# A Two-Stage Stochastic Integer Programming Approach to Integrated Staffing and Scheduling with Application to Nurse Management

Kibaek Kim, Sanjay Mehrotra
Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL 60201
kibaek.kim@u.northwestern.edu, mehrotra@northwestern.edu

We study the problem of integrated staffing and scheduling under demand uncertainty. This problem is formulated as a two-stage stochastic integer program with mixed-integer recourse. The here-and-now decision is to find initial staffing levels and schedules. The wait-and-see decision is to adjust these schedules at a time closer to the actual date of demand realization. We show that the mixed-integer rounding inequalities for the second-stage problem convexify the recourse function. As a result, we present a tight formulation that describes the convex hull of feasible solutions in the second stage. We develop a modified multicut approach in an integer L-shaped algorithm with a prioritized branching strategy. We generate twenty instances (each with more than 1.3 million integer and 4 billion continuous variables) of the staffing and scheduling problem using 3.5 years of patient volume data from Northwestern Memorial Hospital. Computational results show that the efficiency gained from the convexification of the recourse function is further enhanced by our modifications to the L-shaped method. The results also show that compared with a deterministic model, the two-stage stochastic model leads to a significant cost savings. The cost savings increase with mean absolute percentage errors in the patient volume forecast.

*History*: Revised on February 5, 2015

## 1. Introduction

Because labor cost constitutes a large portion of the operating expense of many organizations, staffing and scheduling decisions are important for effective operations management. The labor cost and job satisfaction both can be improved by improving these decisions (Wright and Mahar 2013). The staffing decision consists of knowing the number of employees that should be available to work at a given time. The scheduling decision consists of creating a detailed schedule plan that is offered to the employees. The actual requirements (demand) are typically unknown at the time of making these decisions. Short-term adjustments are made to ensure the desired quality of service. For example, in the healthcare setting, such adjustments may involve calling in extra workers (nurses) and paying overtime if demand surges or canceling the shift of a scheduled staff if demand drops (Bard and Purnomo 2005b,d).

In this paper, we study an integrated staffing and scheduling model (iStaff) that develops staffing plans and initial schedules while allowing adjustment to these schedules at a time closer to the actual demand realization. Our approach aims to reduce overall labor costs by right-sizing staff by balancing under- and overstaffing costs. Scheduling plans and staffing decisions are usually generated well ahead in time, while adjustments are made when more accurate demand information is available. The iStaff model is a challenging large-scale two-stage stochastic integer program with mixed-integer recourse. Such problems have been approached only heuristically in the past (e.g., Easton and Rossin 1996, Easton and Mansour 1999, Bard et al. 2007). The problem is large because the scheduling decisions introduce a large number of integer variables due to possible shift combinations. It is also a two-stage stochastic program because at a distant future adjustments to scheduling decisions are needed. More specifically, in our iStaff model (see Section 3) the here-and-now decision is to generate staffing levels and schedules, and the wait-and-see decision is to adjust these schedules for each demand scenario.

**Contributions of this paper.** This paper makes the following contributions. We identify valid parametric mixed-integer rounding (MIR) inequalities and show that by adding these inequalities we obtain the convex hull of feasible solutions for the second-stage mixed-integer problems in the iStaff model. This approach allows us to relax the integrality requirements on the second-stage variables of the iStaff model. An empirical study based on patient volume data from the Department of Hospital Medicine (HM) at Northwestern Memorial Hospital (NMH) is performed to evaluate the usefulness of the two-stage stochastic programming approach. A 1000-scenario iStaff model for HM in its extensive form has more than 1.3 million general integer variables and more than 4 billion continuous variables. The ability to convexify the second stage problems reduces the number of integer variables to 3,913. Despite this reduction, however, the extensive form could not be solved by the CPLEX MIP solver, and standard implementation of the integer L-shaped method needed further computational enhancements to achieve realistic solution times. We present two enhancements to the standard integer L-shaped method that significantly improve the computational performance for our problems. The first enhancement is in the form of a multicut aggregation approach. This enhancement results in a nearly sixfold reduction in the CPU time compared with the best known approaches. Furthermore, the memory requirement for solving the problems was significantly reduced. The second computational enhancement is in the form of preidentifying thin branching directions and using a priority branching on these directions before branching on the decision variables in the original optimization model. This enhancement results in a nearly threefold improvement in the CPU time during the branch-and-cut phase of the L-shaped method.

The value of the stochastic solution was evaluated. Empirical results show that in a high volume unit such as HM with a mean absolute percentage error in patient volume at around 12% (real setting) and with a cost structure where the overtime is 1.5 times the required wages and there is no salvage value for overstaffing, the two-stage stochastic programming approach yields about 2.3% cost savings. This amounts to an overall cost reduction of more than 3.6 times the hiring cost of full-time nurses. A 2.3% cost savings is significant for a unit that has an operating profit of 10% and incurs nearly 50% of its cost toward staffing. Empirical results under different cost and forecast error conditions are also presented. These results show cost savings ranging from less than 1% to as high as 34%. Empirical results are also given to demonstrate the improvement in solution quality when solving a 1000-scenario problem over a 100 scenario problem.

**Organization of this paper.** This paper is organized as follows. In Section 2.1 we review the relevant literature on the staffing and scheduing problem, with an emphasis on nurse scheduling. Section 2.2 reviews the literature on two-stage stochastic integer programming problems where both the first- and second-stage decision variables are mixed-integer. In Section 3 we formally introduce the iStaff model and study the properties of its second-stage recourse function. Specifically, we identify valid parametric mixed-integer rounding inequalities and show that the addition of these inequalities results in the convex hull of feasible solutions of the second-stage integer programs. We use this property to present an equivalent tight formulation for our problem by relaxing the integrality requirement of the second-stage decision variables. We present our computational enhancements to the integer L-shaped method in Section 4.

We test the iStaff model for application in a realistic environment, using real-world data, in Section 5. The generation of problem instances is described in Section 5.1. The results and analyses on the value of the stochastic solution for our problems are discussed in Section 5.2, and the expected value of perfect information is discussed in Section 5.3. Section 5.4 provides computational performance results with the multicut aggregation approach and the priority branching strategy. In Section 6, we give some concluding remarks and briefly discuss areas for future research. Appendix A gives CPLEX parameter settings. Appendix B gives additional information on patient volume characteristics and patient volume forecasting errors. Appendix C compares the value of stochastic solution in 100- and 1000-scenario problems. Appendix D gives the pseudocode for generating scheduling patterns. Appendix E gives the solution times for the deterministic staffing model that ignores forecasting errors and develops a plan based on point forecasts. Appendix F gives results on other cut aggregation strategies, and Appendix G gives results from an alternative multicut purge approach. Appendix H gives detailed results for the branching strategy that does not include the auxiliary variables to facilitate thin direction branching and therefore branches on the original variables only.

## 2. Literature Review

### 2.1. Literature review of nurse staffing and scheduling

Cheang et al. (2003) and Burke et al. (2004) provide an extensive review of models and methods for the nurse scheduling problem. Hence, the following review focuses primarily on the research that is directly related to our work. Most research studies on staffing, scheduling, and adjustment decision problems (e.g., Bard and Purnomo 2004, 2005a,b,c,d, 2007, Burke et al. 2012, Jaumard et al. 1998, Parr and Thompson 2007, Wright et al. 2006) have focused on only one aspect of these decisions. Only a few studies have attempted to integrate staffing and scheduling decisions and examine the implications of interactions between these decisions. An early paper by Abernathy et al. (1973) presents an integrated staffing and scheduling model to determine an optimal staffing policy with the recourse decisions of staff allocations under demand uncertainty. Abernathy et al. (1973) present two solution procedures to determine the staffing level: the first approach iteratively uses a penalty function for understaffing and overstaffing, whereas the second approach determines a required staffing level based on the chance-constraints. The models presented by Venkataraman and Brusco (1996), Easton and Rossin (1996), and Easton and Mansour (1999) are special cases of our iStaff model. Easton and Rossin (1996) present a stochastic goal programming model that integrates staffing and scheduling decisions under uncertain staffing requirements in a general workforce planning setting. A set of scheduling patterns is enumerated in advance, and the model determines the number of employees assigned to scheduling patterns. Tabu search (Easton and Rossin 1996) and a distributed genetic algorithm (Easton and Mansour 1999) are used to find heuristic solutions for the stochastic goal programming model. Maenhout and Vanhoucke (2013a,b) use a Dantzig-Wolfe decomposition approach to integrate nurse staffing and scheduling decisions in a deterministic setting.

Bard and Purnomo (2004, 2005b,d) consider the problem of short-term nurse rescheduling for daily fluctuations in patient demand, where a given midterm schedule is revised to cover shortages for nursing services. Wright and Bretthauer (2010) study a coordinated nurse planning problem that considers nurse scheduling and adjustments in response to patient demand. However, they solve the nurse scheduling model and the adjustment model separately. Woodall et al. (2013) use separate optimization models for monthly, weekly, and daily scheduling in a simulation framework.

A handful of studies take a two-stage stochastic programming approach for workforce planning. Kao and Queyranne (1985) present a two-stage stochastic program that determines staffing hours in the first stage and overtime in the second stage. Punnakitikashem et al. (2008) present a two-stage stochastic integer programming model for nurse assignment, where the first-stage decision assigns each nurse to patients and the second stage balances the workload for each nurse. The staffing decisions are integrated into the stochastic model by introducing binary variables (Punnakitikashem et al. 2013). An L-shaped method is used to solve both models (Punnakitikashem et al. 2008, 2013). Zhu and Sherali (2007) also present a two-stage stochastic workforce planning model, in which the second-stage decision assigns continuous workload to each worker. In a recent paper, Bodur and Luedtke (2014) present an integrated staffing and scheduling model for service system using two-stage stochastic programming. The second-stage decisions in their model are continuous variables, and a linear programming recourse problem is used to assign real-valued workload to the workers. The previous studies do not integrate workforce adjustment decisions in staffing and/or scheduling models as a recourse to the changed demand.

### 2.2. Literature review of two-stage stochastic programming with mixed-integer recourse

Two-stage stochastic integer programming with mixed-integer recourse is a challenging problem. Louveaux and Schultz (2003) and Sen (2005) give a survey of this problem. Most studies are limited to developing algorithms for problems when the second stage consists of mixed-binary programs (e.g., Carøe and Tind 1997, Gade et al. 2012, Laporte and Louveaux 1993, Sen and Higle 2005, Sherali and Fraticelli 2002, Sherali and Zhu 2006). Only a limited number of research papers focus on pure-integer variables (Ahmed et al. 2004, Kong et al. 2006, Schultz et al. 1998) or mixed-integer variables (Sen and Higle 2005, Sen and Sherali 2006) in the second stage.

A popular approach to solving two-stage stochastic programs is the L-shaped method based on Benders' decomposition (Benders 1962). Van Slyke and Wets (1969) first proposed the L-shaped method for two-stage stochastic linear programs with continuous variables in both stages. The integer L-shaped method, which is also based on the Benders' decomposition approach, was first proposed by Laporte and Louveaux (1993). The integer L-shaped method allows integer variables in the first-stage and/or the second-stage problems. It incorporates a branch-and-bound procedure to ensure optimality. Finiteness of this method is ensured from the finite number of subspaces that are created during branching (Birge and Louveaux 1997). Ahmed et al. (2004) proposed the idea of branching on tender variables defined by the product of first-stage decision variables with the technology matrix for problems having pure-integer variables in the second stage.

A typical description of the L-shaped method is based on a single-cut approach. In this approach a single optimality cut resulting from aggregating information from all the second-stage problems is added at each major iteration. Birge and Louveaux (1988) suggest a multicut approach where cut information from second-stage scenarios is kept separately. They suggest that this approach may significantly reduce the number of major iterations while solving the two-stage stochastic linear program. Moreover, Birge and Louveaux (1988) and Trukhanov et al. (2010) suggest that aggregation of subsets of scenarios to form a smaller number of cuts may be advantageous. These authors also explore the relative advantages of different scenario aggregation approaches. However, their studies are limited to solving a two-stage stochastic linear program. To our knowledge, no study has focused on efficient approaches to scenario aggregation and outer linearization in two-stage stochastic integer programs.

In a recent paper Kong et al. (2013) present several general conditions for totally unimodular property in two-stage stochastic mixed-integer programs. The model studied in our paper does not satisfy the totally unimodularity conditions on the constraint matrix. Furthermore, it does not have an integer right-hand side. However, we do use the total unimodular property of the second-stage constraint matrix when studying the properties of the second-stage polyhedra that is obtained after adding the parametric mixed-integer rounding inequalities.

## 3. Integrated Staffing and Scheduling Model

In this section we present iStaff, an integrated staffing and scheduling model used to find initial optimal staffing levels and schedules, while adjusting them in the future as more accurate demand information becomes available. The model formulation is given in Section 3.1. The properties of the second-stage mixed-integer recourse function are studied in Section 3.2. Specifically, a convex hull of the second-stage feasible solutions is obtained by adding a linear number of valid parametric mixed-integer rounding inequalities to our problem.

### 3.1. Model description and formulation

We consider an 18-week planning horizon to illustrate the decision dynamics of the iStaff model. The schedule is created for a 12-week period. We assume that the scheduling patterns repeat from week to week during this 12-week period. The staffing and scheduling decisions are made six weeks in advance of this 12-week horizon. The 18-week time horizon is chosen because in a realistic healthcare setting 12-week (roughly 3-month) schedules are generated and the schedules are made available to nurses six weeks in advance in order to allow for choices. Weekly decisions are made over the 12-week period to adjust the planned schedules at the beginning of each week for the following week. These adjustment (recourse) decisions are applied for each day of the week and allow for calling in additional staff or finding salvage value from the scheduled staff. Figure 1 shows different time horizons and the corresponding decisions over an 18-week planning horizon. For example, in Figure 1 the adjustments to week 10 schedules are made at the beginning of week 9, when a more accurate demand forecast becomes available. Note that an alternative model may allow daily adjustments 24 hours prior to the actual demand realization.

At the beginning of the planning horizon, the staffing and scheduling decisions are made in order to minimize the sum of total staffing cost, expected adjustment cost, and expected overstaffing

**Figure 1** Illustration of planning horizon, staffing horizon, adjustment horizon, and decision epoch

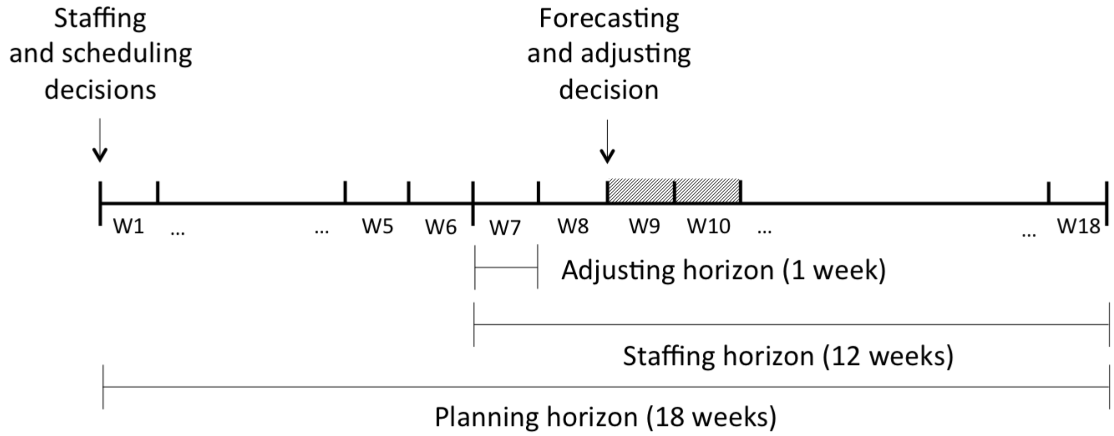



**Table 1** Notation for the iStaff model formulation

| | |
|---|---|
| The first-stage problem: | |
| Parameters | |
| $I$ | set of weekly scheduling patterns |
| $T$ | set of hourly time periods during a week ($=\{1,\ldots,24\times 7\}$) |
| $X$ | set of staffing and scheduling rules |
| $c_i$ | staffing cost per labor working for scheduling pattern $i$ |
| $a_{it}$ | 1 if scheduling pattern $i$ contains hour $t$, and 0 otherwise |
| Decision variables | |
| $x_i$ | decision variable representing the number of workers working in scheduling pattern $i \in I$ |
| $\chi_t$ | decision variable representing the number of workers working at time $t$ |
| The second-stage problem: | |
| Parameters | |
| $J$ | set of adjustment patterns |
| $q_j^+$ | cost of adding a shift for adjustment pattern $j$ |
| $q_j^-$ | cost of canceling a shift scheduled for pattern $j$ |
| $r^+$ | penalty cost of overstaffing per time period |
| $r^-$ | penalty cost of understaffing per time period |
| $w_{jt}$ | 1 if adjustment pattern $j$ contains hour $t$, and 0 otherwise |
| $d_t(\omega)$ | demand forecast at time period $t$ |
| Decision variables | |
| $y_j^+(\omega)$ | number of shifts added for adjustment pattern $j$ |
| $y_j^-(\omega)$ | number of shifts cancelled for adjustment pattern $j$ |
| $u_t(\omega)$ | amount of overstaffing at hour $t$ |
| $v_t(\omega)$ | amount of understaffing at hour $t$ |

and understaffing cost. Hence, the model is in the framework of the classical newsvendor model (see Davis et al. (2013) and references therein) generalized for the staffing problem. The staffing, scheduling, and adjustment decisions are coupled because an understaffed shift requires additional workers in order to maintain the desired quality of service, while an overstaffed shift results in lost wages because of limited salvage value of the scheduled staff.

