Mixed-Integer Rounding Enhanced Benders Decomposition for Multiclass Service System Staffing and Scheduling with Arrival Rate Uncertainty

Merve Bodur, James R. Luedtke
Department of Industrial and Systems Engineering, University of Wisconsin, Madison, Wisconsin 53706
mbodur@wisc.edu, jrluedt1@wisc.edu

November 13, 2014

We study server scheduling in multiclass service systems under stochastic uncertainty in the customer arrival volumes. Common practice in such systems is to first identify staffing levels, and then determine schedules for the servers that cover these targets. We propose a new stochastic integer programming model that integrates these two decisions, which can yield lower scheduling costs by exploiting the presence of alternative server configurations that yield similar quality-of-service. We find that a branch-and-cut algorithm based on Benders decomposition may fail due to the weakness of the relaxation bound. We propose a novel application of mixed-integer rounding to improve the Benders cuts used in this algorithm, a technique that is applicable to any stochastic integer program with integer first-stage decision variables. Numerical examples illustrate the computational efficiency of the proposed approach and the potential benefit of solving the integrated model compared to considering the staffing and scheduling problems separately.

Key words: Service systems scheduling; stochastic integer programming; mixed-integer rounding

1. Introduction

Workforce scheduling is an important aspect of service system operations, and becomes particularly complicated when considering systems that have heterogeneous sets of servers and customers, and which are faced with errors in the forecasts of customer arrival rates. An important example of large-scale service systems facing these challenges is call centers. The fundamental challenge in staffing and scheduling a service system is to maintain an acceptable quality-of-service (QoS) at minimum cost, where quality-of-service can be measured in many ways, such as average customer waiting time, or the proportion of customers who abandon the system without service.

The process of staffing and scheduling servers can be categorized into four steps (Atlason et al. 2008):

1. Forecasting: Estimate customer arrival rates over the planning horizon.
2. Staffing: Determine the minimum number of servers needed in each period to meet quality-of-service targets.

3. Shift scheduling: Select staff shifts to meet staffing levels.
4. Rostering: Assign employees to the shifts.

In a *multiskill multiclass* system, containing multiple pools of servers and customer classes having different characteristics, a further important step is the real-time routing of customers to available servers who have the ability to serve that customer class during the operation of the system. This is known as *skills-based routing*. There are many routing rules in the literature, see e.g., Mandelbaum and Stolyar (2004), Atar et al. (2005), Gurvich and Whitt (2007), Dai and Tezcan (2008), and Stolyar and Tezcan (2010), and the choice of routing rule is important for achieving good system performance. In this work, we assume the routing rule is fixed (we use the "shadow routing" rule proposed by Stolyar and Tezcan (2010)) and focus on the staffing and shift scheduling problems.

A common practice in service system management is to conduct the staffing and shift scheduling steps sequentially. Specifically, the planning horizon is divided into periods with lengths such as 15 or 30 minutes, and staffing levels are determined for the agents in each of these periods that balance the staffing costs with the quality-of-service. Then, with these staffing levels fixed, an integer programming model can be solved to select minimum cost shifts that meet these target staffing levels.

In this paper, we consider a multiskill multiclass system with uncertainty in customer arrival rates, and study the *integrated* problem of staffing and shift scheduling simultaneously. In a multiskill system, there may be multiple different combinations of server staffing levels that can yield an adequate quality-of-service in any given time period. The key advantage of our proposed model is that it is able to exploit this flexibility to choose staffing levels in different time periods in such a way that the scheduling cost required to meet the selected staffing levels is minimized. The model we introduce for this problem is a two-stage stochastic integer programming problem, which we find can be difficult to solve using current methods. Therefore, we also introduce a general technique for using mixed-integer rounding (MIR) inequalities (Nemhauser and Wolsey 1988) within a Benders decomposition algorithm for solving this model, and find that it is very effective at improving the performance of Benders decomposition. While this technique is motivated by the server scheduling application, it can be applied to any two-stage stochastic integer program with integer first-stage variables.

Because a forecast can never predict with certainty the arrival rates over the planning horizon, modern forecasting procedures provide both a point estimate of the arrival rate during a period and a distribution of the possible error from the point estimate. Because it is generally not possible to change the staffing levels in real-time, the forecast error can lead to overstaffing or understaffing. As a result, it is important to consider the uncertainty in arrival rate forecasts when making staffing and scheduling decisions. Thus, our proposed model also accounts for the forecast error

and takes advantage of the estimate of its distribution, leading to a stochastic integer programming formulation.

To the best of our knowledge, Cezik and L'Ecuyer (2008) and Avramidis et al. (2010) are the only authors to have previously studied this integrated problem of staffing and scheduling in a multiskill multiclass setting with uncertain arrival rates. The key difference between our work and these works is that the optimization model we propose uses a linear program (LP) to approximate the quality-of-service of a set of staffing levels in a period, whereas they use simulation. We also suggest using simulation, but only to evaluate a set of candidate solutions, thus significantly reducing the computational burden. Our numerical experiments show that while simulation gives more accurate estimates of quality-of-service, the LP approximation is a good proxy for the quality-of-service because there is a near perfect correlation between the quality-of-service estimates obtained from simulation and the LP approximation we use. Thus, we choose to use the LP approximation to avoid the computationally expensive requirement to generate approximate gradients via simulation as part of the solution procedure. Furthermore, the LP approximation guarantees that the component of the cost function that approximates the cost of quality-of-service is convex, enabling this cost function to be correctly modeled using cuts. In contrast, the simulation-based approaches are based on cuts derived from subgradients of functions that are not guaranteed to be convex (or even pseudoconvex), and therefore run the risk of cutting off promising solutions.

The LP that we use for estimating the quality-of-service impact of a staffing profile within a period is based on the static and fluid approximation introduced in Harrison and Zeevi (2005) and analyzed in Bassamboo et al. (2006). Harrison and Zeevi (2005) proposed a stochastic LP that can be used to determine staffing levels within a single period that optimally balances staffing cost and abandonment costs, as measured by a weighted sum of customer abandonments. In particular, for a given set of arrival rates and server levels, the abandonment cost is determined by seeking a fluid allocation of the servers to customers such that the weighted proportion of customer arrivals that cannot be met by the allocation is minimized. Bassamboo et al. (2006) showed that there exists a dynamic skills-based routing policy such that using this routing policy in combination with the staffing levels provided from the stochastic LP yields a solution that is asymptotically optimal in a limiting regime characterized by high volumes, short service times, and impatient customers. Thus, one can conclude that the staffing solutions provided by the stochastic LP are high quality for systems exhibiting these characteristics. Furthermore, in our numerical results we find that even for a system that does not exhibit high volumes, the LP approximation is highly correlated with the fraction of abandonments as estimated by simulation, and therefore provides a good basis for optimizing shift schedules.

The model that we present is a stochastic integer programming (IP) model, where integer decision variables are used to determine how many servers are assigned to each possible schedule, and the arrival rates (or equivalently, for a given period, counts) are modeled as random variables. As described in the previous paragraphs, given the server staffing levels associated with a schedule and a given scenario of customer counts, the abandonment cost is evaluated using a LP as in Bassamboo et al. (2006). The objective is to minimize the sum of the scheduling costs and the abandonment costs. As in Atlason et al. (2004) and Cezik and L'Ecuyer (2008), we first use sample average approximation (Kleywegt et al. 2002) to construct an approximation of the problem that has a finite number of customer count scenarios, which converts the problem into a large-scale mixed-integer program.

We analyze the relative gap between the proposed stochastic IP formulation and its LP relaxation, obtained by ignoring the integrality restrictions on the scheduling variables. We show that under assumptions on the form of the scheduling constraints that are often satisfied, or approximately satisfied, in real systems, this gap goes to zero as the arrival counts grow. Thus, for high volume systems, the proposed model can be approximated well as a stochastic LP, and thus can be readily solved by Benders decomposition.

