such cases occurs with unpredictable timing. Patients having surgical procedures are either electively scheduled or else are placed into the surgical schedule from a waitlist. In order to understand the way in which beds are currently being used and identify opportunities for getting more use out of this resource, we perform a system-wide analysis of the average weekly bed usage, and break it down further into different patient populations. In Figure 2, we show the average midnight census

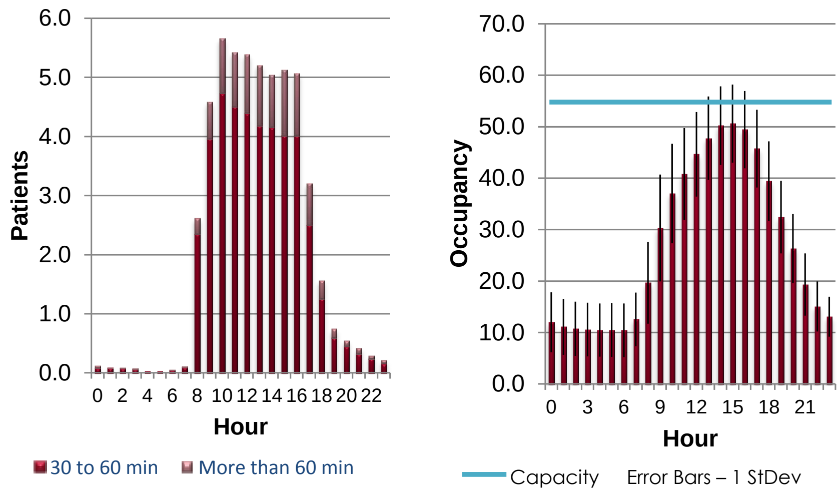

Figure 1: The number of patients waiting in an OR to get into the PACU by hour of the day on left, and the occupancy of the PACU by hour of day on right.

for surgical inpatient floors, broken down by whether the patient underwent a surgical operation, and if so whether they had a scheduled operation or were put in via the waitlist. The scheduled population is further broken down into whether the patient had a short length of stay (less than 7 days) or not.

From this chart we can see that the bed usage at MGH over the course of the week is unimodal with a peak of usage on Wednesdays, which pushes to the limit of available capacity. This means that there are extra beds not being used on some days, while there are not enough beds on others. We can now see that in addition to a daily capacity problem peaking midday, we also have a weekly capacity problem peaking mid-week. In this work we focus on the weekly capacity problem, as any improvement to the weekly situation will also improve the flow of daily operations. If bed-usage could be leveled out so that the same number of beds could be used each day, then one could serve the same size patient population, or even a slightly larger one, without running into capacity limitations. Furthermore, as can be seen in Figure 2, the variation in bed census across the days of the week at MGH is predominantly due to the patients who are electively scheduled, and who have short lengths of stay. If we can alter the days of the week that this patients have their surgical operations scheduled, we can hope to smooth out the bed usage, thereby improving the flow.

A patient is electively scheduled by a surgeon or surgical service, and the day of week of the surgery is largely determined by when that surgeon has the resources available for the type of surgery required. A surgeon's access to the OR is typically set according to a monthly repeating block schedule, which dictates who has ownership of each OR on each day. Having ownership of an OR allows a surgeon exclusive access to scheduling cases to be performed in that OR on the specified date. As the date draws nearer, a block is often released so