We formulate the iStaff model as a two-stage stochastic integer program with mixed-integer recourse. The notation is described in Table 1. Each element $i$ in the set $I$ of weekly scheduling patterns defines one or more blocks of work hours with shift start times during a week, whereas each element $j$ in the set $J$ of adjustment patterns defines only one block of work hours a week. We also consider positive staffing cost $c_i > 0$ for scheduling pattern $i$, positive adjustment cost $q_j^+ > 0$ of adding a shift for adjustment pattern $j$, and non-negative adjustment cost $q_j^- \geq 0$ of canceling a shift scheduled for pattern $j$. Let $x_i$ denote the decision variables representing the number of workers working in scheduling pattern $i$, and let $\chi_t$ be the auxiliary variables representing the number of workers working at time $t$. The scheduling rules $X$ are used to generate the columns of the first-stage constraint matrix. We formulate the two-stage stochastic integer programming (TSSIP) formulation of the iStaff model as follows:

$$\min \quad \sum_{i \in I} c_i x_i + \mathcal{Q}(\boldsymbol{\chi}) \tag{TSSIP}$$

$$\text{s.t.} \quad \chi_t = \sum_{i \in I} a_{it} x_i \quad \forall t \in T \tag{1a}$$

$$\mathbf{x} = (x_1, \ldots, x_{|I|}) \in X \cap \mathbb{Z}^{|I|}, \tag{1b}$$

where the second-stage recourse function $\mathcal{Q}(\boldsymbol{\chi})$ evaluates the expected weekly recourse cost for a given $\boldsymbol{\chi}$ and the vector $\mathbf{a}_i = (a_{i1}, \ldots, a_{i|T|})$ represents an acceptable schedule pattern $i$ for the week. We assume that all acceptable schedule patterns are pregenerated, in order to ensure compliances with scheduling rules and regulations (no back-to-back shifts, weekly duty hours, etc.).

The objective function of (TSSIP) is to minimize the sum of the weekly staffing cost and the expected weekly recourse cost $\mathcal{Q}(\boldsymbol{\chi})$, $\boldsymbol{\chi} = (\chi_1, \ldots, \chi_{|T|})$. The decision variables $\chi_t$ in (1a) are also known as tender variables in the stochastic programming literature (Ahmed et al. 2004); note that they take integer value.

The expected recourse function $\mathcal{Q}(\boldsymbol{\chi}) := \mathbb{E}_\omega[Q(\boldsymbol{\chi}, \omega)]$, and

$$Q(\boldsymbol{\chi}, \omega) = \min \quad \sum_{j \in J} q_j^+ y_j^+(\omega) + \sum_{j \in J} q_j^- y_j^-(\omega) + r^+ \sum_{t \in T} u_t(\omega) + r^- \sum_{t \in T} v_t(\omega) \tag{2a}$$

$$\text{s.t.} \quad \sum_{j \in J} w_{jt}(y_j^+(\omega) - y_j^-(\omega)) - u_t(\omega) + v_t(\omega) = d_t(\omega) - \chi_t \quad \forall t \in T \tag{2b}$$

$$y_j^+(\omega), y_j^-(\omega) \in \mathbb{Z}_+, \; u_t(\omega), v_t(\omega) \geq 0 \quad \forall j \in J, \, t \in T, \tag{2c}$$

where $y_j^+(\omega)$ and $y_j^-(\omega)$ are the decision variables representing the number of shifts added and cancelled, respectively, for adjustment pattern $j$. The overstaffing and understaffing at each hour $t$ are captured by $u_t(\omega)$ and $v_t(\omega)$ at penalty costs $r^+$ and $r^-$, respectively. The vector $\mathbf{w}_j = (w_{j1}, \ldots, w_{j|T|})$ in (2b) represents a pregenerated adjustment pattern for the week. Constraint (2b) ensures that the anticipated demand scenarios $d_t(\omega)$ are satisfied by the staffing levels after adjustments.

### 3.2. Properties of the second-stage problem in the iStaff model

Our main result is given in Theorem 1, which shows that a tight formulation is possible for the second-stage mixed-integer problem after adding certain parametric mixed-integer rounding (MIR) inequalities. Consequently, the integrality requirement of the second-stage variables can be relaxed. This approach allows us to compute a subgradient of the recourse function.

We omit the indexing for the scenario when discussing the second-stage problem in this section. Let $\mathbf{q}^+, \mathbf{q}^- \in \mathbb{R}^{|J|}$, and $\mathbf{d} \in \mathbb{R}^{|T|}$. Let $\mathbf{e} \in \mathbb{R}^{|T|}$ be the vector of all ones, and let $\mathbf{W} \in \mathbb{B}^{|J|\times|T|}$ be a $|J| \times |T|$-dimensional matrix of elements $w_{jt}$. The recourse function using matrix and vector notation is given as follows:

$$\begin{aligned} Q(\boldsymbol{\chi}, \omega) = \min \quad & (\mathbf{q}^+)^T \mathbf{y}^+ + (\mathbf{q}^-)^T \mathbf{y}^- + r^+ \mathbf{e}^T \mathbf{u} + r^- \mathbf{e}^T \mathbf{v} \\ \text{s.t.} \quad & \mathbf{W}^T (\mathbf{y}^+ - \mathbf{y}^-) - \mathbf{u} + \mathbf{v} = \mathbf{d} - \boldsymbol{\chi} \\ & \mathbf{y}^+, \mathbf{y}^- \in \mathbb{Z}_+^{|J|}, \mathbf{u}, \mathbf{v} \in \mathbb{R}_+^{|T|}. \end{aligned}$$

By eliminating $\mathbf{v}$ the second-stage problem is given by

$$\min \quad (\tilde{\mathbf{q}}^+)^T \mathbf{y}^+ + (\tilde{\mathbf{q}}^-)^T \mathbf{y}^- + (r^+ + r^-) \mathbf{e}^T \mathbf{u} + \tilde{r} \tag{3a}$$

$$\text{s.t.} \quad (\mathbf{y}^+, \mathbf{y}^-, \mathbf{u}) \in \mathcal{P}(\boldsymbol{\chi}), \tag{3b}$$

where $\tilde{\mathbf{q}}^+ = \mathbf{q}^+ - r^- \mathbf{W}\mathbf{e}$, $\tilde{\mathbf{q}}^- = \mathbf{q}^- + r^- \mathbf{W}\mathbf{e}$, $\tilde{r} = r^- \mathbf{e}^T (\mathbf{d} - \boldsymbol{\chi})$, and $\mathcal{P}(\boldsymbol{\chi})$ is described by

$$\mathcal{P}(\boldsymbol{\chi}) := \left\{ (\mathbf{y}^+, \mathbf{y}^-, \mathbf{u}) \in \mathbb{Z}_+^{|J|} \times \mathbb{Z}_+^{|J|} \times \mathbb{R}_+^{|T|} \mid \mathbf{w}_t^T (\mathbf{y}^+ - \mathbf{y}^-) - u_t \leq d_t - \chi_t, \ \forall t \in T \right\}. \tag{4}$$

Observe that, since the second-stage problem $Q(\boldsymbol{\chi}, \omega)$ can be reformulated as a piecewise linear convex optimization problem over non-negative integer decision variables by substituting $u_t = \max\{0, \mathbf{w}_t^T (\mathbf{y}^+ - \mathbf{y}^-) + \chi_t - d_t\}$ for all $t \in T$, the iStaff model has complete recourse. We now show the validity of parametric MIR inequalities for (4).

Proposition 1. *The parametric MIR inequalities*

$$\mathbf{w}_t^T (\mathbf{y}^+ - \mathbf{y}^-) - f_t u_t \leq \lfloor d_t \rfloor - \chi_t \quad \forall t \in T, \tag{5}$$

*where $f_t = 1/(1 - d_t + \lfloor d_t \rfloor)$, are valid for $\mathcal{P}(\boldsymbol{\chi})$.*

*Proof.* For a given $t$, we consider two cases.

- Case 1: Suppose $f_t u_t < 1$. Then, $\mathbf{w}_t^T (\mathbf{y}^+ - \mathbf{y}^-) \leq d_t - \chi_t + u_t < \lfloor d_t \rfloor + 1 - \chi_t$. Since $\mathbf{w}_t^T (\mathbf{y}^+ - \mathbf{y}^-)$ and $\boldsymbol{\chi}$ are integral, we have $\mathbf{w}_t^T (\mathbf{y}^+ - \mathbf{y}^-) \leq \lfloor d_t \rfloor - \chi_t$. Subtracting $f_t u_t \geq 0$ gives (5).
- Case 2: Suppose $f_t u_t \geq 1$. Then, $\mathbf{w}_t^T (\mathbf{y}^+ - \mathbf{y}^-) - f_t u_t \leq d_t - \chi_t + u_t - f_t u_t \leq d_t - \chi_t + 1/f_t - 1 = \lfloor d_t \rfloor - \chi_t$. Therefore, the MIR inequalities are valid for $\mathcal{P}(\boldsymbol{\chi})$. $\square$

Note that the valid inequalities (5) are *parameterized* by $\boldsymbol{\chi}$, whose value is not fixed since these are the tender variables in (TSSIP). The main observation leading to the validity of (5) is that in our case $\chi_t$ is integer, giving $\lfloor d_t - \chi_t \rfloor = \lfloor d_t \rfloor - \chi_t$.

THEOREM 1. *Let $\mathcal{P}(\boldsymbol{\chi})$ be given in (4). Then, conv($\mathcal{P}(\boldsymbol{\chi})$) is described by inequalities*

$$\begin{aligned}
&\mathbf{w}_t^T(\mathbf{y}^+ - \mathbf{y}^-) - u_t \le d_t - \chi_t, \quad \forall t \in T\\
&\mathbf{w}_t^T(\mathbf{y}^+ - \mathbf{y}^-) - f_t u_t \le \lfloor d_t \rfloor - \chi_t, \quad \forall t \in T\\
&\mathbf{y}^+, \mathbf{y}^-, \mathbf{u} \ge 0.
\end{aligned}$$

*Proof.* Our proof here borrows the conceptual steps from Miller and Wolsey (2003). Let $\eta_t = \mathbf{w}_t^T(\mathbf{y}^+ - \mathbf{y}^-) + \chi_t$ for all $t \in T$, and consider the set $\mathcal{S}_t = \{(\eta_t, u_t) \mid \eta_t - u_t \le d_t,\ \eta_t - f_t u_t \le \lfloor d_t \rfloor,\ u_t \ge 0\}$. Note that at any extreme point of $\mathcal{S}_t$ only two of the inequalities in the set $\mathcal{S}_t$ are binding. The extreme points of $\mathcal{S}_t$ are given by $(1 + \lfloor d_t \rfloor, 1 - d_t + \lfloor d_t \rfloor)$ and $(\lfloor d_t \rfloor, 0)$ for all $t \in T$. These extreme points are integral in $\eta_t$.

Now consider the set $\mathcal{U} = \{(\boldsymbol{\eta}, \mathbf{y}^+, \mathbf{y}^-, \mathbf{u}) \mid \boldsymbol{\eta} = \mathbf{W}^T(\mathbf{y}^+ - \mathbf{y}^-) + \boldsymbol{\chi},\ \mathbf{y}^+, \mathbf{y}^- \ge 0,\ (\eta_t, u_t) \in \mathcal{S}_t,\ \forall t \in T\}$, and $\mathcal{U} \subseteq \mathbb{R}^{2(|J|+|T|)}$, where $\mathbf{W}$ is defined as a $|J| \times |T|$-dimensional matrix of elements $w_{jt}$. Suppose that $(\hat{\boldsymbol{\eta}}, \hat{\mathbf{y}}^+, \hat{\mathbf{y}}^-, \hat{\mathbf{u}})$ is an extreme point of the polyhedra $\mathcal{U}$. We will show that $(\hat{\boldsymbol{\eta}}, \hat{\mathbf{y}}^+, \hat{\mathbf{y}}^-, \hat{\mathbf{u}})$ is given by the following system of linear equations,

$$\begin{bmatrix} \boldsymbol{\chi} \\ \boldsymbol{\eta}_{\text{LB}} \end{bmatrix} \le \begin{bmatrix} \mathbf{I} & -\mathbf{W}^T & \mathbf{W}^T \\ \mathbf{I} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\eta}} \\ \hat{\mathbf{y}}^+ \\ \hat{\mathbf{y}}^- \end{bmatrix} \le \begin{bmatrix} \boldsymbol{\chi} \\ \boldsymbol{\eta}_{\text{UB}} \end{bmatrix} \tag{6a}$$

$$\hat{\mathbf{u}} = \mathbf{A}^T \hat{\boldsymbol{\eta}} - \mathbf{b}, \tag{6b}$$

where $\boldsymbol{\eta}_{\text{LB}}, \boldsymbol{\eta}_{\text{UB}} \in \mathbb{Z}^{|T|} \cup \{-\infty, \infty\}$, $\mathbf{A} \in \mathbb{R}^{|T| \times |T|}$, and $\mathbf{b} \in \mathbb{R}^{|T|}$. Furthermore, we will show that $(\hat{\boldsymbol{\eta}}, \hat{\mathbf{y}}^+, \hat{\mathbf{y}}^-, \hat{\mathbf{u}})$ is integral in $(\hat{\boldsymbol{\eta}}, \hat{\mathbf{y}}^+, \hat{\mathbf{y}}^-)$.

Note that in $\mathcal{U}$ an extreme point is given by $2(|J| + |T|)$ binding constraints with the matrix defining the constraints to be nonsingular. Let $J_=^+ = \{j \in J \mid \hat{y}_j^+ = 0\}$ and $J_=^- = \{j \in J \mid \hat{y}_j^- = 0\}$, and let $J_>^+ = J \backslash J_=^+$ and $J_>^- = J \backslash J_=^-$. Hence, at this extreme point, at least $|T| + |J_>^+| + |J_>^-|$ binding constraints exist. The binding constraints defining this set include $\boldsymbol{\eta} = \mathbf{W}^T(\mathbf{y}^+ - \mathbf{y}^-) + \boldsymbol{\chi}$. Note that for each set $\mathcal{S}_t$ at least one constraint defining this set should be binding at any extreme point of $\mathcal{U}$. Hence, for at least $|J_>^+| + |J_>^-|$ of the sets $\mathcal{S}_t$, two constraints are binding at the extreme point $(\hat{\boldsymbol{\eta}}, \hat{\mathbf{y}}^+, \hat{\mathbf{y}}^-, \hat{\mathbf{u}})$. The remaining sets $\mathcal{S}_t$ have one binding constraint. The extreme points satisfy

$$\begin{aligned}
\hat{u}_t &= \hat{\eta}_t - d_t && \text{if } \hat{\eta}_t \ge 1 + \lfloor d_t \rfloor,\\
\hat{u}_t &= (1 - d_t + \lfloor d_t \rfloor)(\hat{\eta}_t - \lfloor d_t \rfloor) && \text{if } \lfloor d_t \rfloor \le \hat{\eta}_t \le 1 + \lfloor d_t \rfloor,\\
\hat{u}_t &= 0 && \text{if } \hat{\eta}_t \le \lfloor d_t \rfloor.
\end{aligned}$$