When the LP relaxation is not an adequate approximation of the stochastic IP, the stochastic IP can be solved by reformulation as a large-scale mixed-integer program known as the extensive form in stochastic programming. Unfortunately, solving the extensive form can take prohibitively long if the number of scenarios is large simply due to its size. The size of the formulation can be overcome by embedding the Benders decomposition algorithm within a branch-and-cut algorithm. However, we also find that Benders decomposition fails to solve instances of reasonable size because the LP relaxation may not be a close enough approximation to the true objective value, leading to very large branch-and-bound search trees. To overcome this drawback, but preserve the decomposition nature of the Benders decomposition algorithm, we propose a novel application of the mixed-integer rounding (MIR) procedure of Nemhauser and Wolsey (1988) to improve the cuts used within the Benders decomposition algorithm. The basic idea of this approach is to use the integrality of the first-stage decision variables (the staffing levels of each server type) to strengthen the cuts obtained in the Benders decomposition algorithm. In addition to being the first general application of MIR within Benders decomposition, the MIR inequality that we derive is slightly stronger than the one stated in Nemhauser and Wolsey (1988). As an example of the general applicability of this approach, this technique could potentially be used to improve the computational performance of the simulation-based cutting plane methods of Cezik and L'Ecuyer (2008) and Avramidis et al. (2010).

We conduct a numerical study to illustrate the benefit of the proposed integrated model and the computational efficiency of the proposed MIR-enhanced Benders decomposition solution approach. We compare the solutions obtained with our integrated model to those obtained using a simple two-step heuristic, which mimics the practice of considering the staffing and scheduling problems separately (Atlason et al. 2004, Bhulai et al. 2008, Cezik and L'Ecuyer 2008, Avramidis et al. 2010). We find that by using the integrated model it is possible to find schedules that provide similar quality of service and reduce costs by 1-5%, or reduce abandonment percentages by 0.3-1.3% while keeping the scheduling costs the same. This reinforces similar findings of Avramidis et al. (2010).

We also find that while the extensive form and standard Benders decomposition algorithms cannot solve the integrated model within our one-hour time limit, using our proposed MIR-strengthened Benders cuts leads to a decomposition algorithm that is able to solve all of our test instances in an hour.

The remainder of this paper is organized as follows. Section 2 briefly reviews related literature. Section 3 describes our proposed formulation. Section 4 describes the proposed methodology, including the standard Benders decomposition algorithm, and our MIR-enhanced version; and presents our propositions about the vanishing integrality gap of the model for high volume systems. We provide our numerical illustrations in Section 5.

2. Literature review

There is a huge literature on every step of the workforce management process, especially on call centers. A comprehensive survey on modeling and analysis of call centers can be found in Gans et al. (2003). We restrict our review to the papers addressing staffing and/or scheduling steps.

There is a large variety of methodologies to deal with staffing and scheduling problems such as queueing, simulation and optimization approaches. As stated in the survey by Aksin et al. (2007) of the recent literature on call centers, the survey by Koole and Mandelbaum (2002) focuses on queueing models for call centers; L'Ecuyer (2006) focuses on optimization problems for call centers; and Koole and Pot (2006) focus on multiskill call centers. Also, Aksin et al. (2010) primarily review multiskill call centers.

When staffing a system with a single server and customer type and deterministic arrival rates, Markovian queueing models, in particular the Erlang-C formula for the M/M/N model, are an important tool. The *square-root staffing rule* has also been extensively analyzed in this context (Jennings et al. 1996, Kolesar and Green 1998, Garnett et al. 2002, Borst et al. 2004, Feldman et al. 2008, Mandelbaum and Zeltyn 2009). One of the earliest studies which consider staffing and scheduling steps together in the single-server single-customer type case is by Henderson and Mason (1998) where an integer program is introduced for the integrated problem. Service level functions are estimated via simulation and the model is refined by iterative addition of cuts.

Recently, there has been significant research investigating staffing and scheduling problems while considering the randomness and uncertainty in arrival rate forecasts. Many papers have studied the staffing and/or scheduling problems for the case of only a single server and customer class (Jennings et al. 1996, Thompson 1997, Bassamboo et al. 2006, Whitt 2006, Robbins and Harrison 2008, Ingolfsson et al. 2010). In Atlason et al. (2004), which extends Henderson and Mason (1998), a relaxation of a sample average version of the integrated problem is solved via a combination of integer programming and simulation. Cuts are added to the integer model based on approximate *subgradients* of the service level as a function of staffing levels in each period. This method relies on the assumption that service level functions are concave in the staffing vector. Atlason et al. (2008) provide a modification of this approach that only requires the service level functions to be *pseudoconcave* in the staffing vector. Robbins and Harrison (2010) and Kim and Mehrotra (2014) formulate the integrated problem as a stochastic integer program. Kim and Mehrotra (2014) analyze the structure of their mixed-integer recourse problem and describe its convex hull by using MIR inequalities. Their use of MIR inequalities is specialized to the structure of their model, while the novel MIR approach we propose is applicable to any two-stage stochastic program with integer first-stage variables.

For multiskill/multiclass systems almost all of the papers considering arrival rate uncertainty study only the staffing problem (Harrison and Zeevi 2005, Pot et al. 2008, Bassamboo and Zeevi 2009, Avramidis et al. 2009, Feldman and Mandelbaum 2010, Gurvich et al. 2010). Bhulai et al. (2008) solve the staffing and scheduling problems separately in their two-step algorithm. Cezik and L'Ecuyer (2008) extend the simulation-based cutting plane method of Atlason et al. (2004) to the multiskill setting, but they present numerical results only for a single period staffing problem. Avramidis et al. (2010) also solve the staffing and the scheduling problems simultaneously with a simulation-based cutting plane approach that extends Cezik and L'Ecuyer (2008), and present a comparison with the two-step method of Bhulai et al. (2008) that demonstrates that solving the integrated model indeed can yield significant cost savings.

3. Problem formulation

Let $\mathcal{I}$ be the set of server types, $\mathcal{J}$ be the set of customer classes and $\mathcal{T}$ be the set of time periods. $\mathcal{S}_i$ denotes the set of possible schedules for $i \in \mathcal{I}$. For the cardinalities of the sets, we use $I = |\mathcal{I}|,\ J = |\mathcal{J}|,\ T = |\mathcal{T}|,\ S_i = |\mathcal{S}_i|$. Let a_{ist} be the binary parameter representing the availability of server type i, in period t for schedule s; $a_{ist} = 1$ if schedule s includes time t, and $a_{ist} = 0$ otherwise. The parameter c_{is} is the cost of schedule $s \in \mathcal{S}_i$ for server type $i \in \mathcal{I}$. The number of customers of type $j \in \mathcal{J}$ who arrive in period $t \in \mathcal{T}$ is modeled as a nonnegative random variable Λ_{jt}. The set of random variables $\Lambda_{jt}, j \in \mathcal{J}, t \in \mathcal{T}$, may be correlated.

We formulate the joint staffing and scheduling problem as a two-stage stochastic integer program. In the first stage, before observing the uncertain arrival volumes Λ_{jt}, we decide the number of each server type who is assigned to each schedule, which then determines the staffing levels of the server types in each period. We introduce decision variables y_{it} to represent the number of servers of type $i \in \mathcal{I}$ to staff at time $t \in \mathcal{T}$, and decision variables x_{is} for $i \in \mathcal{I}$, $s \in \mathcal{S}_i$, to represent the number of servers of type i who are assigned to schedule s. After the customer counts during each period are observed, a second-stage problem allocates the available servers to the (now known) customers in order to minimize the number of customers that are not served (abandoned customers). Our problem formulation follows as:

$$\min_{x,y} \sum_{i\in\mathcal{I}} \sum_{s\in\mathcal{S}_i} c_{is}x_{is} + \alpha \sum_{t\in\mathcal{T}} \mathbb{E}_{\Lambda}[Q_t(y,\Lambda)] \tag{1a}$$

$$\text{s.t.} \sum_{s\in\mathcal{S}_i} a_{ist}x_{is} \geq y_{it}, \qquad i \in \mathcal{I},\ t \in \mathcal{T}, \tag{1b}$$

$$x \in X, \tag{1c}$$

$$x \in \mathbb{Z}_+^{IS},\ y \in \mathbb{Z}_+^{IT}. \tag{1d}$$

The objective function (1a) minimizes the weighted sum of scheduling cost and expected abandonment cost. Since this problem is a bi-objective problem (minimize scheduling cost and abandonment cost), we multiply the second component of the objective with a positive parameter α to balance these two different costs. Constraints (1b) enforce schedules to be chosen in such a way that all staffing levels are met. In constraints (1c), X denotes the feasible region formed by other constraints on the schedule variables, such as constraints on the maximum number of a server type that are available or a limit on the number of part-time schedules that can be used. Finally, constraints (1d) require x and y variables to take on nonnegative integer values, respectively.