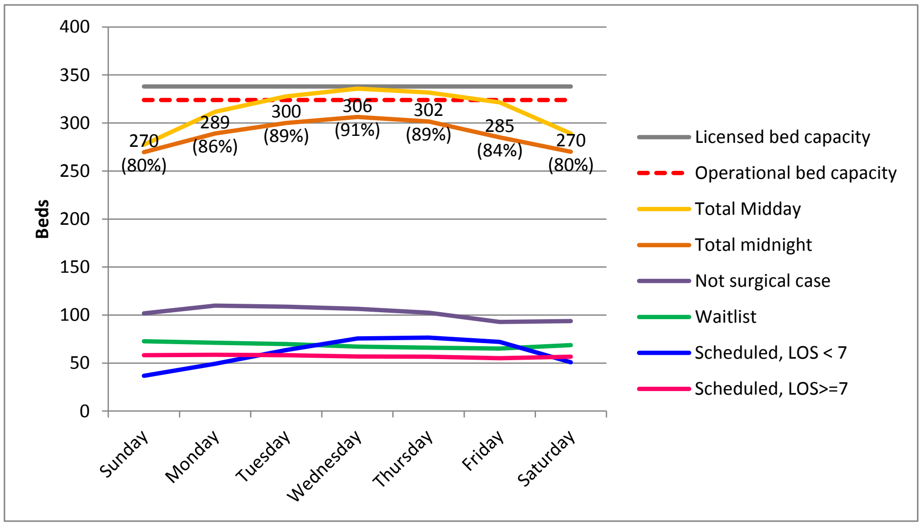

Figure 2: The average bed census across days of the week for surgical inpatient units, broken down into distinct patient populations. Percentages are shown indicating proportion of licensed bed capacity taken by midnight census. Midnight census is shown for each population, other than "total midday."

that other surgeons may schedule cases in the unused time.

The block schedule is set by the OR administration, and by strategically changing what days of the week particular surgeons have access to the OR, one can affect when different types of elective surgeries are performed across the week, and hence the associated lengths of stays of patients arriving electively into the perioperative system. We apply an optimization methodology to rearrange the block schedule so as to reduce the downstream peak inpatient bed usage, thereby increasing the effective capacity without increasing the number of beds. Note that we are not looking at changing how surgical blocks are allocated to services and surgeons, nor how they are currently being used, but rather simply changing how the assigned blocks are distributed amongst the days of the week. This is the first time this type of approach has been applied to an academic hospital of the size of MGH, and also that has been put into implementation. MGH runs 52 ORs each day, which leads to 38,000 surgical operations being performed annually. Of the 907 beds in the hospital, 338 are in surgical units.

2 Literature Review

Cardoen et al. [4] provide a thorough literature review of prior research involving strategic surgical scheduling. Of particular relevance to our approach is the work of Chow et al.[5] who use Monte Carlo simulation to predict what the bed demands will be for a given surgical block schedule. They also use integer programming to optimize peak bed occupancies by changing the surgical block schedule as well as suggesting the case mix for individual surgical blocks. From their model they create a set of scheduling guidelines to be used by hospital managers. Their methodology is applied to the Royal Jubilee Hospital in Victoria, B.C, which runs 16 operating rooms and has 100 surgical inpatient beds. Beliën and Demeulemeester[1, 2] employ several techniques they apply for the Virga Jesse Hospital in Hasselt, Belgium, where the operating room complex consists of 9 rooms. With integer programming they minimize the maximum daily mean, while with quadratic programming they level the mean bed occupancy by minimizing the quadratic sum. Finally, with simulated annealing they minimize the total expected bed shortage. Van Oostrum [12] use mathematical programming with probabilistic constraints to group elective surgical cases into repeating sequences. While they consider the Erasmus Medical Center as their application, in their computational studies they vary the number of operating rooms up to 20. Denton et al.[6] make use of stochastic optimization and heuristics to tackle a slightly different problem: sequencing surgeries based on variability to reduce waiting time for surgeons and staff when surgeries take longer than scheduled. Here they consider Fletcher Allen Health Care, which serves Vermont and upstate New York, where there are 12 operating rooms.

3 Materials and Methods

We make use of an optimization technique known as *integer programming*. In this method, one expresses a problem as optimizing a linear function of variables subject to a set of linear constraints, while also allowing constraints to force variables to take on integer values. The variables are used to represent particular decisions.