Let $T_1$ be a subset of $T$ such that $\mathcal{S}_t$ has one binding constraint at the extreme point so that $|T\backslash T_1| \geq |J^+_>| + |J^-_>|$. Note that $|T\backslash T_1| > |J^+_>| + |J^-_>|$ when the extreme point is degenerate. Then, the following constraints in $\mathcal{U}$ define the extreme point $(\hat{\boldsymbol{\eta}}, \hat{\mathbf{y}}^+, \hat{\mathbf{y}}^-, \hat{\mathbf{u}})$:

$$\hat{\eta}_t = \sum_{j\in J^+_=} w^T_{jt}\hat{y}^+_j + \sum_{j\in J^+_>} w^T_{jt}\hat{y}^+_j - \sum_{j\in J^-_=} w^T_{jt}\hat{y}^-_j - \sum_{j\in J^-_>} w^T_{jt}\hat{y}^-_j + \chi_t, \quad \forall t\in T \tag{7a}$$

$$\hat{y}^+_j = 0, \quad \forall j\in J^+_=, \qquad \hat{y}^+_j > 0, \quad \forall j\in J^+_>, \qquad \hat{y}^-_j = 0, \quad \forall j\in J^-_=, \qquad \hat{y}^-_j > 0, \quad \forall j\in J^-_> \tag{7b}$$

$$\begin{bmatrix} \hat{u}_1 \\ \vdots \\ \hat{u}_{|T_1|} \\ \hline \hat{u}_{|T_1|+1} \\ \vdots \\ \hat{u}_{|T|} \end{bmatrix} = \left[\begin{array}{ccc|c} a_1 & & & \\ & \ddots & & 0 \\ & & a_{|T_1|} & \\ \hline & 0 & & 0 \end{array}\right] \begin{bmatrix} \hat{\eta}_1 \\ \vdots \\ \hat{\eta}_{|T_1|} \\ \hline \hat{\eta}_{|T_1|+1} \\ \vdots \\ \hat{\eta}_{|T|} \end{bmatrix} - \begin{bmatrix} b_1 \\ \vdots \\ b_{|T_1|} \\ \hline 0 \text{ or } 1-d_t+\lfloor d_t\rfloor \\ \vdots \\ 0 \text{ or } 1-d_t+\lfloor d_t\rfloor \end{bmatrix}$$
$$\text{for } (a_t,b_t)\in\{(1,d_t),(1-d_t+\lfloor d_t\rfloor,(1-d_t+\lfloor d_t\rfloor)\lfloor d_t\rfloor),(0,0)\} \text{ and } t\in T_1. \tag{7c}$$

Now we set each element of vector $\boldsymbol{\eta}_{\text{LB}}$ to be $1+\lfloor d_t\rfloor$, $\lfloor d_t\rfloor$, or $-\infty$ and set each element of vector $\boldsymbol{\eta}_{\text{UB}}$ to be $1+\lfloor d_t\rfloor$, $\lfloor d_t\rfloor$, or $\infty$. Also, we let $\mathbf{A}$ be a diagonal matrix where each diagonal element is 0, 1, or $1-d_t+\lfloor d_t\rfloor$, and we let each element of $\mathbf{b}$ be 0, $d_t$, $1-d_t+\lfloor d_t\rfloor$, or $(1-d_t+\lfloor d_t\rfloor)\lfloor d_t\rfloor$. Then, the set of constraints (7) is rewritten as (6). Moreover, each row of $\mathbf{W}$ has consecutive ones. Hence, $\mathbf{W}$ is totally unimodular, and so is $\begin{bmatrix} \mathbf{I} & -\mathbf{W}^T & \mathbf{W}^T \\ \mathbf{I} & \mathbf{0} & \mathbf{0} \end{bmatrix}$ (Nemhauser and Wolsey 1988). Note that $\boldsymbol{\chi}, \boldsymbol{\eta}_{\text{LB}}$ and $\boldsymbol{\eta}_{\text{UB}}$ are integer vectors, and so are $\hat{\boldsymbol{\eta}}, \hat{\mathbf{y}}^+$, and $\hat{\mathbf{y}}^-$. Therefore, an extreme point of $\mathcal{U}$ is integral in $\hat{\boldsymbol{\eta}}, \hat{\mathbf{y}}^+$, and $\hat{\mathbf{y}}^-$. By eliminating $\boldsymbol{\eta}$ in $\mathcal{U}$, we obtain $conv(\mathcal{P}(\boldsymbol{\chi}))$, where the extreme points are integral in $\hat{\mathbf{y}}^+$ and $\hat{\mathbf{y}}^-$. $\square$

Note that the first-stage constraint matrix in (TSSIP) may not be totally unimodular. The reason is that, because of required off-time, the consecutive ones property is violated when considering multiple shifts in a week while generating scheduling patterns in $X$. The consecutive ones property holds for the recourse constraint matrix, however, because the adjustments are for a shift during a week.

Theorem 1 implies that the integrality of the second-stage decision variables can be relaxed after adding the MIR inequalities. The following corollaries provide desirable properties from an algorithmic perspective.

We denote the "convexified" recourse function resulting from adding MIR inequalities (5) by $Q_{\text{MIR}}(\boldsymbol{\chi},\omega)$:

$$Q_{\text{MIR}}(\boldsymbol{\chi},\omega) = \min \quad (\tilde{\mathbf{q}}^+)^T\mathbf{y}^+ + (\tilde{\mathbf{q}}^-)^T\mathbf{y}^- + (r^+ + r^-)\mathbf{e}^T\mathbf{u} + \tilde{r} \tag{8a}$$
$$\text{s.t.} \quad \mathbf{w}^T_t(\mathbf{y}^+ - \mathbf{y}^-) - u_t \leq d_t - \chi_t, \quad \forall t\in T \tag{8b}$$
$$\mathbf{w}^T_t(\mathbf{y}^+ - \mathbf{y}^-) - f_t u_t \leq \lfloor d_t\rfloor - \chi_t, \quad \forall t\in T \tag{8c}$$
$$\mathbf{y}^+, \mathbf{y}^- \in \mathbb{Z}^{|J|}_+, \mathbf{u}\geq 0. \tag{8d}$$

COROLLARY 1. *For each $\omega \in \Omega$ and a given $\boldsymbol{\chi} \in \mathbb{Z}^{|T|}$, $Q_{MIR}(\boldsymbol{\chi},\omega) = Q(\boldsymbol{\chi},\omega)$.*

COROLLARY 2. *Assuming $\boldsymbol{\chi} \in \mathbb{Z}^{|T|}$, the function $\mathcal{Q}_{MIR}(\boldsymbol{\chi}) := \mathbb{E}_\omega[Q_{MIR}(\boldsymbol{\chi},\omega)]$ is piecewise linear convex in $\boldsymbol{\chi}$.*

COROLLARY 3. *For a given $\boldsymbol{\chi}^*$ and scenario $\omega$, let $\boldsymbol{\mu}^*(\omega)$ and $\boldsymbol{\pi}^*(\omega)$ be a dual optimal solution of (8). Then, $-\boldsymbol{\mu}^*(\omega) - \boldsymbol{\pi}^*(\omega) - r^-\mathbf{e}$ is a subgradient of $Q_{MIR}(\boldsymbol{\chi},\omega)$ at point $\boldsymbol{\chi}^*$. Moreover, the recourse function $Q(\boldsymbol{\chi},\omega)$ is underestimated by*

$$Q(\boldsymbol{\chi},\omega) \geq \left(\mathbf{d}(\omega)^T\boldsymbol{\mu}^*(\omega) + \lfloor \mathbf{d}(\omega)\rfloor^T\boldsymbol{\pi}^*(\omega) + r^-\mathbf{e}^T\mathbf{d}(\omega)\right) - \left(\boldsymbol{\mu}^*(\omega) + \boldsymbol{\pi}^*(\omega) + r^-\mathbf{e}\right)^T\boldsymbol{\chi}. \tag{9}$$

## 4. Modified L-Shaped Method for the iStaff Model

In this section we present a modified L-shaped method for solving the iStaff model. The modifications to the L-shaped method proposed here are intended to achieve computational improvement for iStaff model. We assume that the random vector $\omega$ follows a discrete distribution with a finite support. Let $S$ be a finite set of scenario indices, and let $p_s > 0$ be the probability of realizing scenario $\omega_s$ for $s \in S$ such that $\sum_{s\in S} p_s = 1$. Then, $\mathcal{Q}(\boldsymbol{\chi}) = \mathbb{E}_\omega[Q_{\mathrm{MIR}}(\boldsymbol{\chi},\omega)] = \sum_{s\in S} p_s Q_{\mathrm{MIR}}(\boldsymbol{\chi},\omega_s)$.

In Section 4.1, we present a branching strategy that prioritizes the order of branching variables in the model. Auxiliary branching variables are introduced to provide new branching directions for our branch-and-cut (B&C) procedure. In Section 4.2, we develop a multicut aggregation approach with the goal of avoiding an increase in the number of constraints in the B&C node subproblems. In Section 4.3, we summarize the modified L-shaped method with the proposed solution enhancements.

### 4.1. Thin direction branching strategy

We present a heuristic branching strategy that prioritizes the order of branching variables in the model. The proposed branching strategy implicitly performs branching on thin directions of the polyhedron. We first introduce a more detailed description of the staffing model. The iStaff model considers full-time, part-time, and casual employments, denoted by FT, PT, and CA, respectively, and six shift types, denoted by 3S12H, 2S12H, 1S12H, 3S8H, 2S8H, and 1S8H. The employment and shift types are defined in Table 2. A full-time staff works 36 hours per week, part-time staff works 24 hours per week, and casual staff works less than 24 hours per week. Casual staff fills in when additional staff is required. The casual staff (nurses) is called PRN (pro re nata) employment in hospitals and is part of the overall nurse pool.

Let $E$ be the set of employment types (full-time, part-time, and casual), and let $H$ be the set of shift types. We additionally introduce auxiliary variables $\phi_e, e \in E$ and $\psi_h, h \in H$ that represent the number of workers in each employment type $e$ and the number of workers in each shift

**Table 2** Definition of employment and shift types considered in the iStaff model

| Employment Type | Shift Type | Work Hours per Week |
|---|---|---|
| FT: Full-time | 3S12H: Three 12-hour shifts | 36 |
| PT: Part-time | 2S12H: Two 12-hour shifts | 24 |
| PT: Part-time | 3S8H: Three 8-hour shift | 24 |
| CA: Casual | 2S8H: Two 8-hour shift | 16 |
| CA: Casual | 1S12H: One 12-hour shift | 12 |
| CA: Casual | 1S8H: One 8-hour shift | 8 |

type $h$, respectively. We denote $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_{|E|})$ and $\boldsymbol{\psi} = (\psi_1, \ldots, \psi_{|H|})$. The following additional constraints are added to the model (TSSIP):

$$\sum_{i \in I_e} x_i = \phi_e \quad e \in E, \tag{10a}$$

$$\sum_{i \in I_h} x_i = \psi_h \quad h \in H, \tag{10b}$$

where $I_e$ and $I_h$ are the subsets of $I$ representing the scheduling patterns for employment type $e$ and shift type $h$, respectively. Theorem 1 remains applicable after adding these constraints to iStaff. We now consider the following convex mixed-integer programming formulation of (TSSIP):

$$\min \quad \sum_{i \in I} c_i x_i + \sum_{s \in S} p_s \theta_s \tag{CMIP}$$

$$\text{s.t.} \quad Q_{\text{MIR}}(\boldsymbol{\chi}, \omega_s) \leq \theta_s, \quad \forall s \in S, \tag{11a}$$

$$(\mathbf{x}, \boldsymbol{\chi}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\theta}) \in \bar{X} \cap \mathbb{Z}^{|I|} \times \mathbb{Z}^{|T|}, \tag{11b}$$

where $\bar{X} = \{(\mathbf{x}, \boldsymbol{\chi}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\theta}) \mid \text{(1a), (10a), (10b)}, \mathbf{x} \in X\}$ and $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_{|S|})$. Note that the objective function of (TSSIP) is reformulated by the objective function of (CMIP) and the convex constraints (11a). In the L-shaped method, the convex constraints (11a) are outer approximated by linear inequalities (9) in Corollary 3.

The branching priority order for the first-stage variables is defined such that first $\boldsymbol{\phi}$ is considered for branching, followed by $\boldsymbol{\psi}, \boldsymbol{\chi}$, and $\mathbf{x}$. If multiple fractional variables exist with the same branching priority, then branching is performed on the most fractional variable. This branching order can be viewed as a heuristic from the following standpoints. First note that branching on thin directions can be beneficial for solving mixed-integer programming problems as suggested by the theoretical results of Lenstra (1983), Lovász and Scarf (1992), and Mehrotra and Li (2011). More specifically, a polynomial time algorithm is available for mixed-integer linear programs and mixed-integer convex programs in fixed dimensions. Unfortunately, generation of thin branching directions by following the theory can be computationally expensive, since it requires a certain lattice basis reduction. Other strategies have been proposed to identify good thin directions adaptively while simplifying this computation (Aardal et al. 2002, Owen and Mehrotra 2001, Mahajan and Ralphs 2009, Karamanov and Cornuéjols 2011, Cornuéjols et al. 2011, Mehrotra and Huang 2013). Our proposed thin direction branching scheme can be viewed as preidentifying thin branching directions represented by (10). These directions are expected to be thin in cost-effective feasible solutions because $\phi_e$ and $\psi_h$ (the total number of full-time workers and the number of workers in a certain shift type) will have values only in a limited range to satisfy demand.

## 4.2. Multicut aggregation of outer linearization of node subproblems

We now present an outer linearization approach with multicut aggregation to solve (CMIP) in a B&C framework. The multicut aggregation proposed here heuristically aggregates the cuts by using the dual variable values of B&C node LP relaxation subproblem. The optimal value of dual multiplier corresponding to the outer linearization inequality in a B&C node subproblem is used. All the cuts with zero dual variable value are aggregated into a single cut. In an abuse of notation, we now describe this aggregation procedure in some detail and distinguish it from other approaches.

Let $k$ be the index for outer linearization iterations, where optimality cuts are added. Specifically, for a given value of tender variables $\boldsymbol{\chi}^k$, at iteration $k$ an outer linearization cut is generated for each scenario $s \in S$. These cuts are aggregated into $M^k$ cuts, one for each subset $S^k_m$. Let the set of scenarios be divided into mutually exclusive subsets $S^k_m$, $m = 1, \ldots M^k$, (i.e., $S = \cup_{m=1}^{M^k} S^k_m$ and $S^k_i \cap S^k_j = \emptyset$ for any $i \neq j$ and $i, j \in \{1, \ldots, M^k\}$). Let $\bar{X}^{\mathcal{N}} \subseteq \bar{X}$ be the set of feasible solutions at a given node $\mathcal{N}$ in the B&C tree $\mathcal{T}$. A B&C node subproblem after adding $l$ rounds of optimality cuts at node $\mathcal{N} \in \mathcal{T}$ is given by

$$
\begin{aligned}
\min \quad & \sum_{i \in I} c_i x_i + \sum_{s \in S} p_s \theta_s && (\mathrm{SP}^{\mathcal{N}}) \\
\text{s.t.} \quad & \sum_{s \in S^k_m} (\mathbf{G}^T_{ks} \boldsymbol{\chi} + \theta_s - \mathbf{g}_{ks}) \geq 0, \quad m = 1, \ldots, M^k,\ k = 1, \ldots, l, && (12a) \\
& (\mathbf{x}, \boldsymbol{\chi}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\theta}) \in \bar{X}^{\mathcal{N}}, && (12b)
\end{aligned}
$$

where equation (12a) gives the aggregated cuts generated at outer linearization iterations $k = 1, \ldots, l$. These cuts are denoted by using coefficient matrices $\mathbf{G}_{ks} := \boldsymbol{\mu}^k(\omega_s) + \boldsymbol{\pi}^k(\omega_s) + r^- \mathbf{e}$, right-hand side vectors $\mathbf{g}_{ks} := \mathbf{d}(\omega_s)^T \boldsymbol{\mu}^k(\omega_s) + \lfloor \mathbf{d}(\omega_s) \rfloor^T \boldsymbol{\pi}^k(\omega_s) + r^- \mathbf{e}^T \mathbf{d}(\omega_s)$ and use optimal dual variable values $\boldsymbol{\mu}^k(\omega_s)$ and $\boldsymbol{\pi}^k(\omega_s)$ corresponding to constraints (8b) and (8c). The convex constraints (11a) are now approximated by linear inequalities (12a) generated from the $k$th outer linearization for each scenario $s \in S$. Note that the inequalities (12a) are valid at any node $\mathcal{N} \in \mathcal{T}$.

The standard multicut approach adds up to $|S|$ cuts in each outer linearization iteration, whereas for a single-cut approach $M^1 = \cdots = M^l = 1$. A hybrid-cut approach sets $1 < M^k < |S|, M^1 = \cdots = M^l$ (Birge and Louveaux 1988). We note that the cut aggregation level can change from an outer linearization iteration to a subsequent one. Based on this observation, our approach is a variant of the hybrid-cut approach, where $S_m^k$ and $M^k$ may be different for each $k=1,\dots,l$. We call this approach multicut aggregation (MCA). The proposed MCA approach is described in Algorithm 1.

**Algorithm 1** Multicut Aggregation (MCA)

**Initialization.** For a given $\boldsymbol{\chi}^k$, outer linearization cuts are given by

$$\sum_{s\in S_m^k}(\mathbf{G}_{ks}^T\boldsymbol{\chi}+\theta_s-\mathbf{g}_{ks})\geq 0, \quad m=1,\dots,M^k. \tag{13}$$

**Step 1.** Add these cuts (13) to (SP$^{\mathcal{N}}$).

**Step 2.** Solve (SP$^{\mathcal{N}}$), and let $\boldsymbol{\lambda}_m^*$ be the optimal dual variable values corresponding to the cuts (13) for $m=1,\dots,M^k$.