The recommended use of this model is to vary the parameter α to obtain a set of solutions on the efficient frontier between the two objectives of cost and quality-of-service. Then, the abandonment rate (or other quality-of-service metric) should be estimated for these solutions using simulation. The decision-maker can then choose a schedule from among these candidates based on her desired trade-off between cost and quality-of-service. See Section 4.6 for more details.

$Q_t(\cdot,\cdot)$ is the function that we use to measure the quality-of-service, measured as a weighted sum of the unsatisfied (abandoned) customers. Specifically, for a given period $t \in \mathcal{T}$, $Q_t(\cdot,\cdot)$ is defined as

$$Q_t(y,\lambda) := \min_{v,w} \sum_{j\in\mathcal{J}} p_j w_j \tag{2a}$$

$$\text{s.t.} \sum_{i\in\mathcal{I}} \mu_{ij}v_{ij} + w_j = \lambda_{jt},\ \ j \in \mathcal{J}, \tag{2b}$$

$$\sum_{j\in\mathcal{J}} v_{ij} \le y_{it}, \qquad i \in \mathcal{I}, \tag{2c}$$

$$v \in \mathbb{R}^{IJ}_+,\ w \in \mathbb{R}^{I}_+. \tag{2d}$$

Problem (2) is the second stage model which calculates the abandonment cost, $Q_t(y, \lambda)$, for period $t \in \mathcal{T}$ with fixed staffing levels y and customer count vector λ. In problem (2), v_{ij} represents the number of type $i \in \mathcal{I}$ servers allocated to customer class $j \in \mathcal{J}$ and w_j denotes the number of type $j \in \mathcal{J}$ customers that are not served. The objective function (2a) minimizes the total penalty cost, where $p_j > 0$ is the penalty for an unserved customer of class $j \in \mathcal{J}$. The parameter $\mu_{ij} \ge 0$ represents the service rate of server type $i \in \mathcal{I}$ when serving a customer of class $j \in \mathcal{J}$; $\mu_{ij} > 0$ if and only if server type i can serve customer class j. Constraints (2c) limit server allocations to the available staffing levels, while constraints (2b) ensure that the variables $w_j,\ j \in \mathcal{J}$ record the number of unserved customers. As discussed in the introduction, the abandonment cost function $Q_t(\cdot, \cdot)$ is motivated by the staffing model presented in Harrison and Zeevi (2005) and Bassamboo et al. (2006). The LP determines the minimum weighted sum of unserved customers that can be obtained with the given staffing levels and customer arrival volumes, under several optimistic simplifications. In particular, the queueing effects that arise due to the stochastic dynamic process of customer arrivals and service times are ignored, and it is assumed that servers can be fractionally allocated to customer demands (and likewise, customer demands can be continuously split among different servers). In systems characterized by high volumes, short service times, and impatient customers, these assumptions are justified by the analysis in Bassamboo et al. (2006), which demonstrates that with appropriate real-time routing policies, it is possible to nearly achieve this best-case performance. In other systems, we argue that the abandonment cost function $Q_t(y, \lambda)$ still provides a reasonable "first order" estimate of the impacts of a staffing profile y on the quality-of-service. Indeed, in our numerical experiments, we find a nearly perfect correlation between the abandonment rate as estimated by this LP and by simulation, even for a system with low volume. A similar LP was used in Gurvich et al. (2010) to evaluate feasibility of a staffing profile for a given set of customer counts.

Problem (1) is a two-stage stochastic integer program with integer variables in the first stage, and continuous variables in the second-stage. The second-stage problem (2) is always feasible (it has *complete recourse* in the language of stochastic programming). Because the second-stage problem is a feasible and bounded LP, the second stage value function, $Q_t(\cdot, \lambda)$ is piecewise linear and convex (Birge and Louveaux 1997).

4. Solution methodology

In this section, we describe our proposed methodology. We first describe the sample average approximation approach which yields a large-scale mixed-integer program. We discuss the advantages and

drawbacks of the two well-known methods to solve this model, namely solving the extensive form and Benders decomposition. Next, we examine the gap between the optimal value of the mixed-integer program and its LP relaxation. We prove that under certain assumptions, as the system volume increases, the integrality gap goes to zero. Then, in order to overcome the drawbacks of the two existing solution methods, we introduce a novel method which is an MIR-enhanced version of the standard Benders decomposition algorithm. Finally, we summarize the overall solution approach.

The objective of (1) includes an expected value over the random vector Λ. As in Atlason et al. (2004), Atlason et al. (2008) and Cezik and L'Ecuyer (2008), we use *Sample Average Approximation* (SAA) to overcome the difficulty in evaluating this expected value. Using Monte Carlo sampling, we first obtain a set of independent and identically distributed (i.i.d.) samples from the random vector. These realizations are called *scenarios*. Let $\mathcal{K}$ represent the set of scenario indices and $K = |\mathcal{K}|$. Let $\lambda^1, \ldots, \lambda^K$ be the realizations of Λ. The expected value is then replaced by the sample average over these scenarios, where we assign equal weight $1/K$ to each scenario:

$$\sum_{t \in \mathcal{T}} \mathbb{E}_\Lambda[Q_t(y, \Lambda)] \approx \frac{1}{K} \sum_{t \in \mathcal{T}} \sum_{k \in \mathcal{K}} Q_t(y, \lambda^k).$$

The results of Dai et al. (2000) imply that under mild assumptions, the SAA problem yields an optimal solution to the original problem with probability approaching to one exponentially fast in K. In addition, Kleywegt et al. (2002) present a statistical method for generating optimality bounds by solving multiple SAA problems.

Choosing the sample size requires making a trade-off between the quality of an optimal solution of the SAA problem and the computational burden to solve it. As discussed by Kleywegt et al. (2002), when the sample size is large the objective function of the SAA problem tends to be a more accurate estimate of the true objective function, an optimal solution of the SAA problem tends to be a better solution, and the resulting bounds on the optimality gap tend to be tighter. However, the SAA problem gets more computationally demanding to solve due to increased size. As suggested in Kleywegt et al. (2002), the choice of the sample size may be determined based on the results of preliminary computations.

With the scenarios fixed, we obtain the following deterministic equivalent of the SAA problem, which is referred as the *extensive form* in stochastic programming:

$$\min_{x,y,v,w} \sum_{i \in \mathcal{I}} \sum_{s \in \mathcal{S}_i} c_{is} x_{is} + \alpha \frac{1}{K} \sum_{k \in \mathcal{K}} \sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{J}} p_j w_{jtk} \tag{3a}$$

$$\text{s.t.} \sum_{s \in \mathcal{S}_i} a_{ist} x_{is} \geq y_{it}, \qquad i \in \mathcal{I},\ t \in \mathcal{T}, \tag{3b}$$

$$\sum_{i\in\mathcal{I}} \mu_{ij} v_{ijtk} + w_{jtk} = \lambda^k_{jt}, \qquad j\in\mathcal{J},\ t\in\mathcal{T},\ k\in\mathcal{K}, \tag{3c}$$

$$\sum_{j\in\mathcal{J}} v_{ijtk} \le y_{it}, \qquad i\in\mathcal{I},\ t\in\mathcal{T},\ k\in\mathcal{K}, \tag{3d}$$

$$v\in\mathbb{R}^{IJTK}_+,\ w\in\mathbb{R}^{ITK}_+, \tag{3e}$$

$$x\in X,\ x\in\mathbb{Z}^{IS}_+,\ y\in\mathbb{Z}^{IT}_+. \tag{3f}$$

The first-stage decision variables x and y are the same as defined before, while a copy of second-stage decision variables are created for each period and each scenario. So, v_{ijtk} denotes the number of type i servers allocated to customer class j in period t under scenario k and w_{jtk} represents the number of type j customers that are not served in period t under scenario k. The objective function (3a) minimizes the weighted sum of scheduling cost and average abandonment cost.