The current block schedule is set so that the ownership of a particular OR on a particular day of the week may change depending on which week of the month it is. For example, OR 7 may belong to surgeon A on the first Monday of each month, but may belong to surgeon B on the second Monday of each month. For our approach we take a less granular view of the block structure. When deciding how to rearrange the block schedule, we will only consider moving, as a group, all of the blocks that correspond to an OR-weekday pair. Hence, in our example, we may decide to move all blocks in OR 7 on Mondays to OR 3 on Wednesdays, so surgeon A would have OR 3 on the first Wednesday of each month, and surgeon B would have OR 3 on the second Wednesday of each month. Our decision variables will thus capture for a group of blocks in each OR, what OR-weekday pair they will move to. In our integer program, we will have a variable for each pair of OR-weekday pairs, indicating whether the blocks in the first OR-weekday pair will move to the second OR-weekday pair.

The reason we only consider moving the surgical blocks in groups instead of individually is twofold. First, by keeping the OR-weekday grouping of blocks intact, we produce schedules that are more similar to the current schedule, and preserve some of the fine-tuning the OR administration has done to create a feasible schedule. Second, only considering how to move around groups of blocks reduces the total size of the problem, allowing for faster computation times.

Our objective is to minimize the peak usage of the inpatient beds that result from rearranging the block schedule, so our model must be able to calculate the bed usage based on a given schedule. To this end, we make use of one year of surgical case data at MGH, from April 2010 through March 2011. This time frame is large enough to capture a representative use of surgical blocks, as well as incorporating seasonal variation that may occur throughout the year. We also obtained admitting records for this time period, so that we are able to track where a patient stays both before and after surgery. In our model, we store the data for when an electively-scheduled surgical patient stays in a given inpatient unit as an offset from that patient's day of surgery. For our purposes we assume that if the block a surgical patient was scheduled in moves to a different weekday, then the dates of that patient's stay in an inpatient unit would have the same offset from the day of surgery. In this manner we can reconstruct what the bed census would have been for the different surgical inpatient units based on different block schedules. Patients may move through different floors during the course of their stay at the hospital, and these types of occurrences are neatly captured by how our input is structured. There are other possible objectives as well, such as minimizing the peak number of admittances and/or discharges, which could also potentially help improve the overall flow.

Note that by using historical data to model a surgical block's impact on inpatient bed units presents a challenge when new surgeons are allocated block time. These surgeons do not necessarily have a historical usage pattern to draw on, so coordination with OR administrators and surgical service chiefs are needed to predict the surgical case mix expected to be performed by a new surgeon. Furthermore, not all surgical services allocate block time in the same way. While many assign blocks to individual surgeons, some services leave their blocks open to all surgeons in the service. In this case recommendations for change must center on the surgical case mix aimed for on a given weekday as opposed to simply permuting the different surgeon's day of access. The changes suggested by our model do not affect the amount of OR time any surgeon has, though changing the day a surgeon has access does have significant ramifications such as requiring clinic schedules to be rearranged. These are nontrivial changes to be made, but it is precisely because bed access is such a critical bottleneck that we are able to carry this work through to implementation.

By allocating a variable to each inpatient unit in our integer program, we can use linear constraints to force these variables to be at least the average census for that unit, for each day of the week. In setting the goal of the integer program to minimize the sum of these variables, we are in effect minimizing the peak average census for each surgical inpatient unit. Naturally, there are many additional constraints we must impose so that the way surgical blocks are moved around still constitutes a reasonable block schedule, which includes:

- No surgeon has more than one block on a given day
- Blocks may only move to ORs that can support the surgeon and types of surgery associated with the block
- All blocks on Thursdays are shorter due to grand rounds, so these blocks, if moved, must move as a group to a different weekday
- Each surgical service has a reasonable distribution of blocks across the week
- Blocks assigned to surgical chiefs are not moved
- Certain blocks require interaction between different ORs, such as transplants, and thus must be moved as groups

With all of the additional constraints in place, the only permutations allowed on the block schedule are ones which are acceptable to the OR administration. For a full specification of the integer program, see the Appendix. Using the developed model, we are able to offer suggested changes to the block schedule that deliver a more even predicted usage of inpatient beds. Furthermore, the output schedule was then simulated on actual data from MGH. The model is solved using the optimization tool CPLEX 11.2.1, and integer program is specified using the interface AMPL version 20081120 on a PC with 2.83 GHz and 3 GB of RAM. Finding a solution with peak census within one bed of optimality typically took about five minutes, while an hour was usually sufficient to produce an additive optimality gap of 0.1 beds. Most of the runs were not solved to optimality, though the optimality gap was more more than sufficient for the application.