**Step 3.** Let $m_0$ be the number of outer linearization cuts with the corresponding dual multipliers equal to zeros, say $\boldsymbol{\lambda}_m^k=0$ for $m=(M^k-m_0+1),\dots,M^k$. Generate the aggregated cut

$$\sum_{m=M^k-m_0+1}^{M^k}\sum_{s\in S_m^k}(\mathbf{G}_{ks}^T\boldsymbol{\chi}+\theta_s-\mathbf{g}_{ks})\geq 0. \tag{14}$$

**Step 4.** Update $M^k := M^k-m_0$.

Suppose that an integer L-shaped method applies a multicut approach (i.e., $M^k=|S|$ for all $k$) and that outer linearization cuts are generated for a given $\boldsymbol{\chi}^k$. In Algorithm 1, $S_m^k$ for $m=1,\dots,M^k$ are initialized by singletons of the set $S$. (SP$^{\mathcal{N}}$) is first solved with these cuts added according to $M^k$ and $S_m^k$ for $m=1,\dots,M^k$. Next, we aggregate the cuts that have the corresponding dual variable values equal to zeros at the optimum solution of the LP subproblem. The following proposition states that the optimum solution of the current relaxation problem does not change after aggregating the cuts.

PROPOSITION 2. *Suppose that outer linearization cuts* (12a) *are generated for a given* $\boldsymbol{\chi}^k$ *and that* $(\mathbf{x}^{k+1},\boldsymbol{\chi}^{k+1},\boldsymbol{\phi}^{k+1},\boldsymbol{\psi}^{k+1},\boldsymbol{\theta}^{k+1})$ *is an optimal solution of* (SP$^{\mathcal{N}}$) *with these cuts. Then,* $(\mathbf{x}^{k+1},\boldsymbol{\chi}^{k+1},\boldsymbol{\phi}^{k+1},\boldsymbol{\psi}^{k+1},\boldsymbol{\theta}^{k+1})$ *is an optimal solution of* (SP$^{\mathcal{N}}$) *after aggregating the cuts by Algorithm 1.* □

### 4.3. Modified L-shaped method

The modified integer L-shaped method with the thin direction branching branching strategy and the MCA approach is summarized in this section. Algorithm 2 provides the steps of the method.

**Algorithm 2** Modified Integer L-Shaped Method

**Initialization.** Create a root node $\mathcal{N}$ with $\bar{X}^{\mathcal{N}} := \bar{X}$, and set $k := 0$ and $\bar{z} := -\infty$. Set an initial value for $M^0$, and define $S_m^0$ for $m = 1, \ldots, M^0$.

**Step 1.** Convex relaxation programming problem:

**a.** Solve (SP$^{\mathcal{N}}$) to optimality. If the problem is infeasible, then stop. Otherwise, let $(\hat{\mathbf{x}}, \hat{\boldsymbol{\chi}}, \hat{\boldsymbol{\phi}}, \hat{\boldsymbol{\psi}}, \hat{\boldsymbol{\theta}})$ be an optimal solution.

**b.** If $\sum_{s \in S} p_s \hat{\theta}_s < \sum_{s \in S} p_s Q(\hat{\boldsymbol{\chi}}, \omega_s)$, then set $k := k+1$, $M^k := M^0$ and $S_m^k := S_m^0$ for $m = 1, \ldots, M^0$, add outer linearization cuts (13), and go to Step 1a.

**c.** If the current solution is fractional in $\hat{\mathbf{x}}, \hat{\boldsymbol{\chi}}, \hat{\boldsymbol{\phi}}$, or $\hat{\boldsymbol{\psi}}$, then aggregate the inequalities (13) as described in Algorithm 1, and go to Step 2. Otherwise, an optimal solution has been found: **stop**.

**Step 2.** Branch-and-cut procedure:

**a.** Select a node $\mathcal{N} \in \mathcal{T}$ according to a best-bound rule. If none exists (i.e., $\mathcal{T} = \emptyset$), then an optimal solution has been found: **stop**.

**b.** Solve (SP$^{\mathcal{N}}$), and let $z^{\mathcal{N}}$ be the optimal objective value. If $z^{\mathcal{N}} > \bar{z}$, then fathom the current node, and go to Step 2a. Otherwise, let $(\hat{\mathbf{x}}, \hat{\boldsymbol{\chi}}, \hat{\boldsymbol{\phi}}, \hat{\boldsymbol{\psi}}, \hat{\boldsymbol{\theta}})$ be an optimal solution.

**c.** Check the integrality of $\hat{\boldsymbol{\phi}}, \hat{\boldsymbol{\psi}}, \hat{\boldsymbol{\chi}}$, and $\hat{\mathbf{x}}$ in a certain branching priority. If the current solution is fractional, then create two nodes $\mathcal{N}_1$ and $\mathcal{N}_2$ by branching on the most fractional variable, update $\mathcal{T} := \mathcal{T} \cup \{\mathcal{N}_1, \mathcal{N}_2\}$, and go to Step 2a. Otherwise, the current solution is integral in $\hat{\boldsymbol{\phi}}, \hat{\boldsymbol{\psi}}, \hat{\boldsymbol{\chi}}$, and $\hat{\mathbf{x}}$.

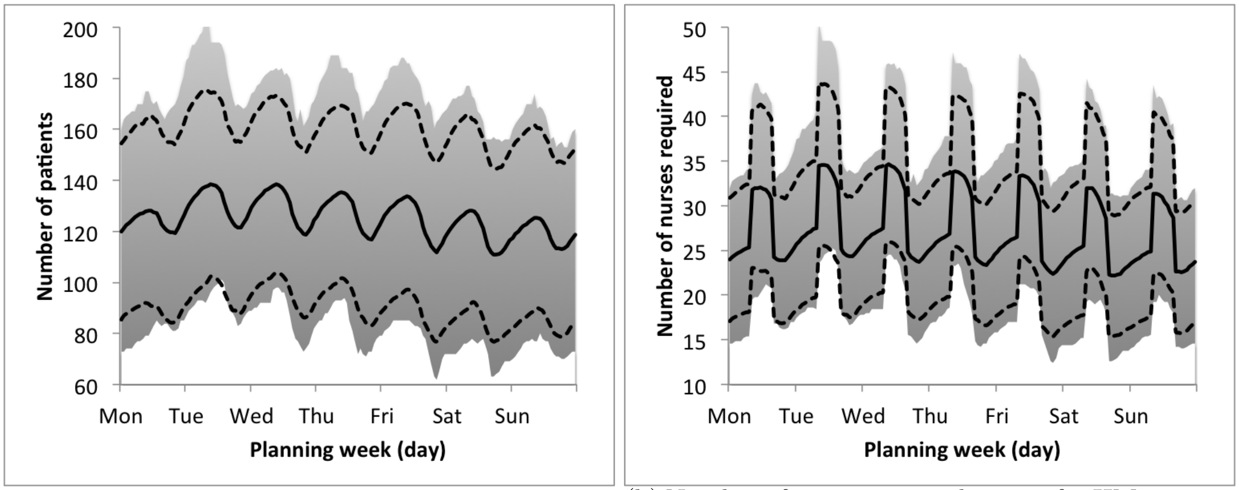
**d.** If $\bar{z} > \sum_{i \in I} c_i \hat{x}_i + \sum_{s \in S} p_s Q(\hat{\boldsymbol{\chi}}, \omega_s)$, then update upper bound $\bar{z} := \sum_{i \in I} c_i \hat{x}_i + \sum_{s \in S} p_s Q(\hat{\boldsymbol{\chi}}, \omega_s)$. Otherwise, fathom the current node, and go to Step 2a.

**e.** If $\sum_{s \in S} p_s \hat{\theta}_s < \sum_{s \in S} p_s Q(\hat{\boldsymbol{\chi}}, \omega_s)$, then set $k := k+1$, $M^k := M^0$ and $S_m^k := S_m^0$ for $m = 1, \ldots, M^0$, generate outer linearization cuts (13), aggregate the cuts as described in Algorithm 1, and go to Step 2b. Otherwise, fathom the current node, and go to Step 2a.

We assume that the proposed method solves (SP$^{\mathcal{N}}$) with a cut generation approach specified by the subsets $S_m^0$ of scenario indices for $m = 1, \ldots, M^0$, a predefined number $M^0$ of outer linearization inequalities.

The outer linearization cuts added at the root node are aggregated when an optimal solution to the convex relaxation node subproblem is found (Step 1c). This approach decreases the time spent in solving the convex relaxation programming problem at the root node, while starting with a smaller number of node subproblems in the B&C procedure. The MCA approach presented in Section 4.2 has helped in keeping the memory needs for the solver relatively low (Step 2e). In the B&C procedure, outer linearization inequalities are generated only at incumbent solutions. This approach avoids a significant increase in the memory required to save a larger size of node

**Figure 2** Fluctuations in patient census and required staffing levels for HM




(a) Number of patients in HM on a day of the week

(b) Number of nurses required to care for HM patients based on the desired nurse-to-patient ratios

subproblems (see Appendix F) and also avoids a number of computations for solving the second-stage problems. The algorithm terminates if the B&C tree becomes empty.

REMARK 1. Algorithm 2 stops after a finite number of steps for problems with a bounded set feasible set $X$. To see this, first note that the dual multipliers $\boldsymbol{\mu}, \boldsymbol{\pi}$ correspond to one of the bases of $Q_{\mathrm{LR}}(\boldsymbol{\chi}, \omega)$ for a given $\boldsymbol{\chi}, \omega$. The finite convergence of Step 1 follows from the fact that there are a finite number of different combinations of the dual multipliers $\boldsymbol{\mu}, \boldsymbol{\pi}$. Moreover, Step 2 terminates finitely by branching on variables, since the set $X$ is bounded.

## 5. Empirical Study: Nurse Staffing at NMH Department of Hospital Medicine

We now present results from an empirical study for nurse staffing and scheduling in the Department of Hospital Medicine (HM) at Northwestern Memorial Hospital (NMH). This empirical study aims to (1) evaluate the value of the stochastic programming approach as compared with the decisions recommended by a deterministic version of the model based on a point forecast and (2) examine the computational performance of the integer L-shaped method with enhancements presented in Section 4.

### 5.1. Model instances and input data

**5.1.1. Patient census data** Hourly patient census data was collected for HM service from the Northwestern Medicine Enterprise Data Warehouse from January 2009 to June 2012. Figure 2a shows patient volume during a week for the HM service over the entire study period. The shaded region represents the minimum and maximum patient volume observed during the study period.

The solid line shows the mean patient volume. Two standard deviations of the patient volume above and below the mean are shown by dashed lines. The nurse-to-patient ratios were provided by NMH operations managers. A 1:4 nurse-to-patient ratio is applied to the patient volume from 8am to 4pm, and a 1:5 ratio is applied for the rest of the day. The ideal nurse count required to provide patient care is shown in Figure 2b. The detailed patient volume characteristics are given in Appendix B.

**5.1.2. Problem instances** Using the patient census data, we created 20 problem instances in our empirical study. Each problem instance considers an 18-week planning horizon that rolls by 4 weeks. Rolling by 4 weeks allowed us to generate a sufficient number of problem instances for numerical testing. Each problem instance is generated as follows.

Based on the employment types, nurses at the NMH Department of Hospital Medicine work either 8 hours or 12 hours per shift (see Table 2). Each 12-hour shift starts from 8AM, 10AM, 12PM, or 8PM; and each 8-hour shift starts from 12AM, 8AM, 12PM, or 4PM. We assume a 12-hour break between two successive shifts worked by a nurse. Scheduling patterns were generated by using a recursive procedure (see Appendix D). For a given shift type, the procedure checks over hours of a week whether the hour is valid for a shift start time. For a valid shift start time, the procedure adds it to the set of start times and calls itself with the set. In subsequent calls, the procedure generates a scheduling pattern if the required number of shifts (i.e., start times) is generated. As a result, we generated 3,913 ($=|I|$) viable weekly scheduling patterns. Similarly, eight adjustment patterns were generated on the basis of one 12-hour shift and one 8-hour shift for each day with start times given above. In the model, adjustment decisions are made once a week for all days of the week. Thus the model has 56 ($=|J|$) ($8 \times 7$) adjustment patterns. The adjustment decision for day $i$ is not chained by the decisions for day $(i-1)$ over the adjustment horizon. Hence, the adjustments are separable by day as well as week.

The full model has 3,913 first-stage integer variables. The second stage has 112 integer variables and 336 continuous variables for each week of the 12-week staffing horizon and for each scenario. Note that the second-stage problem is separable by each of the 12 weeks since the schedules are based on weekly patterns. Consequently, each problem instance has 1,347,913 general integer variables and 4,032,000,000 continuous variables. 1,344,000 integer variables in the second stage are treated as continuous variables after adding MIR inequalities due to Theorem 1. Hence, the resulting integer program has only 3,913 integer variables. However, the CPLEX MIP optimizer resulted in an out-of-memory error (128 GB) on our computation server when we attempted to solve the extensive form formulation.

**5.1.3. Scenario generation** We used empirical forecast errors to generate demand scenarios for our model. We used the available 3.5 years of data as follows for generating the forecast errors. A 613-day time window was used for generating the forecasts and computing the errors in these forecasts. The window was moved by an hour between January 1, 2009, and June 17, 2010, for 532 days. Note that the test problems are generated for Sept. 2010 – Feb. 2012; hence, the empirical forecast errors are generated by using data for a period that does not overlap with the decision period. On our data, the autoregressive integrated moving average (ARIMA) method outperformed the other methods among a variety of known forecasting techniques for long-term forecasts. The MAPE for the forecast are shown in Figure 3 in Appendix B. Details of the predictability of different time series forecasting methods to the HM patient volume are discussed by Kim et al. (2014). An ARIMA model was estimated for each time window, and forecasting errors were evaluated for the next 18-week forecasts generated from the time window. These steps provided a pool of 12,768 ($= 532 \times 24$) forecast error vectors that reflect the past behavior of the ARIMA forecasting method. From this pool, we randomly selected 1,000 error vectors using a uniform distribution over the error vectors. A selected error vector was added to a mean point forecast to generate a demand scenario. The choice of using 1,000 scenarios for the results reported in our empirical study is arbitrary. Results in Appendix C show that using a smaller number of scenarios results in larger staffing costs and lower value of stochastic solution.

**5.1.4. Additional parameter settings and model modifications** We set the cost coefficients $c_i, q_j^+, q_j^-, r^+$, and $r^-$ as relative weights to staffing cost per hour. Staffing cost $c_i$ is set to 1 per hour, while the underage cost $q_j^+$ is 1.5 per hour. The reason is that the nurses called to do an overtime shift get paid 50% more than the base salary. We assume that the salvage value (overage cost) of a nurse is zero ($q_j^- = 0$). The labor surplus benefit is usually zero unless better service is provided (Easton and Rossin 1996) or alternative work is found. In the base model, the penalty costs $r^+$ and $r^-$ are set sufficiently large ($r^+ = r^- = 50$) to have a model that tracks the nurse-to-patient ratio as closely as possible at the time of service. The need to track patient census was suggested to us by NMH operations managers for patient safety reasons (see also Woodall et al. (2013)). However, we also report results with alternative penalty cost parameters in Section 5.2.3. The model also assumes that the number of full-time nurses should be more than 80% of the total number of nurses and that the number of part-time nurses should be more than 10% of the total number of nurses. This full-time/part-time/casual staff mix is the current policy at NMH. Therefore, the following additional constraints were added to the model:

$$\sum_{i \in I_{\mathrm{FT}}} x_i \geq 0.8 \sum_{i \in I} x_i \quad \text{and} \quad \sum_{i \in I_{\mathrm{PT}}} x_i \geq 0.1 \sum_{i \in I} x_i,$$

where $I_{\mathrm{FT}}$ and $I_{\mathrm{PT}}$ are the subsets of scheduling patterns for full-time shifts and part-time shifts, respectively. Note that Theorem 1 remains applicable even after adding these constraints.