There are $IT + KT(J+I)$ constraints, $I(S+T)$ integer variables and $ITK(J+1)$ continuous variables in this model. The size of the model is highly dependent on the number of scenarios. A common planning horizon is one day or one week, and a common scheduling interval is 15 or 30 minutes. Thus, when the number of scenarios is large (e.g., 1000), (3) is a large-scale mixed-integer program (MIP). One option for solving the SAA problem is to directly use a commercial solver to solve (3). However, as we will demonstrate in the numerical results, this approach does not scale well with the number of scenarios, since the formulation (3) simply becomes too large. We therefore explore decomposition strategies.

4.1. Benders decomposition

Benders decomposition is a well-known algorithm that divides a mixed-integer program (MIP) into simpler problems by reformulating the original problem into a problem with fewer variables but an exponential number of constraints (Benders 1962). Since most of the constraints are not active at an optimal solution, they are ignored in the beginning of the solution process and added in a cutting plane fashion as needed. This iterative algorithm is also known as the *L-Shaped method* in the stochastic programming literature. We refer the reader to §5 of Birge and Louveaux (1997) for the details of the algorithm and focus our discussion on the application of this algorithm to our problem.

We decompose our problem into a master problem where the staffing and scheduling decisions are made, corresponding to the first stage model explained in the previous section, and a set of subproblems corresponding to (2) for each customer count scenario. Observe that for fixed values of x and y, problem (3) decomposes into TK independent subproblems. Each subproblem is the same as (2), which is an LP with $IJ + J$ decision variables.

Define $q_t(\cdot)$ as:

$$q_t(y) := \frac{1}{K}\sum_{k\in\mathcal{K}} Q_t(y, \lambda^k). \tag{4}$$

Then, we introduce continuous variables z to represent these function values, and state the Benders master problem as follows:

$$\min_{x,y,z} \sum_{i\in\mathcal{I}}\sum_{s\in\mathcal{S}_i} c_{is}x_{is} + \alpha \sum_{t\in\mathcal{T}} z_t \tag{5a}$$

$$\text{s.t.} \sum_{s\in\mathcal{S}_i} a_{ist}x_{is} \geq y_{it}, \qquad i\in\mathcal{I},\ t\in\mathcal{T}, \tag{5b}$$

$$(z_t, y_{\cdot t}) \in \hat{Z}_t, \qquad t\in\mathcal{T}, \tag{5c}$$

$$x\in X,\ x\in\mathbb{Z}_+^{IS},\ y\in\mathbb{Z}_+^{IT}. \tag{5d}$$

Here, $\hat{Z}_t$ is a polyhedral relaxation of the epigraph of $q_t(\cdot)$:

$$Y_t := \{(z_t, y)\in\mathbb{R}_+\times\mathbb{R}_+^I :\ z_t \geq q_t(y)\}.$$

If we use $\hat{Z}_t = Y_t$, then the resulting problem is an exact reformulation of (3). In the Benders decomposition algorithm, the approximation $\hat{Z}_t$ is progressively improved by adding cuts defining the set Y_t. Thus, the master problem always yields a lower bound on the optimal value of (3), and this lower bound improves as we add cuts to the set $\hat{Z}_t$.

To derive the cuts that will be used to enforce $(z_t, y_{\cdot t})\in Y_t$, consider the dual of (2) corresponding to $\lambda = \lambda^k$, where we let γ and π be the dual variables associated with constraints (2b) and (2c), respectively:

$$Q_t(y,\lambda^k) = \max_{\pi,\gamma} \sum_{j\in\mathcal{J}} \gamma_j\lambda_{jt}^k - \sum_{i\in\mathcal{I}} \pi_i y_{it} \tag{6a}$$

$$\text{s.t. } \mu_{ij}\gamma_j - \pi_i \leq 0, \qquad i\in\mathcal{I},\ j\in\mathcal{J}, \tag{6b}$$

$$\gamma_j \leq p_j, \qquad j\in\mathcal{J}, \tag{6c}$$

$$\pi\in\mathbb{R}_+^I,\ \gamma\in\mathbb{R}^J. \tag{6d}$$

Let Υ denote the set of extreme points of the feasible region of (6), which does not depend on y or λ^k. Then, for each $k\in\mathcal{K}$, we have

$$Q_t(y,\lambda^k) = \max\Big\{\sum_{j\in\mathcal{J}} \bar{\gamma}_j\lambda_{jt}^k - \sum_{i\in\mathcal{I}} \bar{\pi}_i y_{it} : (\bar{\gamma},\bar{\pi})\in\Upsilon\Big\}. \tag{7}$$

Thus, for each $t\in\mathcal{T}$, every cut that is valid for Y_t has the form

$$z_t \geq \frac{1}{K}\sum_{k\in\mathcal{K}}\Big(\sum_{j\in\mathcal{J}} \bar{\gamma}_j^k\lambda_{jt}^k - \sum_{i\in\mathcal{I}} \bar{\pi}_i^k y_{it}\Big) \tag{8}$$

where $(\bar{\gamma}^k,\bar{\pi}^k)\in\Upsilon$ for each $k\in\mathcal{K}$. We refer to cuts of the form (8) as *Benders* inequalities. Given a solution $(\bar{x},\bar{y})$ of the current master problem relaxation, a most violated cut of the form (8) can be obtained by solving (6) for each $k\in\mathcal{K}$ and letting $(\bar{\gamma}^k,\bar{\pi}^k)$ be an extreme point optimal solution.

In the version of Benders decomposition that we have described, we use cuts of the form (8) that directly approximate the expected value functions $q_t(y)$ for each $t \in \mathcal{T}$. This is known as the *single-cut* version of Benders decomposition in the stochastic programming literature. An alternative is the *multi-cut* version, in which cuts would be used to separately approximate the functions $Q_t(y, \lambda^k)$ for each $t \in \mathcal{T}$ and $k \in \mathcal{K}$. While the multi-cut version can lead to convergence in fewer iterations, we found that for this problem the single-cut version is more effective because the multi-cut version introduced a very large number of cuts in the master problem, significantly slowing progress of the algorithm.

4.2. Vanishing integrality gap for high volume systems

Before moving to further improvements in our solution method, we provide conditions under which the relative integrality gap of our model vanishes as arrival volume increases. For this analysis, assume that we fix a sample and focus on the deterministic two-stage stochastic program corresponding to this sample. We define *scaling* the problem by increasing or decreasing the system volume by a given scaling factor. Specifically, we multiply the arrival counts and the right-hand-side values of the constraints on the scheduling variables ($x \in X$) by the given scaling factor keeping everything else the same. Recall that the constraints $x \in X$ may contain constraints such as upper bounds on each server type, or an upper bound on the number of part-time schedules, etc. Then, we show that under certain conditions, as we increase the volume of the system, the linear relaxation bound of our model converges to the optimal value of the mixed-integer program.