4 Results

By applying the integer program model, we obtain a new block schedule that has a predicted bed usage which is much more evenly distributed across the days of the week. A plot showing the original census as compared with the census resulting from the optimal permuted schedule for both midnight and midday is shown in Figure 3.

The result of the optimization shows that on average the midnight peak census over all of the floors is reduced by eleven beds. The midday average peak census also is reduced by a similar amount. Because the model takes into account the individual floors in achieving its objective, we avoid having one surgical unit have an improved peak census at the expense of a different unit having a worse peak. Instead each unit is improved as a result of the permutation, though some units may improve more than others. In addition to improving the census on average, we wish to make sure that we are also improving the bed usage across the individual days of the year. Another view of analyzing the downstream bed usage induced by the two block schedules is shown in Figure 4. Here the percentage of Wednesdays where each census exceeds a particular value is indicated.

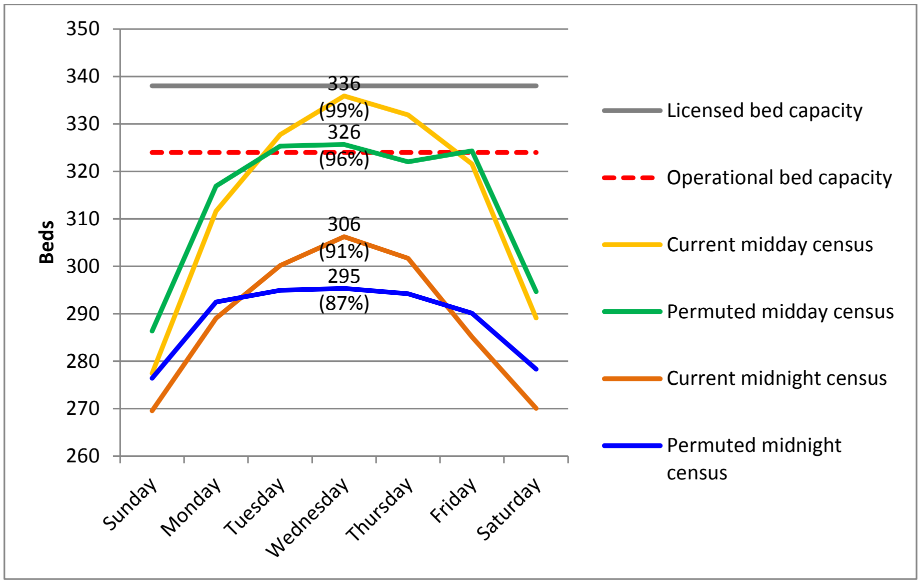

Figure 3: The current versus permuted average census across the days of the week, for both midnight and midday. Percentages shown are taken out of total licensed bed capacity.

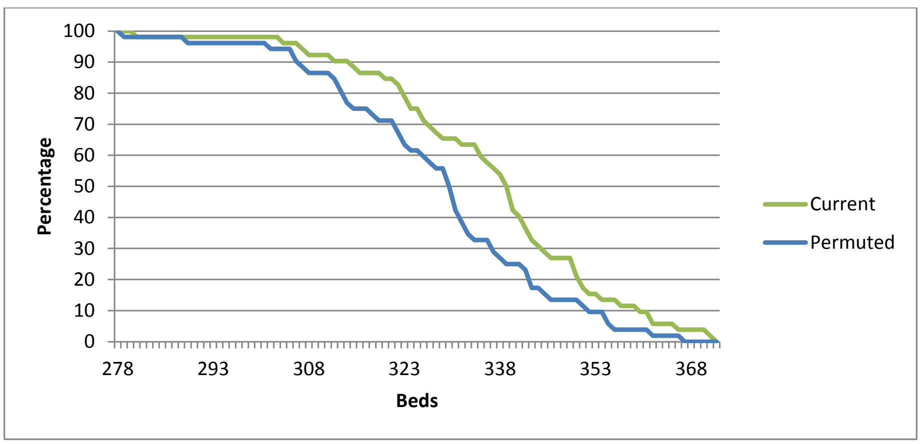

Figure 4: The percentage of Wednesdays for which the current and permuted censuses exceeds a particular number of beds.