**Table 3** Comparison of deterministic solutions and stochastic solutions, and the value of the stochastic solutions for the 20 problem instances

| | Deterministic Solutions | | | | Stochastic Solutions | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Instance | Staffing Hours | Overtime Hours | Paid Time-Off Hours | Total Cost | Staffing Hours | Overtime Hours | Paid Time-Off Hours | Total Cost | VSS |
| 20FEB2012 | 5000 | 230 | 343 | 5344 | 4676 | 365 | 165 | 5223 | 121 |
| 23JAN2012 | 5556 | 252 | 425 | 5934 | 5096 | 455 | 188 | 5778 | 156 |
| 26DEC2011 | 5064 | 216 | 315 | 5388 | 4760 | 344 | 149 | 5276 | 112 |
| 28NOV2011 | 5272 | 228 | 337 | 5614 | 4928 | 376 | 152 | 5492 | 122 |
| 31OCT2011 | 5392 | 268 | 377 | 5794 | 5012 | 438 | 183 | 5669 | 125 |
| 03OCT2011 | 5204 | 241 | 394 | 5566 | 4760 | 446 | 174 | 5430 | 136 |
| 05SEP2011 | 5108 | 258 | 381 | 5495 | 4676 | 462 | 171 | 5368 | 127 |
| 08AUG2011 | 5704 | 288 | 407 | 6136 | 5236 | 511 | 184 | 6003 | 133 |
| 11JUL2011 | 5564 | 255 | 391 | 5947 | 5180 | 422 | 189 | 5813 | 135 |
| 13JUN2011 | 5544 | 240 | 384 | 5903 | 5132 | 420 | 169 | 5762 | 141 |
| 16MAY2011 | 5712 | 253 | 407 | 6091 | 5236 | 472 | 170 | 5944 | 147 |
| 18APR2011 | 5588 | 255 | 379 | 5970 | 5180 | 443 | 176 | 5844 | 126 |
| 21MAR2011 | 5296 | 232 | 359 | 5644 | 4928 | 388 | 160 | 5511 | 134 |
| 21FEB2011 | 5588 | 263 | 383 | 5982 | 5180 | 452 | 181 | 5858 | 124 |
| 24JAN2011 | 5388 | 230 | 390 | 5733 | 5012 | 384 | 182 | 5588 | 145 |
| 27DEC2010 | 5252 | 229 | 351 | 5595 | 4928 | 360 | 169 | 5468 | 127 |
| 29NOV2010 | 5120 | 226 | 337 | 5460 | 4760 | 386 | 148 | 5339 | 121 |
| 01NOV2010 | 5224 | 225 | 332 | 5562 | 4928 | 344 | 165 | 5444 | 118 |
| 04OCT2010 | 5316 | 238 | 371 | 5673 | 4928 | 403 | 163 | 5533 | 140 |
| 06SEP2010 | 5296 | 226 | 333 | 5635 | 4928 | 390 | 140 | 5512 | 123 |
| Mean | 5359 | 243 | 370 | 5723 | 4973 | 413 | 169 | 5593 | 130 |
| Stdev | 214.3 | 18.1 | 29.9 | 233.8 | 182.8 | 46.1 | 13.9 | 227.7 | 11.1 |

**5.1.5. Computational environment** We implemented the modified integer L-shaped method (see Algorithm 2) in C++ and used the CPLEX 12.5.0 callable library (CPLEX 2009) to solve the generated linear and mixed integer programming subproblems. CPLEX callback functions were used for the branch-and-cut procedure in Algorithm 2. The CPLEX parameters used with nondefault values in our empirical study are given in Appendix A. Note that the relative optimality gap was set to zero. The code was run on a 32-core Intel Xeon 2.2 GHz machine with 128 GB RAM, although we report only the total CPU time in this paper. Patient volume forecasts were generated by using the procedures in the R software package (Eddelbuettel and François 2010, Kim et al. 2014).

## 5.2. Value of the stochastic solution (VSS)

**5.2.1. VSS based on scenarios** We now evaluate the value of the stochastic solutions (VSS; see Birge (1982)) resulting from our iStaff model when compared with the solutions from the deterministic model that considers the iStaff model with a single scenario based on point forecast of patient volume. Table 3 provides solutions resulting from the deterministic model, the stochastic programming iStaff model, and the VSS. The problem instance is named by the first date of the planning horizon. The column “Staffing Hours” gives the number of nursing hours per week. The columns “Overtime Hours” and “Paid Time-Off Hours” give the expected number of hours for which HM will need to call in an overtime nurse or have a nurse with no salvage value. The total cost is the sum of the staffing cost and the expected adjustment cost per week. The VSS calculates the difference between two total costs. In calculating the total cost we excluded the penalty cost, which corresponds to $\mathbb{E}[r^+ \sum_{t\in T} u_t(\omega) + r^- \sum_{t\in T} v_t(\omega)]$, because the penalty cost is not out-of-pocket of hospital operating cost.

As can be seen in Table 3, the use of the stochastic iStaff model saves the cost of staffing 130 nursing hours a week on average, which is equivalent to the cost of hiring 3.6 full-time nurses on average. The VSS was also evaluated by using all 12,768 error vectors (see Appendix C). This value is not significantly different from that reported in Table 3. On average the stochastic model staffs 386 fewer nursing hours while calling in only 170 nursing overtime hours in comparison with the deterministic solutions. The deterministic solutions have more paid time-off hours than do the stochastic solutions. In Appendix E, we report computation time and other computational results for solving the deterministic models. The average CPU time for the deterministic model was 13 seconds. The optimal solutions were found at the root node for 16 of 20 problem instances.

**5.2.2. VSS based on actual patient census** The VSS reported in Table 3 assumes that (1) the scenarios generated from the ARIMA models follow the empirical demand distribution for the planning horizon and (2) improved demand information by a week-ahead forecast represents actual patient volume at the adjustment decision epoch. In practice, however, improved demand information could still have forecast error. In our empirical setting, a week-ahead forecast has 9% mean absolute percentage error on average, whereas a six-week-ahead forecast has an average 12% error. Hence, one may be interested in the VSS based on actual realizations of patient census for the planning horizon. Table 4 gives results from the deterministic solutions and the stochastic solutions using actual patient census. On our data during the study period, the use of the stochastic iStaff model on average saves the cost of staffing 108 nursing hours a week (i.e., three full-time nurses). Unlike the results in Table 3, the stochastic solutions incurred more cost than the deterministic solutions did for some instances (26DEC2011, 29NOV2010, and 04OCT2010 in Table 4). In addition, the VSS was low for instances 28NOV2011, 01NOV2010, and 06SEP2010. The reason is that the patient volume forecast used for the recourse considerations was significantly lower than that actually realized. Consequently, the plan required significantly more overtime hours for these cases.

**Table 4** The deterministic solutions and the stochastic solutions evaluated using the actual realizations of patient census

| | Deterministic Solutions | | | | Stochastic Solutions | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Instance | Staffing Hours | Overtime Hours | Paid Time-Off Hours | Total Cost | Staffing Hours | Overtime Hours | Paid Time-Off Hours | Total Cost | VSS |
| 20FEB2012 | 5000 | 19 | 285 | 5028 | 4676 | 216 | 162 | 5001 | 28 |
| 23JAN2012 | 5556 | 45 | 657 | 5623 | 5096 | 160 | 312 | 5336 | 288 |
| 26DEC2011 | 5064 | 248 | 216 | 5437 | 4760 | 466 | 132 | 5459 | -23 |
| 28NOV2011 | 5272 | 229 | 355 | 5615 | 4928 | 454 | 235 | 5609 | 6 |
| 31OCT2011 | 5392 | 95 | 597 | 5535 | 5012 | 287 | 408 | 5443 | 92 |
| 03OCT2011 | 5204 | 73 | 695 | 5314 | 4760 | 267 | 446 | 5161 | 153 |
| 05SEP2011 | 5108 | 26 | 594 | 5148 | 4676 | 218 | 355 | 5004 | 144 |
| 08AUG2011 | 5704 | 1 | 944 | 5706 | 5236 | 70 | 546 | 5342 | 365 |
| 11JUL2011 | 5564 | 65 | 484 | 5662 | 5180 | 225 | 262 | 5518 | 144 |
| 13JUN2011 | 5544 | 81 | 388 | 5665 | 5132 | 293 | 190 | 5572 | 93 |
| 16MAY2011 | 5712 | 31 | 595 | 5758 | 5236 | 206 | 297 | 5545 | 213 |
| 18APR2011 | 5588 | 0 | 531 | 5588 | 5180 | 121 | 247 | 5362 | 226 |
| 21MAR2011 | 5296 | 43 | 424 | 5361 | 4928 | 248 | 262 | 5300 | 61 |
| 21FEB2011 | 5588 | 49 | 601 | 5662 | 5180 | 205 | 349 | 5488 | 174 |
| 24JAN2011 | 5388 | 104 | 445 | 5545 | 5012 | 300 | 266 | 5463 | 82 |
| 27DEC2010 | 5252 | 142 | 370 | 5466 | 4928 | 302 | 207 | 5381 | 85 |
| 29NOV2010 | 5120 | 204 | 247 | 5427 | 4760 | 451 | 135 | 5436 | -10 |
| 01NOV2010 | 5224 | 220 | 342 | 5554 | 4928 | 400 | 228 | 5528 | 26 |
| 04OCT2010 | 5316 | 204 | 318 | 5623 | 4928 | 465 | 191 | 5626 | -3 |
| 06SEP2010 | 5296 | 216 | 338 | 5620 | 4928 | 456 | 212 | 5613 | 7 |
| Mean | 5359 | 105 | 471 | 5517 | 4973 | 291 | 272 | 5409 | 108 |
| Stdev | 214 | 85 | 180 | 187 | 182 | 121 | 106 | 182 | 106 |

**5.2.3. Sensitivity analysis in model parameters** In this section, we give results showing the absolute VSS and the relative VSS under different parameter settings. Table 5 presents the results by varying the recourse function cost $r^+$ and $r^-$ and patient volume characteristics. Absolute and relative savings (in parentheses) are given in this table. We experimented with different patient volume by taking $\max\{0, d_t(\omega) - \Delta\}$, where $\Delta$ are 0, 40, and 70. Hence, the mean patient volume is reduced by a given $\Delta$. Note that the standard deviation does not change. We calculated the mean absolute percentage error of each patient volume forecast reduced by $\Delta$. The combinations of $r^+$ and $r^-$ are taken with the following justifications: (1) $r^+ = 0.75, r^- = 1.25$: some salvage cost and low cost nurse addition; (2) $r^+ = 0.5, r^- = 2$: low salvage cost and agency nurse short notice; (3) $r^+ = r^- = 50$: closely track demand; (4) $r^+ = 5, r^- = 10$: loosely track demand; (5) $r^+ = 0.5, r^- = 50$: low salvage cost and significant safety costs due to understaffing; and (6) $r^+ = 0, r^- = 50$: no salvage cost and significant safety costs due to understaffing.

For a given recourse function cost, the absolute VSS shows a decreasing tendency as mean absolute percent error increases. The largest absolute VSS was obtained when some salvage cost and low nurse replacement cost were used in the model with high MAPE, whereas the smallest absolute VSS was obtained when the adjustment decisions were made to strictly meet the patient volume demand regardless of overstaffing level.

**Table 5** Sensitivity analysis of the value of the stochastic solutions on different cost function coefficients ($r^+$ and $r^-$) and mean absolute percentage errors of patient volume forecast

| | | Mean Absolute Percentage Error | | |
|---|---|---|---|---|
| $r^+$ | $r^-$ | 12% | 18% | 27% |
| 0.75 | 1.25 | 547.5 (10.7%) | 524.6 (20.6%) | 480.2 (34.0%) |
| 0.5 | 2 | 180.73 (3.4%) | 167.89 (6.0%) | 162.45 (9.6%) |
| 50 | 50 | 130.01 (2.3%) | 122.84 (3.9%) | 115.94 (5.6%) |
| 5 | 10 | 83.45 (1.5%) | 76.68 (2.5%) | 76.95 (3.8%) |
| 0.5 | 50 | 52.11 (0.8%) | 49.42 (1.3%) | 51.04 (1.8%) |
| 0 | 50 | 52.75 (0.8%) | 51.01 (1.3%) | 48.32 (1.7%) |

Note that the relative VSS decreases as the recourse function costs $r^+$ and $r^-$ increase. The reason is that the denominator of the relative VSS is the optimal objective value of the model, which increases in the recourse function cost. Moreover, since $d_t(\omega)$ are fractional in the model, positive residuals $u_t(\omega)$ and $v_t(\omega)$ must exist regardless of any model parameter setting. Hence, one can easily see that the relative VSS goes to zero as the recourse function cost increases.

We also evaluated the objective values of the solutions based on 100 scenarios by using all 12,768 error vectors. The average VSS based on a choice of 100 scenarios was significantly lower than that based on the 1,000 scenarios (110 vs. 130 with p-value $< 0.01$). Moreover, the standard deviation of the objective function values based on the 100 scenarios was larger than that based on the 1,000 scenarios on average (599 vs. 559). Detailed results are given in Appendix C.

## 5.3. Expected value of perfect information

We evaluate the maximum staffing cost saving if perfect patient volume were available in the iStaff model. The expected value of perfect information (EVPI) has been used in the context of decision analysis to measure the amount that a decision maker is willing to pay in return for perfect information about uncertain factors (Pratt et al. 1995, Birge and Louveaux 1997). The EVPI is calculated as follows. We call the staffing cost resulting from (TSSIP) the wait-and-see staffing cost. We compare the wait-and-see staffing cost with the so-called here-and-now staffing cost, which is obtained by solving $\min\left\{\mathbb{E}_\omega\left[\sum_{i\in I} c_i x_i + Q(\boldsymbol{\chi},\omega)\right] \mid (1a)-(1b)\right\}$. Then, the EVPI is the difference between the wait-and-see staffing cost and the here-and-now staffing cost.

Table 6 provides the wait-and-see staffing cost, the here-and-now staffing cost, and the EVPI for each problem instance. The EVPI for the iStaff model is the cost of staffing 300 nursing hours a week on average. The HM service would save 5.4% of the staffing cost more than the wait-and-see staffing cost resulting from (TSSIP) if perfect and accurate patient volume information were available to the iStaff model.

**Table 6** Expected value of perfect patient volume information (EVPI) for the iStaff model

| Instance | Wait-and-See Staffing Cost | Here-and-Now Staffing Cost | EVPI | |
|---|---|---|---|---|
| 20FEB2012 | 5223 | 4946 | 277 | 5.3% |
| 23JAN2012 | 5778 | 5443 | 335 | 5.8% |
| 26DEC2011 | 5276 | 5025 | 251 | 4.8% |
| 28NOV2011 | 5492 | 5223 | 269 | 4.9% |
| 31OCT2011 | 5669 | 5346 | 324 | 5.7% |
| 03OCT2011 | 5430 | 5111 | 318 | 5.9% |
| 05SEP2011 | 5368 | 5046 | 322 | 6.0% |
| 08AUG2011 | 6003 | 5644 | 359 | 6.0% |
| 11JUL2011 | 5813 | 5491 | 322 | 5.5% |
| 13JUN2011 | 5762 | 5461 | 301 | 5.2% |
| 16MAY2011 | 5944 | 5619 | 325 | 5.5% |
| 18APR2011 | 5844 | 5526 | 318 | 5.4% |
| 21MAR2011 | 5511 | 5229 | 281 | 5.1% |
| 21FEB2011 | 5858 | 5531 | 328 | 5.6% |
| 24JAN2011 | 5588 | 5288 | 300 | 5.4% |
| 27DEC2010 | 5468 | 5190 | 278 | 5.1% |
| 29NOV2010 | 5339 | 5069 | 270 | 5.0% |
| 01NOV2010 | 5444 | 5177 | 267 | 4.9% |
| 04OCT2010 | 5533 | 5243 | 290 | 5.2% |
| 06SEP2010 | 5512 | 5250 | 263 | 4.8% |
| Mean | 5593 | 5293 | 300 | 5.4% |
| Stdev | 227 | 206 | 29 | 0.4% |

## 5.4. Computational experience with the modified L-shaped method

We now discuss the computational performances of our algorithmic approach. In Section 5.4.1 we compare our multicut aggregation (MCA) approach with the single-cut approach and the hybrid-cut approach given in Section 4.2. The outer linearization cuts were generated only at the incumbent solutions. More aggressive cut generation increased the computation time. For example, when the cuts are generated at every feasible integer solution, the computation time is doubled (see Appendix F). However, the maximum memory (128 GB) available on our computation server was not enough when the cuts are generated at all fractional solutions. In Section 5.4.2 we discuss the computational results from the proposed thin direction branching strategy.