We represent our model for a system scaled by integer $\rho > 0$ as

$$\text{IP}(\rho): \quad \nu_{\text{IP}}(\rho) := \min\ c^T x + \alpha\ \frac{1}{K} \sum_{t \in \mathcal{T}} \sum_{k \in \mathcal{K}} Q_t(y, \rho\lambda^k) \tag{9a}$$

$$\text{s.t. } Ax \geq y, \tag{9b}$$

$$Dx \geq \rho d, \tag{9c}$$

$$x \in \mathbb{Z}_+^{IS},\ y \in \mathbb{R}_+^{IT}. \tag{9d}$$

In this formulation we have relaxed the y variables to be continuous, which does not change the optimal value because the matrix A is binary so the integrality restrictions on the x variables imply the left-hand-side of (9b) is integral. Because $Q_t(y, \cdot)$ is a non-increasing function of y, a continuous solution of this model can always be rounded up without increasing the solution cost. The constraints (9c) provide an explicit representation of $x \in X$. We assume that the elements of D and d are integral, but we don't always assume that D and d are nonnegative.

Let $\text{LP}(\rho)$ be the LP relaxation of $\text{IP}(\rho)$ and $\nu_{\text{LP}}(\rho)$ denote its objective value. We start with a simple lemma whose proof is provided in the e-companion to this paper.

LEMMA 1. *Assume LP(1) has an optimal solution. Then LP(ρ) has an optimal solution for any $\rho > 0$ and $\nu_{LP}(\rho) = \rho\nu_{LP}(1)$.*

The following proposition provides our first condition under which the integrality gap decreases as ρ increases. e denotes a vector of ones.

PROPOSITION 1. *Assume D is a nonnegative matrix and LP(1) has an optimal solution. Then, for any $\rho > 0$,*

$$\frac{\nu_{IP}(\rho)}{\nu_{LP}(\rho)} \leq 1 + \frac{\kappa}{\rho}$$

where $\kappa = c^T e/\nu_{LP}(1)$.

Proof. Let $\rho > 0$ be fixed. Suppose that $(\hat{x}, \hat{y})$ is an optimal solution of LP(ρ). Let $\bar{x} := \lceil \hat{x} \rceil$ and $\bar{y} := \hat{y}$. Also, let $\bar{\nu}(\rho) = c^T\bar{x} + (\alpha/K)\sum_{t\in\mathcal{T}}\sum_{k\in\mathcal{K}} Q_t(\bar{y}, \rho\lambda^k)$. Since A and D are nonnegative matrices, $(\bar{x}, \bar{y})$ satisfies (9b) and (9c). Also, $\bar{x}$ is integer. Therefore, $(\bar{x}, \bar{y})$ is a feasible solution to IP(ρ). Hence $\bar{\nu}(\rho)$ is an upper bound on the optimal value of IP(ρ). Then, we have

$$\nu_{\mathrm{IP}}(\rho) \leq \bar{\nu}(\rho) = \nu_{\mathrm{LP}}(\rho) + c^T(\bar{x} - \hat{x}) \leq \nu_{\mathrm{LP}}(\rho) + c^T e.$$

Dividing through by $\nu_{\mathrm{LP}}(\rho)$ and applying Lemma 1 yields the result. □

In the proof of this proposition, we used the fact that rounding a fractional scheduling solution up yields a feasible solution. However, this may fail even in the simple case where we have an upper bound on the number of agents of each type available to be scheduled. In the following proposition we eliminate this assumption and replace it with the assumption that the constraint matrix defining the scheduling problem is totally unimodular. This assumption is inspired by the well-known observation that if there are no breaks in the possible schedules, then A is an interval matrix, so it is totally unimodular.

PROPOSITION 2. *Assume that $[A; D]$ is totally unimodular and LP(1) has an optimal solution. Then, for any integer $\rho > 0$,*

$$\frac{\nu_{IP}(\rho)}{\nu_{LP}(\rho)} \leq 1 + \frac{\kappa'}{\rho}$$

where $\kappa' = \alpha T \sum_{i\in\mathcal{I}} \max_{j\in\mathcal{J}}\{\mu_{ij}p_j\}/\nu_{LP}(1)$.

Proof. Let $\rho > 0$ be a fixed integer. Suppose that $(\hat{x}, \hat{y})$ is an optimal solution of LP(ρ). Let $\bar{y} := \lfloor \hat{y} \rfloor$ and $\bar{x}$ be an optimal solution of the following LP:

$$\min_{x\in\mathbb{R}^{IS}_+} \{c^T x : Ax \geq \bar{y}\ ,\ Dx \geq \rho d\}. \tag{10}$$

Since the coefficient matrix is totally unimodular and the right-hand-side values are all integers, we may assume $\bar{x}$ is integer. Moreover, since $\hat{x}$ was feasible to LP(ρ) and $\hat{y} \geq \bar{y}$, $\hat{x}$ is feasible to (10). Then, we obtain

$$c^T\bar{x} \leq c^T\hat{x}. \tag{11}$$

Now, let $(\bar{\gamma}, \bar{\pi}) \in \arg\max\{\sum_{j\in\mathcal{J}} \gamma_j \rho\lambda^k_{jt} - \sum_{i\in\mathcal{I}} \pi_i \bar{y}_{it} : (\gamma,\pi) \in \Upsilon\}$ so that

$$Q_t(\bar{y}, \rho\lambda^k) = \sum_{j\in\mathcal{J}} \bar{\gamma}_j \rho\lambda^k_{jt} - \sum_{i\in\mathcal{I}} \bar{\pi}_i \bar{y}_{it}. \tag{12}$$

Since $(\bar{\gamma}, \bar{\pi}) \in \Upsilon$, we know that

$$Q_t(\hat{y}, \rho\lambda^k) \geq \sum_{j\in\mathcal{J}} \bar{\gamma}_j \rho\lambda^k_{jt} - \sum_{i\in\mathcal{I}} \bar{\pi}_i \hat{y}_{it}. \tag{13}$$

Next, $\bar{\nu}(\rho) := c^T\bar{x} + (\alpha/K)\sum_{t\in\mathcal{T}}\sum_{k\in\mathcal{K}} Q_t(\bar{y}, \rho\lambda^k)$ is an upper bound on $\nu_{\mathrm{IP}}(\rho)$ since $(\bar{x}, \bar{y})$ is a feasible solution to $\mathrm{IP}(\rho)$ by construction. Combining these observations, we have

$$\begin{aligned}
\nu_{\mathrm{IP}}(\rho) - \nu_{\mathrm{LP}}(\rho) &\leq c^T(\bar{x} - \hat{x}) + \alpha\ \frac{1}{K}\sum_{t\in\mathcal{T}}\sum_{k\in\mathcal{K}} \big(Q_t(\bar{y}, \rho\lambda^k) - Q_t(\hat{y}, \rho\lambda^k)\big) \\
&\leq \alpha\ \frac{1}{K}\sum_{t\in\mathcal{T}}\sum_{k\in\mathcal{K}} \big(Q_t(\bar{y}, \rho\lambda^k) - Q_t(\hat{y}, \rho\lambda^k)\big), && \text{by (11)} \\
&\leq \alpha\ \frac{1}{K}\sum_{t\in\mathcal{T}}\sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{I}} \bar{\pi}_i(\hat{y}_{it} - \bar{y}_{it}) && \text{by (12), (13)} \\
&\leq \alpha\ \frac{1}{K}\sum_{t\in\mathcal{T}}\sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{I}} \bar{\pi}_i \\
&= \alpha T \sum_{i\in\mathcal{I}} \bar{\pi}_i = \alpha T \sum_{i\in\mathcal{I}} \max_{j\in\mathcal{J}}\{\mu_{ij}\bar{\gamma}_j\} && \text{by optimality of } (\bar{\pi}, \bar{\gamma}) \text{ to (6)} \\
&\leq \alpha T \sum_{i\in\mathcal{I}} \max_{j\in\mathcal{J}}\{\mu_{ij} p_j\} && \text{by (6).}
\end{aligned}$$

Applying Lemma 1 yields the result. □

The vanishing optimality gaps results in Propositions 1 and 2 are intuitive, but they do not hold without assumptions such as those given in either of the propositions. As a trivial example, if the constraint matrix $Dx \geq d$ includes constraints $2x_{11} \geq 1$ and $-2x_{11} \geq -1$ (i.e., $2x_{11} = 1$), then the integer program with constraints $Dx \geq \rho d$ is infeasible for any ρ not divisible by two, so that the vanishing integrality result does not hold.