From the plot we can see the peak in the middle of the week is reduced not just on average but also across the days of the year. In particular the percentage of Wednesdays for which the census exceeds 333 beds goes from 63% in the current schedule to 35% in the permuted schedule.

5 Discussion

While all of our work was focused on improving patient flow at MGH, our approach is sufficiently general to be applicable to any hospital that makes use of a surgical block schedule to manage its ORs. We expect the potential reduction in peak bed usage to dramatically alleviate the current bottleneck issues that arise from the current bed capacity limitations. As the suggested schedule resulting from our optimization will be put into use, we will be able measure the performance of the solution and analyze how well the outcome matches our predictions. This will provide additional information that can be used to make future modifications and continue to improve the flow of the perioperative environment.

In order to get to the point of implementation, we have had much communication with the individual surgical services. This has resulted in many iterations of development, and the addition of special constraints into the model. Based on the feedback from the services, we are able to use the tool to evaluate the impact of the changes, and to understand the tradeoffs involved with balancing surgeon convenience and peak bed usage reduction. In this way our model has become a tool to organize large-scale institutional change, and if we can demonstrate successful results then this can heavily motivate other hospitals similarly facing bed capacity issues.

All of the work in this paper has been focused on the variance of bed usage occurring across days of the week. Additional work can be done to investigate the potential of smoothing bed usage over the course of a day, as currently most capacity issues occur around midday. There is also potential to alleviate other potential bottlenecks to flow in the perioperative system, such as managing PACU transfers, as well as decreasing the turnaround time between surgical cases. The impact of resolving inefficiencies within the OR and PACU environments will be much greater at MGH, once the bottleneck of inpatient beds has been resolved.

References

[1] J. Beliën and E. Demeulemeester. Building cyclic master surgery schedules with leveled resulting bed occupancy. *European Journal of Operational Research*, 176(2):1185–1204, 2007.

[2] J. Beliën, E. Demeulemeester, and B. Cardoen. A decision support system for cyclic master surgery scheduling with multiple objectives. *Journal of scheduling*, 12(2):147–161, 2009.

[3] B. Cardoen and E. Demeulemeester. Operating room planning and scheduling: A classification scheme. 2010.

[4] B. Cardoen, E. Demeulemeester, and J. Beliën. Operating room planning and scheduling: A literature review. *European Journal of Operational Research*, 201(3):921–932, 2010.

[5] V. S. Chow, M. L. Puterman, N. Salehirad, W. Huang, and D. Atkins. Reducing surgical ward congestion through improved surgical scheduling and uncapacitated simulation. *Production and Operations Management*, 20(3):418–430, 2011.

[6] B. Denton, J. Viapiano, and A. Vogl. Optimization of surgery sequencing and scheduling decisions under uncertainty. *Health Care Management Science*, 10(1):13–24, 2007.

[7] M. Harders, M. Malangoni, S. Weight, and T. Sidhu. Improving operating room efficiency through process redesign. *Surgery*, 140(4):509–516, 2006.

[8] P. Harper and A. Shahani. Modelling for the planning and management of bed capacities in hospitals. *Journal of the Operational Research Society*, pages 11–18, 2002.

[9] J. May, W. Spangler, D. Strum, and L. Vargas. The surgical scheduling problem: Current research and future opportunities. *Production and Operations Management*, 2011.

[10] P. Santibáñez, M. Begen, and D. Atkins. Surgical block scheduling in a system of hospitals: an application to resource and wait list management in a british columbia health authority. *Health care management science*, 10(3):269–282, 2007.