**5.4.1. Computational performance of multicut aggregation** We now give results comparing the computational performance of the single-cut approach, the hybrid-cut approach, and the proposed MCA approach within the framework of the L-shaped method. Table 7 gives the average computational performances of different approaches on the 20 problem instances. The single-cut approach adds only one cut (i.e., $M^k = 1$) generated for each $\boldsymbol{\chi}^k$ at iteration $k$. The hybrid-cut approach adds $M^k$ number of cuts generated for each $\boldsymbol{\chi}^k$ at iteration $k$. The MCA approach also adds $M^k$ number of cuts generated for each $\boldsymbol{\chi}^k$ but aggregates those having dual multipliers equal to zero. We consider the hybrid cut and the MCA approaches to add 10, 100, 500, and 1,000 cuts for

**Table 7** Average computational performances of different cut aggregation approaches for the 20 problem instances

| Approach | No. of Cuts | No. of Nodes | Root Node CPU Time (sec.) | B&C CPU Time (sec.) | Total CPU Time (sec.) | Absolute Optimality Gap at the First Feasible Solution |
|---|---|---|---|---|---|---|
| Single cut | 533 | 944 | 56492 | 2917 | 59409 | 15 |
| Hybrid cut (10 cuts) | 3284 | 1133 | 35020 | 5860 | 40879 | 48 |
| Hybrid cut (100 cuts) | 11309 | 1919 | 12290 | 52091 | 64381 | 46 |
| Hybrid cut (500 cuts) | NA† | NA† | NA† | NA† | NA† | NA† |
| Hybrid cut (1000 cuts) | NA† | NA† | NA† | NA† | NA† | NA† |
| MCA (10 cuts) | 94 | 1831 | 34704 | 38382 | 73086 | 48 |
| MCA (100 cuts) | 615 | 2572 | 11828 | 53908 | 65736 | 44 |
| MCA (500 cuts) | 2944 | 1653 | 6468 | 20524 | 26991 | 31 |
| MCA (1000 cuts) | 4354 | 918 | 5261 | 4749 | 10010 | 26 |

† Hybrid cut (500 cuts) and hybrid cut (1,000 cuts) reached the maximum memory (128 GB) available in our computation server.

each $\boldsymbol{\chi}^k$ (i.e., $M^k = 10, 100, 500$, and 1,000). The subsets $S_m^k$ of scenario indices are constructed in a round-robin fashion for $m = 1, \ldots, M^k$. The absolute optimality gap at the first feasible solution is given by the absolute difference between the optimal objective value and the objective value at a feasible solution that was first found.

The MCA approach used the least total CPU time (10,010 sec.) when allowing 1,000 cuts. The known hybrid-cut approach had the least total CPU time (40,879 sec.) for 10 cuts among different parameter settings. The total CPU time was reduced by a factor of 4.1 on average for our problem instances. In addition to the reduced root node CPU time, the branch-and-cut (B&C) CPU time resulting from the MCA with 1,000 cuts approach was also reduced by a factor of 1.2 when compared with that of the hybrid cut with 10 cuts. The hybrid cut with 500 cuts and that with 1,000 cuts could not find a solution as they reached the maximum memory (128 GB) available in our computation server. The single-cut approach required more than ten times the CPU time at the root node, but it did produce better root node solutions that required less time subsequently to reach optimality. Overall, MCA with 1,000 cuts required about one-sixth the time to solve the test problems.

Appendix G reports the results from a different multicut approach that purges the cuts with zero dual multipliers. Although this multicut purge approach resulted in nearly the same computational performances for 16 of the 20 instances, the proposed MCA approach outperformed in the other four instances.

**5.4.2. Computational performance of the thin direction branching strategy** We now examine the computational performance of the thin direction branching strategy devised by introducing the auxiliary variables $\boldsymbol{\phi}, \boldsymbol{\psi}$, and $\boldsymbol{\chi}$ with priority as discussed in Section 4.1. Recall that with this branching strategy, we first seek integrality in the number of full-time and part-time nurses and the number of nurses in each shift type. Table 8 gives results comparing the computational performance of our priority branching with a tender variable branching strategy that gives priority to branching on tender variables $\boldsymbol{\chi}$, followed by branching on the original variables $\mathbf{x}$.

The proposed thin direction branching strategy on average used 918 nodes in the B&C tree. This is smaller by a factor of 1.6 when compared with the tender variable branching strategy (1,499 nodes). The thin direction branching strategy branched on original variable $\mathbf{x}$ only a few times (i.e., 32 vs. 1,484). Moreover, in the B&C procedure, the thin direction branching strategy found 19 feasible solutions on average, whereas the tender variable branching strategy found 56 feasible solutions on average. This increased B&C CPU time from the tender variable branching results, because of the additional efforts required in objective function evaluation. Hence, the thin direction branching strategy took less time per node subproblem for all the problem instances (i.e., 5.2 vs. 9.4 CPU-seconds per node on average). As a result, the B&C CPU time resulting from the thin direction branching strategy is lower than that from the tender variable branching by a factor of 3. This result implies that by branching on thin directions, $\boldsymbol{\phi}$ and $\boldsymbol{\psi}$, the integrality of the solution is more efficiently achieved in the B&C search tree.

In Appendix H we report the results from the standard branching strategy that considers branching on original variables $\mathbf{x}$ only. The reduction in the B&C CPU time by the tender variable branching is not significant (14,068 vs. 14,175 sec. on average). The reason is that the tender variable branching strategy on average performed branching 15 times on tender variables compared with 1,484 times on original variables. Note that in contrast the prioritized branching strategy performs most of its branching on the tender variables $\boldsymbol{\chi}$, after generating some branches using $\boldsymbol{\phi}$ and $\boldsymbol{\psi}$. This result suggests the benefits of branching on the implied thin directions by the proposed prioritized branching strategy.

## 6. Concluding Remarks

The integrated staffing and scheduling model studied in this paper is applicable for a single unit. This model assumes that nurses are identical in their skill set. This assumption is justified for a single unit (e.g., hospital medicine) problem. In more general hospital-wide settings, nurses across different units have different certifications and skill sets. In certain situations, or with additional skill sets, these nurses are interchangeable across units. An example is the nurse float pool, which is often maintained in a large hospital. Nurses in the float pool may work in many different units. Developing and studying hospital-wide staffing and scheduling models are a topic of future research. Another important issue in developing schedules is individual preferences (see, e.g., Bard and Purnomo 2005b,c, Maenhout and Vanhoucke 2013b). There are two possible approaches to address this issue. The first approach is to develop a detailed preference-based optimization model

**Table 8** Computational performance of the proposed priorty branching strategy and tender variable branching

| | Thin Direction Branching | | | | | | | | Tender Variable Branching | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | No. of | No. of | | | | B&C | Total | | No. of | No. of | | B&C | Total |
| | No. of | Feasible | Branching on | | | | CPU Time | CPU Time | No. of | Feasible | Branching on | | CPU Time | CPU Time |
| Instance | Nodes | Solutions | $\phi$ | $\psi$ | $\chi$ | $\mathbf{x}$ | (sec.) | (sec.) | Nodes | Solutions | $\chi$ | $\mathbf{x}$ | (sec.) | (sec.) |
| 20FEB2012 | 294 | 10 | 36 | 38 | 152 | 11 | 2020 | 8405 | 1256 | 66 | 6 | 1254 | 10887 | 17204 |
| 23JAN2012 | 2118 | 44 | 99 | 59 | 1784 | 112 | 10565 | 15734 | 1111 | 53 | 0 | 1114 | 11791 | 16908 |
| 26DEC2011 | 2131 | 26 | 107 | 141 | 1692 | 59 | 7107 | 13047 | 1508 | 70 | 49 | 1453 | 18078 | 23959 |
| 28NOV2011 | 926 | 17 | 72 | 56 | 699 | 12 | 4032 | 9356 | 1808 | 74 | 11 | 1801 | 14609 | 19837 |
| 31OCT2011 | 621 | 28 | 52 | 45 | 477 | 18 | 5728 | 10512 | 1915 | 75 | 0 | 1890 | 25221 | 30041 |
| 03OCT2011 | 1159 | 22 | 115 | 78 | 852 | 12 | 5722 | 10726 | 1568 | 46 | 5 | 1566 | 15008 | 20012 |
| 05SEP2011 | 536 | 14 | 57 | 48 | 364 | 15 | 2953 | 7740 | 1418 | 39 | 12 | 1405 | 7460 | 12179 |
| 08AUG2011 | 1213 | 23 | 79 | 71 | 913 | 81 | 7746 | 12322 | 1403 | 37 | 13 | 1390 | 8339 | 12839 |
| 11JUL2011 | 655 | 19 | 49 | 37 | 475 | 23 | 5930 | 11095 | 2409 | 65 | 0 | 2408 | 18311 | 23506 |
| 13JUN2011 | 626 | 23 | 58 | 63 | 448 | 14 | 4556 | 10100 | 295 | 30 | 0 | 298 | 4372 | 10063 |
| 16MAY2011 | 1206 | 25 | 122 | 114 | 866 | 18 | 6127 | 11270 | 2045 | 48 | 164 | 1870 | 21560 | 26690 |
| 18APR2011 | 830 | 14 | 69 | 64 | 592 | 26 | 3315 | 8045 | 1592 | 54 | 0 | 1595 | 9837 | 14856 |
| 21MAR2011 | 800 | 14 | 96 | 67 | 502 | 60 | 3501 | 8286 | 1544 | 60 | 0 | 1546 | 12346 | 17519 |
| 21FEB2011 | 1146 | 29 | 90 | 47 | 859 | 80 | 5870 | 10745 | 1585 | 64 | 11 | 1574 | 15649 | 20990 |
| 24JAN2011 | 777 | 18 | 68 | 59 | 584 | 5 | 3400 | 8497 | 1627 | 72 | 32 | 1597 | 27468 | 32924 |
| 27DEC2010 | 658 | 13 | 45 | 37 | 500 | 17 | 3560 | 9278 | 1824 | 67 | 0 | 1828 | 16804 | 22527 |
| 29NOV2010 | 292 | 7 | 68 | 41 | 177 | 7 | 1795 | 7218 | 946 | 40 | 0 | 949 | 8271 | 13864 |
| 01NOV2010 | 722 | 7 | 47 | 32 | 557 | 18 | 3019 | 8628 | 2348 | 76 | 0 | 2350 | 18569 | 24174 |
| 04OCT2010 | 1536 | 15 | 79 | 95 | 1237 | 46 | 6372 | 11599 | 722 | 41 | 0 | 730 | 8544 | 13754 |
| 06SEP2010 | 111 | 10 | 11 | 16 | 46 | 4 | 1653 | 7590 | 1059 | 51 | 0 | 1061 | 8244 | 14067 |
| Mean | 918 | 19 | 71 | 60 | 689 | 32 | 4749 | 10010 | 1499 | 56 | 15 | 1484 | 14068 | 19396 |
| Stdev | 540 | 9 | 28 | 29 | 457 | 31 | 2258 | 2144 | 515 | 14 | 37 | 504 | 6200 | 6203 |

and optimize for overall satisfaction. The alternative approach is to first create schedules without preferences and then offer these schedules to the nurses for their bidding. The issue of scheduling preference was discussed with the NMH administration in the context of changing schedules based on our results. The suggestion was that, based on nurse preferences, use of certain scheduling patterns could be further restricted by additional lower and upper bounds on the first-stage decision variables.

Despite these limitations, this paper takes several important steps in the direction of incorporating uncertainty in the workforce planning. For a single-unit problem, the computational results in this paper show that realistic integrated staffing and scheduling problems, modeled as two-stage stochastic mixed-integer programs, can be solved in a reasonable amount of time. A key to solving such problems is the ability to add inequalities that tighten the relaxation of the second-stage mixed-integer sets. For the problems studied in this paper we showed that the tighter relaxations obtained by adding parametric mixed-integer inequalities were sufficient to specify the convex hull of the second-stage mixed-integer set. This remarkably reduced the complexity of the model. Further algorithmic enhancements to the L-shaped method resulted in another tenfold improvement in solution time and significant reduction in memory usage. On the practical side, the empirical study based on real data showed that one can achieve significant cost savings by appropriately modeling the future uncertainty. This result was demonstrated by evaluating the value of stochastic solution, as well as testing the performance of the model against real demand data. We also observed that the value of stochastic solution increases with mean percentage forecast error.

## Appendix A: CPLEX Parameter Setting

In Table 9, we list the CPLEX parameters with nondefault values used in our computational study. The first four parameter (`CPX_PARAM_MIPSEARCH`, `CPX_PARAM_MIPCBREDLP`, `CPX_PARAM_PRELINEAR`, and `CPX_PARAM_REDUCE`) settings are necessary for the implementation of our algorithm. Parameter `CPX_PARAM_PARALLELMODE` is set to 1 in order to enable deterministic parallel search mode in the CPLEX MIP optimizer. With the parameter setting to turn on all the CPLEX internal cut procedures, we observed that CPLEX did not generate any cut throughout the branch-and-cut procedure.

## Appendix B: Characteristics of Patient Census Data and Problem Instances

Patient census during the study period has a mean of 125 and a standard deviation of 19. As can be seen in Figure 2a, the patient volume fluctuates during different times of the day and days of the week. The patient volume is typically higher during Tuesdays through Fridays. Figure 2a also shows a large variability in patient volume on a given day of the week. NMH manages its nurse staffing levels using nurse-to-patient ratios for HM.

**Table 9** CPLEX parameters used with nondefault values in the computational study

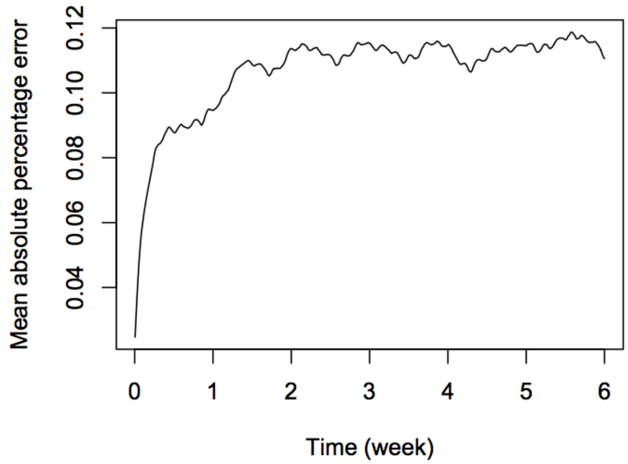
| Parameter Name | Value |
|---|---|
| CPX_PARAM_MIPSEARCH | CPX_MIPSEARCH_TRADITIONAL |
| CPX_PARAM_MIPCBREDLP | CPX_OFF |
| CPX_PARAM_PRELINEAR | 0 |
| CPX_PARAM_REDUCE | CPX_PREREDUCE_PRIMALONLY |
| CPX_PARAM_PARALLELMODE | 1 |
| CPX_PARAM_THREADS | 32 |
| CPX_PARAM_EPGAP | 0.0 |
| CPX_PARAM_LPMETHOD | CPX_ALG_DUAL |
| CPX_PARAM_MIPEMPHASIS | 3 |
| CPX_PARAM_BNDSTRENIND | 1 |
| CPX_PARAM_VARSEL | 2 |
| CPX_PARAM_DIVETYPE | 3 |
| CPX_PARAM_PRESLVND | 2 |
| CPX_PARAM_PROBE | 3 |

**Table 10** Characteristics of problem instances and patient census data

| | Training Horizon | | | | | Planning Horizon | | | |
|---|---|---|---|---|---|---|---|---|---|
| Instance | Date Range (613 days) | Mean | Stdev | Max | Min | Mean | Stdev | Max | Min |
| 20FEB2012 | 06/17/2010 - 02/19/2012 | 120 | 18.9 | 188 | 62 | 112 | 12.4 | 152 | 84 |
| 23JAN2012 | 05/20/2010 - 01/22/2012 | 121 | 19.1 | 188 | 62 | 114 | 12.8 | 152 | 84 |
| 26DEC2011 | 04/22/2010 - 12/25/2011 | 122 | 19.1 | 188 | 62 | 114 | 13.3 | 152 | 73 |
| 28NOV2011 | 03/25/2010 - 11/27/2011 | 123 | 18.8 | 188 | 62 | 113 | 14.8 | 152 | 70 |
| 31OCT2011 | 02/25/2010 - 10/30/2011 | 125 | 18.3 | 188 | 73 | 109 | 15.4 | 146 | 62 |
| 03OCT2011 | 01/28/2010 - 10/02/2011 | 126 | 18.4 | 188 | 73 | 109 | 15.5 | 146 | 62 |
| 05SEP2011 | 12/31/2009 - 09/04/2011 | 127 | 18.7 | 188 | 73 | 109 | 15.2 | 146 | 62 |
| 08AUG2011 | 12/03/2009 - 08/07/2011 | 127 | 18.9 | 188 | 73 | 113 | 14.6 | 146 | 62 |
| 11JUL2011 | 11/05/2009 - 07/10/2011 | 127 | 18.9 | 188 | 73 | 117 | 10.9 | 146 | 79 |
| 13JUN2011 | 10/08/2009 - 06/12/2011 | 128 | 18.9 | 188 | 73 | 118 | 10.9 | 147 | 77 |
| 16MAY2011 | 09/10/2009 - 05/15/2011 | 129 | 18.8 | 188 | 73 | 119 | 11.4 | 148 | 77 |
| 18APR2011 | 08/13/2009 - 04/17/2011 | 129 | 18.6 | 188 | 73 | 117 | 12.0 | 148 | 77 |
| 21MAR2011 | 07/16/2009 - 03/20/2011 | 130 | 18.8 | 189 | 73 | 117 | 12.9 | 152 | 77 |
| 21FEB2011 | 06/18/2009 - 02/20/2011 | 132 | 18.9 | 203 | 73 | 116 | 12.7 | 152 | 83 |
| 24JAN2011 | 05/21/2009 - 01/23/2011 | 133 | 18.3 | 203 | 73 | 115 | 13.6 | 152 | 74 |
| 27DEC2010 | 04/23/2009 - 12/26/2010 | 133 | 17.9 | 203 | 73 | 116 | 14.2 | 174 | 74 |
| 29NOV2010 | 03/26/2009 - 11/28/2010 | 135 | 17.2 | 203 | 78 | 113 | 15.2 | 174 | 73 |
| 01NOV2010 | 02/26/2009 - 10/31/2010 | 136 | 16.7 | 203 | 85 | 114 | 16.2 | 174 | 73 |
| 04OCT2010 | 01/29/2009 - 10/03/2010 | 136 | 16.5 | 203 | 85 | 121 | 21.6 | 177 | 73 |
| 06SEP2010 | 01/01/2009 - 09/05/2010 | 135 | 16.5 | 203 | 85 | 129 | 23.1 | 181 | 73 |

Table 10 shows the characteristics of patient census data used for each problem instance. The first column presents the problem instances, which are named by the first date of each planning horizon. For each problem instance, 613 days of patient census data were taken to train the ARIMA forecasting model. Mean, standard deviation, and maximum and minimum of patient census data are given for each of the training horizon and the planning horizon.