4.3. Mixed-integer rounding (MIR)

The advantage of Benders decomposition is that it decomposes the large problem (3) into a master problem and a set of small single-scenario linear programs. However, as we will see in our numerical results, the Benders decomposition algorithm may fail due to having a weak master problem LP relaxation, leading to a large number of nodes in the branch-and-bound search tree.

Recall that Benders decomposition works by approximating the sets Y_t, $t \in \mathcal{T}$ with cuts, or equivalently, approximating the functions $q_t(y)$ with piecewise-linear convex lower bound functions. A key piece of information that Benders decomposition ignores, which leads to the weak LP relaxations, is that the variables y are *integer* decision variables. This observation motivates us to

develop valid inequalities for the following set in order to obtain better LP relaxation values for the master problem:

$$Z_t := \{(z_t, y) \in \mathbb{R}_+ \times \mathbb{Z}_+^I : \ z_t \geq q_t(y)\}.$$

The key difference between Y_t and Z_t is that Z_t includes the integrality restrictions of the first-stage variables y. Since $Z_t \subseteq Y_t$, the Benders inequalities (8) are valid for Z_t, but inequalities that are valid for the convex hull of Z_t may yield a stronger relaxation than using just the Benders inequalities. We therefore explore the use of *mixed-integer rounding* (MIR) to derive valid inequalities for Z_t for use in Benders decomposition.

The basic mixed integer inequality stated in Wolsey (1998) is as follows:

PROPOSITION 3 **(8.6 of Wolsey 1998)**. *Let $b \in \mathbb{R}$, $f_0 = b - \lfloor b \rfloor > 0$ and $U = \{(u, \eta) \in \mathbb{R}_+ \times \mathbb{Z} : u + \eta \geq b\}$. Then, $u \geq f_0(\lceil b \rceil - \eta)$ is a valid inequality for U.*

Proposition (3) can be used to obtain a more general valid inequality as follows. The following result is a simple generalization of proposition 8.7 of Wolsey (1998). For completeness, a proof is provided in the e-companion.

THEOREM 1. *Let $H^1 := \{(z, y) \in \mathbb{R}_+ \times \mathbb{Z}_+^I : z \geq d_0 - \sum_{i \in \mathcal{I}} d_i y_i\}$, $\beta > 0$ and $f_0 := \beta d_0 - \lfloor \beta d_0 \rfloor$. If $f_0 > 0$, then the inequality*

$$z \geq \frac{f_0 \lceil \beta d_0 \rceil}{\beta} - \sum_{i \in \mathcal{I}} \frac{\min\{f_0 \lceil \beta d_i \rceil, f_i + f_0 \lfloor \beta d_i \rfloor\}}{\beta} y_i \tag{15}$$

is valid for H^1, where $f_i := \beta d_i - \lfloor \beta d_i \rfloor$, $\forall i \in \mathcal{I}$.

Nemhauser and Wolsey (1988) proposed a procedure for applying mixed-integer rounding to a *pair* of inequalities.

PROPOSITION 4 **(6.3 of Nemhauser and Wolsey 1988)**. *Let $U = \{(u, \eta) \in \mathbb{R}_+^p \times \mathbb{Z}_+^n : E\eta + Gu \geq b\}$. Given two valid inequalities*

$$\sum_{j=1}^{n} \delta_j^i \eta_j + \sum_{j=1}^{p} \theta_j^i u_j \leq \delta_0^i \quad \text{for } i = 1, 2 \tag{16}$$

for U, it follows that

$$\sum_{j=1}^{n} \lfloor \delta_j^2 - \delta_j^1 \rfloor \eta_j + \frac{1}{1 - f_0} \Big(\sum_{j=1}^{n} \delta_j^1 \eta_j + \sum_{j=1}^{p} \min\{\theta_j^1, \theta_j^2\} u_j - \delta_0^1 \Big) \leq \lfloor \delta_0^2 - \delta_0^1 \rfloor \tag{17}$$

where $f_0 = (\delta_0^2 - \delta_0^1) - \lfloor \delta_0^2 - \delta_0^1 \rfloor$, is also valid for U.

Thus, we can choose pairs of Benders cuts and apply Proposition 4 to obtain a mixed-integer rounding inequality. However, we find that it is useful to first extend Theorem 1 to derive an inequality for a set defined by two inequalities.

COROLLARY 1. *Let* $H^2 := \{(z,y) \in \mathbb{R} \times \mathbb{Z}^I_+ : z \geq d^1_0 - \sum_{i\in\mathcal{I}} d^1_i y_i,\ z \geq d^2_0 - \sum_{i\in\mathcal{I}} d^2_i y_i\}$, $\beta > 0$ *and* $\bar{f}_0 := \beta(d^2_0 - d^1_0) - \lfloor \beta(d^2_0 - d^1_0)\rfloor$. *If* $\bar{f}_0 > 0$, *then the inequality*

$$z \geq d^1_0 + \frac{\bar{f}_0 \lceil \beta(d^2_0 - d^1_0)\rceil}{\beta} - \sum_{i\in\mathcal{I}} \Big(\frac{\min\{\bar{f}_0 \lceil \beta(d^2_i - d^1_i)\rceil, \bar{f}_i + \bar{f}_0 \lfloor \beta(d^2_i - d^1_i)\rfloor\}}{\beta} + d^1_i \Big) y_i \tag{18}$$

is valid for H^2, *where* $\bar{f}_i := \beta(d^2_i - d^1_i) - \lfloor \beta(d^2_i - d^1_i)\rfloor,\ \forall i \in \mathcal{I}$.

Proof. Let $(z,y) \in H^2$ and define $z' := z - \big(d^1_0 - \sum_{i\in\mathcal{I}} d^1_i y_i\big)$. Then, $z' \geq 0$ and $z' \geq (d^2_0 - d^1_0) - \sum_{i\in\mathcal{I}} (d^2_i - d^1_i) y_i$, i.e. (z', y) belongs to the set

$$\bar{H}^1 := \{(z', y) \in \mathbb{R}_+ \times \mathbb{Z}^I_+ :\ z' \geq (d^2_0 - d^1_0) - \sum_{i\in\mathcal{I}} (d^2_i - d^1_i) y_i\}.$$

Then, by applying Theorem 1, we get a valid inequality for $\bar{H}^1$, to which we substitute z' expression back and obtain (18). □

Note that Proposition 4 provides a procedure similar to that of Corollary 1 for using mixed-integer rounding on a pair of inequalities. However, as we show in the e-companion, the inequality that would be obtained by applying Theorem 4 to the inequalities in the set H^2 is in general weaker than (18). As in Nemhauser and Wolsey (1988), we call the method of applying the procedure in the proof of Corollary 1 as the *MIR procedure*, and the valid inequality (18) an *MIR inequality*.

We next explore how the MIR procedure can be used to obtain valid inequalities for the sets $Z_t, t \in \mathcal{T}$. To apply Corollary 1, we assume that for each $t \in \mathcal{T}$, we have a current approximation of Z_t, obtained using inequalities derived throughout the algorithm, as follows:

$$\hat{Z}_t := \Big\{ (z_t, y_{\cdot t}) \in \mathbb{R}_+ \times \mathbb{Z}^I_+ :\ z_t \geq d^{t,\ell}_0 - \sum_{i\in\mathcal{I}} d^{t,\ell}_i y_{it} : \ell = 1, \ldots, L_t \Big\},$$

where the list of inequalities contains, at least, all Benders inequalities of the form (8) that have been identified up to the current point in the algorithm. We adopt the convention that the inequality $z_t \geq 0$ is also included in this list of inequalities. We can then derive a valid inequality (18) for any pair of inequalities from the list of L_t inequalities and any scaling parameter β. In our implementation, given a current master problem solution $(\bar{z}, \bar{y})$ that we wish to find a valid inequality for, we always require that one of the inequalities used in the pair be the Benders inequality that is most violated by $(\bar{z}, \bar{y})$, thus limiting the search to $O(L_t)$ pairs rather than $O(L^2_t)$ if all pairs are considered. For the scaling parameter β, we try the values 1 and $1/|d^2_i - d^1_i|$ for $i \in \mathcal{I}$.