[11] M. Van Houdenhoven, J. Van Oostrum, E. Hans, G. Wullink, and G. Kazemier. Improving operating room efficiency by applying bin-packing and portfolio techniques to surgical case scheduling. *Anesthesia & Analgesia*, 105(3):707, 2007.

[12] J. van Oostrum, M. Van Houdenhoven, J. Hurink, E. Hans, G. Wullink, and G. Kazemier. A master surgical scheduling approach for cyclic scheduling in operating room departments. *OR spectrum*, 30(2):355–374, 2008.

A Integer programming formulation

We will now fully specify the integer programming model used to modify the surgical schedule. To start, we specify the notation used. Important sets referenced by the model are:

$$
\begin{aligned}
W &: \text{days in the week } (0, \ldots, 6) \\
N &: \text{weeks in the month } (1, \ldots, 5) \\
O &: \text{ORs} \\
B &: \text{OR-day pairs corresponding to surgical blocks} \\
P &: \text{pairs of blocks corresponding to allowed moves} \\
F &: \text{surgical floor units} \\
S &: \text{surgeons} \\
V &: \text{services} \\
D_s &: \text{set of block-week pairs owned by surgeon } s, \text{ for } s \in S
\end{aligned}
$$

The parameters are:

$$
\begin{aligned}
cases_{fbi} &: \text{number of patients expected to occupy a bed on floor } f, \\
&\quad \text{having surgery } i \text{ days ago (mod 7) in block } b, \ i \in \{0, \ldots, 6\} \\
peak_f &: \text{the current peak bed census for floor } f \\
alloc_{bv} &: \text{number of weeks service } v \text{ owns block } b \\
\ell_v &: \text{lower bound on number of blocks allocated to service } v \text{ each day} \\
u_v &: \text{upper bound on number of blocks allocated to service } v \text{ each day}
\end{aligned}
$$

The decision variables are:

$$
x_p = \begin{cases} 1, & \text{if block move corresponding to pair } p \text{ occurs} \\ 0, & \text{otherwise} \end{cases}
$$

$$
z_f = \text{peak bed census on floor } f \text{ resulting from permuted schedule}
$$

The integer program is then given by:

$$\max \sum_{f \in F} (peak_f - z_f) \tag{1}$$

$$\text{s.t.} \sum_{\substack{b_1 \in B: \\ (b_1,b_2) \in P}} x_{b_1 b_2} = 1 \qquad \forall b_2 \in B \tag{2}$$

$$\sum_{\substack{(b_1,b_2) \in P: \\ b_2=(o_2,d_2), \\ i=d-d_2 \mod 7}} cases_{f b_1 i} \cdot x_{b_1 b_2} \leq z_f \qquad \forall d \in W, f \in F \tag{3}$$

$$\sum_{\substack{(b_1,b_2) \in P: \\ b_2=(o_2,d), \\ (b_1,i) \in D_s}} x_{b_1 b_2} \leq 1 \qquad \forall d \in W, i \in N, s \in S \tag{4}$$

$$\ell_v \leq \sum_{\substack{(b_1,b_2) \in P: \\ b_2=(o_2,d)}} alloc_{b_1 v} \cdot x_{b_1 b_2} \leq u_v \qquad \forall d \in W, v \in V \tag{5}$$

Here the objective (1) maximizes the reduction in peak bed census across the floors. Since we are summing the reductions, we will avoid improving the situation in one floor unit at the expense of another. Constraint (2) ensures that we will end up with an actual permutation of the schedule. In constraint (3) we link the z_f decision variables so that they are at least as great as the census on each day, and thus correspond to the peak census of each floor. The expression on the left-hand side calculates the expected census by finding the contribution of each block on its new day to the day in question. We need to make sure that we do not overbook the surgeons, which is handled in constraint (4). Note that the same surgeon is not necessarily working in an OR each week of the month, which is why it is necessary to go down to the level of day of week and week of month. Finally we need to have each service have a relatively balanced access to ORs throughout the week, which is handled by constraint (5), with specified upper and lower bounds for the number of blocks. There are many additional minor constraints that are used to handle situations like surgeon availability and special linked blocks, but these are omitted here for the sake of clarity.