**Figure 3** Mean absolute percentage error resulting from the base scenarios in our empirical study.




In our empirical setting, a set of demand scenarios is generated by using an ARIMA model and empirical error vectors, which simulate hourly patient volume for an 18-week planning horizon. The forecast resulting from the ARIMA model has 11% mean absolute percentage error (MAPE) on average for a 6-week period (see Figure 3). For weekly adjustment, a patient census forecast made a week ahead of the adjusting week is used and has 9% MAPE on average in our empirical setting. Note, however, that such forecast for the adjusting week is not based on actual patient census but on the scenarios. Therefore, overall performance of the stochastic solution does not depend on the weekly adjusting forecasts but on the scenarios generated for the 18-week planning horizon.

## Appendix C: Value of the Stochastic Solution Resulting from 100 Patient Volume Scenarios

Table 11 reports the value of the stochastic solution (VSS) obtained by solving (TSSIP) with 100-patient volume scenarios. The 100 patient volume scenarios were generated from an arbitrary choice of 100 error vectors. We compare the deterministic solution, the stochastic solution from the 100-patient scenario, and the stochastic solution with the 1,000-patient scenario. Moreover, p-values were calculated from Welch's t-test in order to see whether the staffing cost from the 1,000-patient scenario is significantly lower than that from the 100-patient scenario. To achieve a greater statistical power, we evaluate the solutions using 12,768 error vectors available from our empirical study.

The stochastic solution from the 100-patient scenario results in the cost saving of 110 nursing hours a week on average. This is smaller than that from the 1,000-patient solution by average 20 nursing hours a week. Moreover, the difference of the staffing cost between the 100-patient scenario solution and the 1,000-patient scenario solution is statistically significant with a p-value $< 0.01$.

**Table 11** Staffing cost and value of the stochastic solutions resulting from the different number of scenarios. estimated using 12,768 error vectors

| | Deterministic Solution | | 100-Patient Scenario Solution | | | | 1000-Patient Scenario Solution | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Instance | Mean | Stdev | Mean | Stdev | VSS | | Mean | Stdev | VSS | | p-value |
| 20FEB2012 | 5360 | 375 | 5263 | 564 | 97 | 1.8% | 5240 | 508 | 119 | 2.2% | $< 0.01$ |
| 23JAN2012 | 5956 | 431 | 5825 | 654 | 132 | 2.2% | 5803 | 620 | 153 | 2.6% | $< 0.01$ |
| 26DEC2011 | 5384 | 310 | 5285 | 489 | 99 | 1.8% | 5262 | 444 | 121 | 2.3% | $< 0.01$ |
| 28NOV2011 | 5624 | 357 | 5516 | 527 | 108 | 1.9% | 5495 | 501 | 129 | 2.3% | $< 0.01$ |
| 31OCT2011 | 5823 | 448 | 5721 | 658 | 102 | 1.7% | 5699 | 602 | 124 | 2.1% | $< 0.01$ |
| 03OCT2011 | 5591 | 425 | 5475 | 630 | 116 | 2.1% | 5456 | 604 | 135 | 2.4% | 0.01 |
| 05SEP2011 | 5523 | 441 | 5424 | 651 | 98 | 1.8% | 5401 | 613 | 122 | 2.2% | $< 0.01$ |
| 08AUG2011 | 6154 | 467 | 6049 | 696 | 106 | 1.7% | 6028 | 658 | 126 | 2.0% | 0.01 |
| 11JUL2011 | 5973 | 434 | 5860 | 648 | 112 | 1.9% | 5843 | 590 | 130 | 2.2% | 0.01 |
| 13JUN2011 | 5933 | 404 | 5810 | 614 | 123 | 2.1% | 5790 | 575 | 143 | 2.4% | $< 0.01$ |
| 16MAY2011 | 6116 | 430 | 5994 | 648 | 122 | 2.0% | 5977 | 620 | 139 | 2.3% | 0.01 |
| 18APR2011 | 5993 | 439 | 5892 | 636 | 101 | 1.7% | 5875 | 600 | 118 | 2.0% | 0.01 |
| 21MAR2011 | 5668 | 387 | 5551 | 565 | 116 | 2.1% | 5531 | 539 | 136 | 2.4% | $< 0.01$ |
| 21FEB2011 | 6010 | 447 | 5909 | 654 | 101 | 1.7% | 5888 | 611 | 122 | 2.0% | $< 0.01$ |
| 24JAN2011 | 5758 | 401 | 5636 | 611 | 122 | 2.1% | 5618 | 554 | 139 | 2.4% | 0.01 |
| 27DEC2010 | 5615 | 381 | 5506 | 573 | 109 | 1.9% | 5485 | 516 | 131 | 2.3% | $< 0.01$ |
| 29NOV2010 | 5468 | 355 | 5364 | 530 | 104 | 1.9% | 5341 | 505 | 127 | 2.3% | $< 0.01$ |
| 01NOV2010 | 5563 | 353 | 5463 | 535 | 100 | 1.8% | 5442 | 476 | 121 | 2.2% | $< 0.01$ |
| 04OCT2010 | 5698 | 398 | 5579 | 585 | 119 | 2.1% | 5557 | 559 | 141 | 2.5% | $< 0.01$ |
| 06SEP2010 | 5636 | 333 | 5527 | 514 | 108 | 1.9% | 5504 | 489 | 132 | 2.3% | $< 0.01$ |
| Mean | 5742 | 401 | 5632 | 599 | 110 | 1.9% | 5612 | 559 | 130 | 2.3% | $< 0.01$ |

**Algorithm 3** Scheduling Pattern Generation

$ReqShifts \leftarrow$ the required number of shifts for a given shift type.

$ShiftLength \leftarrow$ the number of work hours per shift for a given shift type.

$Schedules \leftarrow \emptyset$

Call GeneratePatterns($Schedules, \emptyset, 0$)

**function** GeneratePatterns($Schedules, Shifts, StartTime$)

  **if** $|Shifts| = ReqShifts$ **then**

    $Schedules \leftarrow Schedules \cup \{Shifts\}$; **return**

  **end if**

  **for** $t = StartTime, \ldots, 168$ **do** ▷ 168 hours = 1 week.

    **if** $t$ is a valid shift start time **then**

      Call GeneratePatterns($Schedules, Shifts \cup \{t\}, StartTime + ShiftLength$)

    **end if**

  **end for**

**end function**

**Table 12** Computational results from the deterministic iStaff model

| | Root Node | | | Branch-and-Cut | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Instance | No. of Iterations | No. of Cuts | CPU Time (sec.) | No. of Cuts | No. of Nodes | Branching on $\phi$ | $\psi$ | $\chi$ | $x$ | CPU Time (sec.) | Total CPU Time (sec.) |
| 20FEB2012 | 136 | 30 | 10 | 5 | 0 | 0 | 0 | 0 | 0 | 2 | 11 |
| 23JAN2012 | 134 | 30 | 10 | 2 | 0 | 0 | 0 | 0 | 0 | 1 | 11 |
| 26DEC2011 | 150 | 30 | 11 | 4 | 0 | 0 | 0 | 0 | 0 | 1 | 12 |
| 28NOV2011 | 128 | 30 | 10 | 3 | 0 | 0 | 0 | 0 | 0 | 1 | 11 |
| 31OCT2011 | 131 | 30 | 10 | 4 | 0 | 0 | 0 | 0 | 0 | 2 | 11 |
| 03OCT2011 | 137 | 30 | 10 | 4 | 0 | 0 | 0 | 0 | 0 | 1 | 11 |
| 05SEP2011 | 196 | 30 | 15 | 2 | 0 | 0 | 0 | 0 | 0 | 1 | 16 |
| 08AUG2011 | 155 | 30 | 11 | 4 | 0 | 0 | 0 | 0 | 0 | 1 | 12 |
| 11JUL2011 | 154 | 30 | 12 | 3 | 0 | 0 | 0 | 0 | 0 | 1 | 13 |
| 13JUN2011 | 132 | 29 | 10 | 3 | 0 | 0 | 0 | 0 | 0 | 1 | 11 |
| 16MAY2011 | 173 | 30 | 13 | 7 | 90 | 6 | 7 | 3 | 75 | 12 | 25 |
| 18APR2011 | 164 | 28 | 12 | 11 | 10 | 5 | 2 | 2 | 2 | 4 | 16 |
| 21MAR2011 | 127 | 30 | 9 | 4 | 0 | 0 | 0 | 0 | 0 | 2 | 11 |
| 21FEB2011 | 156 | 30 | 12 | 4 | 0 | 0 | 0 | 0 | 0 | 1 | 13 |
| 24JAN2011 | 138 | 29 | 10 | 3 | 0 | 0 | 0 | 0 | 0 | 1 | 11 |
| 27DEC2010 | 132 | 30 | 9 | 4 | 0 | 0 | 0 | 0 | 0 | 1 | 10 |
| 29NOV2010 | 197 | 30 | 15 | 5 | 50 | 19 | 15 | 8 | 8 | 6 | 21 |
| 01NOV2010 | 120 | 29 | 9 | 12 | 50 | 13 | 4 | 12 | 22 | 8 | 17 |
| 04OCT2010 | 170 | 30 | 13 | 4 | 0 | 0 | 0 | 0 | 0 | 1 | 14 |
| 06SEP2010 | 134 | 30 | 10 | 4 | 0 | 0 | 0 | 0 | 0 | 1 | 12 |
| Mean | 148 | 30 | 11 | 5 | 10 | 2 | 1 | 1 | 5 | 3 | 13 |
| Stdev | 22 | 1 | 2 | 3 | 24 | 5 | 4 | 3 | 17 | 3 | 4 |

## Appendix D: Algorithm for Generating Scheduling Patterns

For generating scheduling patterns in our iStaff model, Algorithm 3 calls a recursive function GeneratePatterns for a given shift type and results in a set of schedules, *Schedules*, where each schedule, *Shifts*, consists of a set of valid start times.

## Appendix E: Computational Results from the Deterministic iStaff Model

We present computational results from the deterministic iStaff model that considers a single scenario based on the mean point forecast. A deterministic problem instance has 4,025 integer variables and 336 continuous variables and was solved by the proposed L-shaped method. Table 12 shows the computational results for 20 problem instances. The column “No. of Iterations” presents the number of times that the master problem was resolved. The column “No. of Cuts” presents the number of optimality cuts after aggregation at the end of Step 1 of Algorithm 2. The deterministic problems were solved to optimality in 13 CPU-seconds on average. Of the 20 problem instances, 16 were solved at the root node of the branch-and-cut tree as the CPLEX MIP optimizer found optimum solutions using its internal heuristic procedures.

**Table 13** Computational performance from generating outer linearization cuts only at incumbent solutions and generating outer linearization cuts at any feasible solutions

| | Generating Cuts Only at Incumbent Solutions | | | | Generating Cuts at Any Feasible Integer Solutions | | | |
|---|---|---|---|---|---|---|---|---|
| Instance | No. of Cuts | No. of Nodes | CPU Time (sec.) | Wallclock (sec.) | No. of Cuts | No. of Nodes | CPU Time (sec.) | Wallclock (sec.) |
| 20FEB2012 | 3323 | 294 | 2020 | 334 | 5647 | 552 | 4745 | 532 |
| 23JAN2012 | 3987 | 2118 | 10565 | 904 | 2000 | 318 | 3777 | 302 |
| 26DEC2011 | 1998 | 2131 | 7107 | 563 | 4369 | 1226 | 10620 | 703 |
| 28NOV2011 | 2314 | 926 | 4032 | 377 | 2147 | 210 | 2498 | 255 |
| 31OCT2011 | 3947 | 621 | 5728 | 538 | 7668 | 1020 | 7681 | 775 |
| 03OCT2011 | 2724 | 1159 | 5722 | 479 | 10302 | 768 | 7190 | 863 |
| 05SEP2011 | 2678 | 536 | 2953 | 340 | 17269 | 1382 | 18499 | 2002 |
| 08AUG2011 | 3471 | 1213 | 7746 | 661 | 2780 | 320 | 4362 | 353 |
| 11JUL2011 | 6931 | 655 | 5930 | 692 | 14381 | 1090 | 10729 | 1156 |
| 13JUN2011 | 3889 | 626 | 4556 | 453 | 8399 | 945 | 6651 | 791 |
| 16MAY2011 | 3372 | 1206 | 6127 | 555 | 15730 | 2237 | 17873 | 3293 |
| 18APR2011 | 2790 | 830 | 3315 | 361 | 3000 | 287 | 3883 | 294 |
| 21MAR2011 | 2508 | 800 | 3501 | 382 | 4000 | 920 | 3341 | 342 |
| 21FEB2011 | 2981 | 1146 | 5870 | 536 | 19891 | 989 | 14294 | 1987 |
| 24JAN2011 | 4601 | 777 | 3400 | 341 | 3539 | 789 | 5470 | 469 |
| 27DEC2010 | 2935 | 658 | 3560 | 368 | 8026 | 540 | 5850 | 624 |
| 29NOV2010 | 2868 | 292 | 1795 | 272 | 9484 | 982 | 5679 | 942 |
| 01NOV2010 | 2522 | 722 | 3019 | 343 | 5497 | 625 | 7345 | 805 |
| 04OCT2010 | 2692 | 1536 | 6372 | 538 | 3000 | 127 | 2487 | 233 |
| 06SEP2010 | 3963 | 111 | 1653 | 306 | 4322 | 646 | 5412 | 513 |
| Mean | 3325 | 918 | 4749 | 467 | 7573 | 799 | 7419 | 862 |
| Stdev | 1082 | 540 | 2258 | 158 | 5399 | 488 | 4713 | 760 |

## Appendix F: Computational Results from More Aggressively Generating Outer Linearization Cuts in the Branch-and-Cut Procedure

We compare computational results from generating outer linearization cuts only at incumbent solutions with those from generating the cuts at any feasible integer solutions. When the cuts were generated at any feasible integer solutions, average computation time increased from 4,749 to 7,419 CPU-seconds. The reason is that the size of the B&C node subproblem becomes larger with more cuts, which results in longer computation time spent per B&C node subproblem on average. Specifically, the number of cuts generated at any feasible integer solutions was nearly twice that generated only at incumbent solutions. By generating the cuts at any feasible integer solutions, the number of nodes solved in the B&C tree was reduced because of tighter lower bounds.