We also use the MIR procedure to derive valid inequalities separately for each scenario, and then aggregate the results into a single cut to be used in our single-cut implementation of Benders decomposition. Specifically, for each $t \in \mathcal{T}$, $k \in \mathcal{K}$ we maintain a list of inequalities:

$$\eta_{tk} \geq d^{t,\ell}_0 - \sum_{i\in\mathcal{I}} d^{t,\ell}_i y_{it}, \quad \ell = 1, \ldots, L'_{tk} \tag{19}$$

which are valid for the set

$$\mathcal{Z}_{tk} := \{(\eta_{tk}, y) \in \mathbb{R}_+ \times \mathbb{Z}_+^I : \ \eta_{tk} \geq Q_t(y, \lambda^k)\}.$$

These inequalities are obtained when solving the subproblems (6) that are used to obtain the aggregated single-cut Benders inequalities (8). We emphasize that we save these inequalities, but do not directly add them to the master formulation. Again, we apply Corollary 1 to any pair of inequalities from the list (19), which we also assume contains the inequality $\eta_{tk} \geq 0$. When attempting to find a valid inequality that cuts off a given solution $(\bar{z}, \bar{y})$, we first solve (6) to find the most violated Benders inequality at that solution and add it to the list (19). We then consider pairing this inequality with each of the other inequalities in the list, and identify the MIR inequality that has the largest scaled violation (where the violation is scaled by the 2-norm of the cut coefficients). Then, for each scenario we choose either the original Benders inequality or this best MIR inequality, based on which has the larger scaled violation. Finally, having chosen a single inequality for each scenario, we aggregate these inequalities and, if the current solution violates the resulting inequality we add it to the master problem. In either case, we also add this inequality to the list of aggregated inequalities defining the set $\hat{Z}_t$ in (19). Therefore, MIR inequalities obtained for the set $\hat{Z}_t$ may be based on inequalities derived using mixed-integer rounding at the scenario level.

4.4. Additional improvements

We present two strategies to improve the Benders decomposition algorithm. We employ these strategies in all of our numerical tests, including those that do not use the mixed-integer rounding inequalities.

4.4.1. Upper bound. Consider a master problem node LP relaxation solution $(\bar{z}, \bar{y}, \bar{x})$ for which $\bar{y}$ and $\bar{x}$ satisfy the integer restrictions. Then, in order to be sure the cost is correctly recorded, the branch-and-cut method solves the subproblems (2) to calculate $Q_t(\bar{y}, \lambda^k)$ for each $k \in \mathcal{K}$, $t \in \mathcal{T}$, and adds a cut of the form (8) if it is violated. At this point, from (4), we also know the value of $q_t(\bar{y})$. We define $z_t^{UB} := q_t(\bar{y}), \ \forall t \in \mathcal{T}$, $y^{UB} := \bar{y}$, and $x^{UB} := \bar{x}$. Then (z^{UB}, y^{UB}, x^{UB}) is a feasible solution and its objective value,

$$\sum_{i \in \mathcal{I}} \sum_{s \in \mathcal{S}_i} c_{is} x_{is}^{UB} + \alpha \sum_{t \in \mathcal{T}} z_t^{UB},$$

is a valid upper bound to the problem. We therefore update the incumbent solution if this value is better than the best known feasible solution.

We may also apply this heuristic when the master problem solution $(\tilde{z}, \tilde{y}, \tilde{x})$ is not integer-feasible. First, we solve the integer program

$$\begin{aligned} \min_{x} \ & \sum_{i\in\mathcal{I}}\sum_{s\in\mathcal{S}_i} c_{is}x_{is} \\ \text{s.t.} \ & \sum_{s\in\mathcal{S}_i} a_{ist}x_{is} \geq \lfloor \tilde{y}_{it} \rfloor, \qquad i\in\mathcal{I},\ t\in\mathcal{T}, \\ & x\in X,\ x\in\mathbb{Z}_+^{IS}, \end{aligned}$$

and let its solution be x^{UB}. We then set $y_{it}^{UB} = \sum_{s\in\mathcal{S}_i} a_{ist}x_{is}^{UB}$ for $i\in\mathcal{I}$, $t\in\mathcal{T}$, and finally let $z_t^{UB} = q_t(\bar{y}^{UB})$ for $t\in\mathcal{T}$, where the last step requires solving subproblem (2) for all $t\in\mathcal{T}, k\in\mathcal{K}$. Since this routine requires a significant amount of work due to the necessity of solving all the subproblems, we apply this procedure once at the end of the root node, at all nodes up to depth five of the branch-and-bound tree and at every 100 nodes after that. When we execute this heuristic, we also add the valid inequalities of type (8) if they cut off the current solution and save them for future use in the MIR procedure.

4.4.2. Initialization of master problem via Jensen's inequality. Another drawback of the Benders decomposition algorithm is that early on in the algorithm, the master problem has few or no cuts, and so is a poor approximation of the true feasible region. Thus, time can be spent generating cuts at solutions that are ultimately not promising. We remedy this weakness by including in the master problem a set of constraints based on the average value of the sampled values λ_{jt}^k, as proposed in Batun et al. (2011). In particular, this technique is motivated by Jensen's inequality: $E_\Lambda[Q_t(y,\Lambda)] \geq Q_t(y, E_\Lambda[\Lambda])$ for any y because $Q_t(y,\lambda)$ is convex in λ. We therefore introduce constraints that model the valid inequality $z_t \geq Q_t(y, E_\Lambda[\Lambda])$ for each $t\in\mathcal{T}$. This is accomplished by defining continuous variables $\bar{v}_{ijt}$ and $\bar{w}_{jt}$, and adding the following constraints to our initial master problem (5):

$$\begin{aligned} & z_t \geq \sum_{j\in\mathcal{J}} p_j \bar{w}_{jt}, && t\in\mathcal{T}, \\ & \sum_{i\in\mathcal{I}} \mu_{ij}\bar{v}_{ijt} + \bar{w}_{jt} = \bar{\lambda}_{jt}, && j\in\mathcal{J}, t\in\mathcal{T}, \\ & \sum_{j\in\mathcal{J}} \bar{v}_{ijt} \leq y_{it}, && i\in\mathcal{I}, t\in\mathcal{T}, \\ & \bar{v}\in\mathbb{R}_+^{IJT}, \bar{w}\in\mathbb{R}_+^{JT}, \end{aligned}$$

where $\bar{\lambda}_{jt} = K^{-1}\sum_{k\in\mathcal{K}} \lambda_{jt}^k$ is the average of the *sampled* values of Λ_{jt}.

4.5. Details of branch-and-cut algorithm

We next describe how these ideas are integrated into a branch-and-cut algorithm for solving a single instance of the two-stage stochastic programming model (3) for a fixed value of α. In the next section, we discuss how this procedure can be used within an overall procedure to choose a schedule that balances quality-of-service and cost.

In a "textbook" implementation of Benders decomposition for solving a mixed-integer program, the master problem solved at each iteration is a mixed-integer program. However, repeatedly solving this MIP to optimality can be computationally slow. Therefore, following the standard practice in mixed-integer programming, we instead embed the generation of Benders inequalities within a branch-and-bound algorithm, leading to a *branch-and-cut* algorithm. In this algorithm, a branch-and-bound tree search is done over the integer variables in the master problem. For each node that is not pruned by bound or infeasibility, the cutting plane algorithm is executed and cuts are added to the master problem formulation. If the current master problem LP relaxation solution is not integer feasible, then the cutting plane procedure can be terminated early and branching can be done. When the LP relaxation satisfies the integrality constraints, the algorithm always adds cuts if any violated ones can be found. If not, the best found feasible solution (the incumbent) is updated with the current solution, and the node is pruned.