## Appendix G: Computational Results from Multicut Purge Approach

We compare computational results from two different multicut approaches. One approach purges the cuts whose dual multipliers equal zero, while the other aggregates these cuts as presented in Section 4.2. Results are given in Table 14. Two approaches resulted in the nearly same computational performances for 16 of the 20 instances. The multicut aggregation approach generated

**Table 14** Computational performance resulting from multicut approach with cut purge and cut aggregation

| | Multicut Purge | | | | | Multicut Aggregation | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Instance | No. of Cuts | No. of Nodes | Root Node CPU Time (sec.) | B&C CPU Time (sec.) | Total CPU Time (sec.) | No. of Cuts | No. of Nodes | Root Node CPU Time (sec.) | B&C CPU Time (sec.) | Total CPU Time (sec.) |
| 20FEB2012 | 4352 | 294 | 6452 | 2002 | 8454 | 4352 | 294 | 6384 | 2020 | 8405 |
| 23JAN2012 | 5016 | 2118 | 5184 | 10638 | 15822 | 5016 | 2118 | 5168 | 10565 | 15734 |
| 26DEC2011 | 3027 | 2132 | 5913 | 6987 | 12900 | 3027 | 2131 | 5940 | 7107 | 13047 |
| 28NOV2011 | 3343 | 926 | 5294 | 4113 | 9407 | 3343 | 926 | 5324 | 4032 | 9356 |
| 31OCT2011 | 5972 | 689 | 4778 | 6166 | 10944 | 4976 | 621 | 4784 | 5728 | 10512 |
| 03OCT2011 | 3753 | 1156 | 5069 | 5691 | 10760 | 3753 | 1159 | 5004 | 5722 | 10726 |
| 05SEP2011 | 3707 | 535 | 4787 | 2887 | 7674 | 3707 | 536 | 4788 | 2953 | 7740 |
| 08AUG2011 | 4500 | 1212 | 4606 | 6405 | 11011 | 4500 | 1213 | 4576 | 7746 | 12322 |
| 11JUL2011 | 7737 | 887 | 5141 | 7226 | 12367 | 7960 | 655 | 5164 | 5930 | 11095 |
| 13JUN2011 | 4918 | 626 | 5592 | 4581 | 10173 | 4918 | 626 | 5544 | 4556 | 10100 |
| 16MAY2011 | 4401 | 1205 | 5182 | 5939 | 11121 | 4401 | 1206 | 5143 | 6127 | 11270 |
| 18APR2011 | 3819 | 830 | 5037 | 3547 | 8584 | 3819 | 830 | 4730 | 3315 | 8045 |
| 21MAR2011 | 3537 | 800 | 5199 | 3766 | 8965 | 3537 | 800 | 4785 | 3501 | 8286 |
| 21FEB2011 | 5006 | 1214 | 5315 | 6929 | 12244 | 4010 | 1146 | 4875 | 5870 | 10745 |
| 24JAN2011 | 8621 | 5874 | 5562 | 106371 | 111933 | 5630 | 777 | 5097 | 3400 | 8497 |
| 27DEC2010 | 3964 | 658 | 5802 | 3576 | 9378 | 3964 | 658 | 5718 | 3560 | 9278 |
| 29NOV2010 | 3897 | 292 | 5632 | 1846 | 7478 | 3897 | 292 | 5423 | 1795 | 7218 |
| 01NOV2010 | 3551 | 724 | 5582 | 2988 | 8570 | 3551 | 722 | 5609 | 3019 | 8628 |
| 04OCT2010 | 3721 | 1536 | 5281 | 6480 | 11761 | 3721 | 1536 | 5227 | 6372 | 11599 |
| 06SEP2010 | 4992 | 111 | 5900 | 1639 | 7538 | 4992 | 111 | 5937 | 1653 | 7590 |
| Mean | 4592 | 1191 | 5365 | 9989 | 15354 | 4354 | 918 | 5261 | 4749 | 10010 |
| Stdev | 1472 | 1225 | 446 | 22798 | 22830 | 1082 | 540 | 477 | 2258 | 2144 |

fewer nodes in the branch-and-bound tree for instances 31OCT2011, 11JUL2011, 21FEB2011, and 24JAN2011. This resulted in the B&C CPU time reduction for these problems. In particular, for the 24JAN2011 instance, the B&C CPU time was reduced by a factor of 31.

## Appendix H: Computational Results from the Standard Branching Strategy

In Table 15 we report computational results from the standard branching strategy that considers branching on original variables $\mathbf{x}$ only. The computational performances are not significantly different from those that first use tender variable branching (see Table 8).

**Table 15** Computational performance resulting from branching on tender variable with priority over original variable

| Instance | No. of Nodes | No. of Feasible Solutions | No. of Branching on $\mathbf{x}$ | B&C CPU Time (sec.) | Total CPU Time (sec.) |
|---|---|---|---|---|---|
| 20FEB2012 | 1303 | 69 | 1307 | 12803 | 19155 |
| 23JAN2012 | 1111 | 53 | 1114 | 11220 | 16373 |
| 26DEC2011 | 1978 | 54 | 1972 | 18857 | 24787 |
| 28NOV2011 | 1808 | 74 | 1812 | 14667 | 19980 |
| 31OCT2011 | 1915 | 75 | 1890 | 25221 | 30041 |
| 03OCT2011 | 1784 | 46 | 1783 | 14760 | 19806 |
| 05SEP2011 | 1418 | 40 | 1417 | 7824 | 12579 |
| 08AUG2011 | 1404 | 37 | 1403 | 8732 | 13352 |
| 11JUL2011 | 2409 | 65 | 2408 | 18311 | 23506 |
| 13JUN2011 | 295 | 30 | 298 | 4372 | 10063 |
| 16MAY2011 | 1757 | 40 | 1748 | 12239 | 17391 |
| 18APR2011 | 1592 | 54 | 1595 | 9837 | 14856 |
| 21MAR2011 | 1544 | 60 | 1546 | 12346 | 17519 |
| 21FEB2011 | 2001 | 82 | 2002 | 17733 | 23074 |
| 24JAN2011 | 1925 | 99 | 1934 | 34151 | 39721 |
| 27DEC2010 | 1824 | 67 | 1828 | 16804 | 22527 |
| 29NOV2010 | 946 | 40 | 949 | 8271 | 13864 |
| 01NOV2010 | 2348 | 76 | 2350 | 18569 | 24174 |
| 04OCT2010 | 722 | 41 | 730 | 8544 | 13754 |
| 06SEP2010 | 1059 | 51 | 1061 | 8244 | 14067 |
| Mean | 1557 | 58 | 1557 | 14175 | 19529 |
| Stdev | 535 | 18 | 533 | 6897 | 6947 |

## Acknowledgments

The research was partially supported by grant NSF-CMMI-0928936 and ONR-N00014210051.

## References

Aardal, Karen, Arjen K Lenstra, HW Lenstra. 2002. Hard equality constrained integer knapsacks. *IPCO*. Springer, 350–366.

Abernathy, William J, Nicholas Baloff, John C Hershey, Sten Wandel. 1973. A three-stage manpower planning and scheduling model—a service-sector example. *Operations Research* **21**(3) 693–711.

Ahmed, Shabbir, Mohit Tawarmalani, Nikolaos V Sahinidis. 2004. A finite branch-and-bound algorithm for two-stage stochastic integer programs. *Mathematical Programming* **100**(2) 355–377.

Bard, Jonathan F, David P Morton, Yong Min Wang. 2007. Workforce planning at USPS mail processing and distribution centers using stochastic optimization. *Annals of Operations Research* **155**(1) 51–78.

Bard, Jonathan F, Hadi W Purnomo. 2004. Real-time scheduling for nurses in response to demand fluctuations and personnel shortages. *Proceedings of the 5th International Conference on the Practice and Theory of Automated Timetabling*. 67–87.

Bard, Jonathan F, Hadi W Purnomo. 2005a. A column generation-based approach to solve the preference scheduling problem for nurses with downgrading. *Socio-Economic Planning Sciences* **39**(3) 193–213.

Bard, Jonathan F, Hadi W Purnomo. 2005b. Hospital-wide reactive scheduling of nurses with preference considerations. *IIE Transactions* **37**(7) 589–608.

Bard, Jonathan F, Hadi W Purnomo. 2005c. Preference scheduling for nurses using column generation. *European Journal of Operational Research* **164**(2) 510–534.

Bard, Jonathan F, Hadi W Purnomo. 2005d. Short-term nurse scheduling in response to daily fluctuations in supply and demand. *Health Care Management Science* **8**(4) 315–324.

Bard, Jonathan F, Hadi W Purnomo. 2007. Cyclic preference scheduling of nurses using a Lagrangian-based heuristic. *Journal of Scheduling* **10**(1) 5–23.

Benders, Jacques F. 1962. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik* **4**(1) 238–252.

Birge, John R. 1982. The value of the stochastic solution in stochastic linear programs with fixed recourse. *Mathematical Programming* **24**(1) 314–325.

Birge, John R, Francois V Louveaux. 1988. A multicut algorithm for two-stage stochastic linear programs. *European Journal of Operational Research* **34**(3) 384–392.

Birge, John R, François V Louveaux. 1997. *Introduction to Stochastic Programming*. Springer.

Bodur, Merve, James Luedtke. 2014. Integrated Service System Staffing and Scheduling via Stochastic Integer Programming. *www.optimization-online.org* .

Burke, Edmund K, Patrick De Causmaecker, Greet Vanden Berghe, Hendrik Van Landeghem. 2004. The state of the art of nurse rostering. *Journal of Scheduling* **7**(6) 441–499.

Burke, Edmund K, Jingpeng Li, Rong Qu. 2012. A pareto-based search methodology for multi-objective nurse scheduling. *Annals of Operations Research* **196**(1) 91–109.

Carøe, Claus C, Jørgen Tind. 1997. A cutting-plane approach to mixed 0–1 stochastic integer programs. *European Journal of Operational Research* **101**(2) 306–316.

Cheang, Brenda, Hao Li, Andrew Lim, Brian Rodrigues. 2003. Nurse rostering problems–a bibliographic survey. *European Journal of Operational Research* **151**(3) 447–460.

Cornuéjols, Gérard, Leo Liberti, Giacomo Nannicini. 2011. Improved strategies for branching on general disjunctions. *Mathematical Programming* **130**(2) 225–247.

CPLEX, IBM ILOG. 2009. V12. 1: Users manual for CPLEX. *International Business Machines Corporation* **46**(53) 157.

Davis, Ashley, Mark S. Daskin, Sanjay Mehrotra, Jane Holl. 2013. Nurse staffing to reduce costs and enhance patient safety. *To appear in Asia-Pacific Journal of Operations Research* .

Easton, Fred F, Nashat Mansour. 1999. A distributed genetic algorithm for deterministic and stochastic labor scheduling problems. *European Journal of Operational Research* **118**(3) 505–523.

Easton, Fred F, Donald F Rossin. 1996. A stochastic goal program for employee scheduling*. *Decision Sciences* **27**(3) 541–568.

Eddelbuettel, D, R François. 2010. RInside: C++ classes to embed R in C++ applications. *R package version 0.2* **3**.

Gade, Dinakar, Simge Küçükyavuz, Suvrajeet Sen. 2012. Decomposition algorithms with parametric gomory cuts for two-stage stochastic integer programs. *Mathematical Programming* 1–26.

Jaumard, Brigitte, Frédéric Semet, Tsevi Vovor. 1998. A generalized linear programming model for nurse scheduling. *European Journal of Operational Research* **107**(1) 1–18.

Kao, Edward PC, Maurice Queyranne. 1985. Budgeting costs of nursing in a hospital. *Management Science* **31**(5) 608–621.

Karamanov, Miroslav, Gérard Cornuéjols. 2011. Branching on general disjunctions. *Mathematical Programming* **128**(1-2) 403–436.

Kim, Kibaek, Changhyeok Lee, Kevin O'Leary, Shannon Rosenauer, Sanjay Mehrotra. 2014. Predicting patient volumes in hospital medicine: A comparative study of different time series forecasting methods. Tech. rep., Northwestern University.

Kong, Nan, Andrew J. Schaefer, Shabbir Ahmed. 2013. Totally unimodular stochastic programs. *Mathematical Programming* **138**(1-2) 1–13.

Kong, Nan, Andrew J Schaefer, Brady Hunsaker. 2006. Two-stage integer programs with stochastic right-hand sides: a superadditive dual approach. *Mathematical Programming* **108**(2-3) 275–296.

Laporte, Gilbert, François V Louveaux. 1993. The integer l-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters* **13**(3) 133–142.

Lenstra, Hendrik W. 1983. Integer programming with a fixed number of variables. *Mathematics of Operations Research* 538–548.

Louveaux, François V, Rüdiger Schultz. 2003. Stochastic integer programming. *Handbooks in Operations Research and Management Science* **10** 213–266.

Lovász, László, Herbert E Scarf. 1992. The generalized basis reduction algorithm. *Mathematics of Operations Research* **17**(3) 751–764.

Maenhout, Broos, Mario Vanhoucke. 2013a. Analyzing the nursing organizational structure and process from a scheduling perspective. *Health Care Management Science* 1–20.

Maenhout, Broos, Mario Vanhoucke. 2013b. An integrated nurse staffing and scheduling analysis for longer-term nursing staff allocation problems. *Omega* **41**(2) 485–499.

Mahajan, Ashutosh, Theodore K Ralphs. 2009. Experiments with branching using general disjunctions. *Operations Research and Cyber-Infrastructure*. Springer, 101–118.

Mehrotra, Sanjay, Kuo-Ling Huang. 2013. On implementing a general disjunctive branching algorithm using lattice basis reduction for mixed integer convex programming. Tech. rep., Northwestern University.

Mehrotra, Sanjay, Zhifeng Li. 2011. Branching on hyperplane methods for mixed integer linear and convex programming using adjoint lattices. *Journal of Global Optimization* **49**(4) 623–649.

Miller, Andrew J, Laurence A Wolsey. 2003. Tight formulations for some simple mixed integer programs and convex objective integer programs. *Mathematical Programming* **98**(1-3) 73–88.

Nemhauser, George L, Laurence A Wolsey. 1988. *Integer and Combinatorial Optimization*, vol. 18. Wiley New York.

Owen, Jonathan H, Sanjay Mehrotra. 2001. Experimental results on using general disjunctions in branch-and-bound for general-integer linear programs. *Computational Optimization and Applications* **20**(2) 159–170.

Parr, D, Jonathan M Thompson. 2007. Solving the multi-objective nurse scheduling problem with a weighted cost function. *Annals of Operations Research* **155**(1) 279–288.

Pratt, John Winsor, Howard Raiffa, Robert Schlaifer. 1995. *Introduction to Statistical Decision Theory*. MIT press.

Punnakitikashem, Prattana, Jay M. Rosenberger, Deborah F. Buckley-Behan. 2008. Stochastic programming for nurse assignment. *Computational Optimization and Applications* **40**(3) 321–349.

Punnakitikashem, Prattana, Jay M. Rosenberger, Deborah F. Buckley-Behan. 2013. A stochastic programming approach for integrated nurse staffing and assignment. *IIE Transactions* **45**(10) 1059–1076.

Schultz, Rüdiger, Leen Stougie, Maarten H Van Der Vlerk. 1998. Solving stochastic programs with integer recourse by enumeration: A framework using Gröbner basis. *Mathematical Programming* **83**(1-3) 229–252.

Sen, Suvrajeet. 2005. Algorithms for stochastic mixed-integer programming models. *Handbooks in Operations Research and Management Science* **12** 515–558.

Sen, Suvrajeet, Julia L Higle. 2005. The $C^3$ theorem and a $D^2$ algorithm for large scale stochastic mixed-integer programming: set convexification. *Mathematical Programming* **104**(1) 1–20.

Sen, Suvrajeet, Hanif D Sherali. 2006. Decomposition with branch-and-cut approaches for two-stage stochastic mixed-integer programming. *Mathematical Programming* **106**(2) 203–223.

Sherali, Hanif D, Barbara MP Fraticelli. 2002. A modification of Benders' decomposition algorithm for discrete subproblems: An approach for stochastic programs with integer recourse. *Journal of Global Optimization* **22**(1-4) 319–342.

Sherali, Hanif D, Xiaomei Zhu. 2006. On solving discrete two-stage stochastic programs having mixed-integer first-and second-stage variables. *Mathematical Programming* **108**(2-3) 597–616.

Trukhanov, Svyatoslav, Lewis Ntaimo, Andrew Schaefer. 2010. Adaptive multicut aggregation for two-stage stochastic linear programs with recourse. *European Journal of Operational Research* **206**(2) 395–406.

Van Slyke, Richard M, Roger Wets. 1969. L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics* **17**(4) 638–663.

Venkataraman, R, MJ Brusco. 1996. An integrated analysis of nurse staffing and scheduling policies. *Omega* **24**(1) 57–71.

Woodall, Jonathan C, Tracy Gosselin, Amy Boswell, Michael Murr, Brian T Denton. 2013. Improving patient access to chemotherapy treatment at Duke Cancer Institute. *Interfaces* **43**(5) 449–461.

Wright, P Daniel, Kurt M Bretthauer. 2010. Strategies for addressing the nursing shortage: Coordinated decision making and workforce flexibility. *Decision Sciences* **41**(2) 373–401.

Wright, P Daniel, Kurt M Bretthauer, Murray J Côté. 2006. Reexamining the nurse scheduling problem: Staffing ratios and nursing shortages. *Decision Sciences* **37**(1) 39–70.

Wright, P Daniel, Stephen Mahar. 2013. Centralized nurse scheduling to simultaneously improve schedule cost and nurse satisfaction. *Omega* **41**(6) 1042–1052.

Zhu, Xiaomei, Hanif D Sherali. 2007. Two-stage workforce planning under demand fluctuations and uncertainty. *Journal of the Operational Research Society* **60**(1) 94–103.