After each LP relaxation is solved in the branch-and-bound tree, cuts are searched for and added as follows. If the LP relaxation solution is integer feasible, then the second-stage subproblems (7) are solved and the Benders inequality (8) is added if it is violated. The addition of these so-called *lazy cuts* is implemented within a LazyConstraintCallback in CPLEX. We also evaluate the objective value of this solution as described in section 4.4.1, and add if it is better than the incumbent solution we update the incumbent. If the LP relaxation solution is not integer feasible, then we optionally search for violated MIR-enhanced Benders inequalities as described in section 4.3. The addition of these so-called *user cuts* is implemented within a UserCutCallback in CPLEX. In our implementation, we always search for violated MIR-enhanced Benders inequalities at the root node, and at most once per node at other nodes in the branch-and-bound tree.

4.6. Suggested overall solution approach

Because our proposed stochastic integer programming model uses an LP approximation to estimate abandonments, we suggest using this model to generate candidate schedules, and then use a simulation model to more accurately evaluate the fraction of abandonments for these schedules. Specifically, the steps of our suggested approach are as follows:

1. Generate candidate schedules. The goal is to generate a variety of schedules on the efficient frontier between cost and estimated quality-of-service. This is done by solving the two-stage stochastic integer program for multiple values of α (SAA problem (3)) and recording the schedules obtained from each of these problems. Because the stochastic integer programming formulation includes discrete decisions, using a weighted average of the two objectives (scheduling cost and expected abandonments) may not be sufficient to yield all possible solutions on the efficient frontier (see e.g., Ehrgott (2005)). Thus, after approximating the efficient frontier by varying the parameter α, if there are large gaps in the efficient frontier, these may be filled in by solving a model in which the expected abandonment costs are minimized, while the scheduling cost is constrained to be less than a given upper bound, or in which the scheduling costs are minimized with a constraint on the expected abandonment costs.

2. Evaluate the schedules via simulation. Use a simulation model to obtain estimates of the fraction of abandoned customers for each of the candidate schedules. Driving the simulation requires choosing the routing rule that will be used when the schedule is implemented. In our experiments, we use the "shadow routing" rule proposed by Stolyar and Tezcan (2010).

3. Choose a solution. A scatter plot showing cost vs. quality-of-service, as estimated via simulation, of each schedule is then plotted. The nondominated solutions in this plot define an estimated efficient frontier of solutions in terms of the two objectives of quality-of-service and cost. The schedule is chosen by the decision-maker according to her preference on the trade-off between quality-of-service and cost.

5. Numerical illustration

We next describe our numerical experiments that illustrate the potential benefit from solving the integrated staffing and scheduling model and the computational efficiency of the proposed approach. We first give details about the data used in our experiments. We then compare the solutions of our integrated model with the solutions obtained by considering staffing and scheduling separately. Finally, we present results on the performance of the proposed MIR-enhanced Benders decomposition approach.

5.1. Test instances

We base our test instances on data derived from a bank call center, documented by Guedj and Mandelbaum (2000). The data is collected over 12 months of 1999 at the level of individual calls. It includes 30-40 thousand calls per month involving callers who desire to speak to an agent. A comprehensive description of the data is presented in Mandelbaum et al. (2000). Statistical analysis of this data from a queueing perspective is given in Brown et al. (2005).

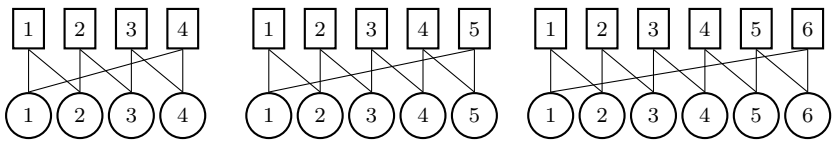
We chose the data from April 1999 to derive our test data, and only considered data from weekdays, during which the call center is staffed from 7 am to midnight. Whenever possible, the data in our tests are based on the available data, although in some cases we needed to make some assumptions on the data, in particular regarding the server costs and skills. Full details about our test instances are provided in the e-companion.

In the empirical analysis done by Brown et al. (2005), the authors find that a time-inhomogeneous Poisson process fits the arrival data. We follow common call center practice and assume arrival rates are constant during individual time periods. We focus our study on planning the schedule for a single day and use time periods of one-half hour. Therefore, our planning horizon is $T = 34$ periods.

There are six types of calls in the data. To forecast the arrival rate functions of the time-inhomogeneous Poisson process, we follow an approach introduced in Ye et al. (2014), which extends some of the ideas in Shen and Huang (2008b,a), Weinberg et al. (2007) to a system with multiple customer arrival streams. Specifically, the number of arrivals of customer type $j \in \mathcal{J}$ in period $t \in \mathcal{T}$ is assumed to be a Poisson random variable with random rate $\Gamma_{jt} = (R_j \delta_{jt})^2$. Here $(R_1, \ldots, R_J)$ is modeled as a joint normal random vector with mean r and covariance matrix Σ, and for each $j \in \mathcal{J}$, $\delta_j \in \mathbb{R}_+^T$ is a profile vector satisfying $\sum_{t \in \mathcal{T}} \delta_{jt} = 1$. The parameters r, Σ, and $\delta_j, j \in \mathcal{J}$ are estimated from the data as described in the electronic companion. The random vector $(R_1, \ldots, R_J)$ captures correlation between customer classes, whereas the profile vectors δ_j capture intra-day correlations. After estimating this forecast distribution of the arrival rates throughout the day, we generate samples of customer counts for the SAA problem by first sampling the arrival rate from the forecast distribution, and then sampling from the Poisson distribution with the given sampled arrival rate.

Agents of different types have different skills where a skill is defined as the subset of the customer classes that an agent can serve. We represent the available server-customer class relationships as a bipartite graph where the partite sets correspond to server types and customer classes and edges correspond to existing skills. Figure 1 shows the topologies we used, where the rectangles represent customer classes and the circles represent server types. The topologies we use all have $I = J$, and

Figure 1 Three topologies used in the numerical experiments.



(a) 4by4 (b) 5by5 (c) 6by6

have the "1-chain" structure (also called a "long chain") in which each customer $j \in \mathcal{J}$ can be served by the two agents j and $(j+1) \mod J$. Jordan and Graves (1995) showed that this type of network performs almost as well as a completely flexible network in certain settings. Similarly, Wallace and Whitt (2005) demonstrated empirically that for properly balanced systems with good routing strategies, one or two skills per agent often yields similar quality-of-service to that obtained by a system where all agents have all skills. For service rates, we first estimate from the data the service rate of each customer class $j \in \mathcal{J}$, independent of server, and set μ_{jj} to be this estimate. For $i = (j+1) \mod J$, the service rate μ_{ij} is set to $0.9\mu_{jj}$. This represents a system where each customer type j has a corresponding "specialist" server type j, and the service rate for a customer being served by its non-specialist server is slightly smaller.

We created a variety of different schedules, including full-time and overtime schedules (eight to nine and a half hour shifts) having one half-an-hour break, and also part-time schedules (three to five hours long) with no break. There are 333 schedules in total. We assume that every schedule is eligible to each server type. The cost of a schedule is calculated as the number of periods in the schedule multiplied by the cost per period for the agent type. Per-period server costs are assumed to be 1.1 for all agent types, up to eight hours. Overtime periods are charged at 1.5 times the normal cost. Customer penalty costs p_j are set to one. Complete details about the schedules and the costs can be found in the e-companion. We assume we have at most 50 of each server type available, so our model includes the constraints: $\sum_{s \in \mathcal{S}_i} x_{is} \leq 50$ for all $i \in \mathcal{I}$. Also, unless otherwise stated, we limit the total number of part-time agents (over all agent types) to four, i.e., $\sum_{i \in \mathcal{I}} \sum_{s \in \mathcal{S}'_i} x_{is} \leq 4$, where $\mathcal{S}'_i$ is the set of part-time schedules for server $i \in \mathcal{I}$.

We next describe the simulation model used to evaluate the fraction of abandonments in a schedule. We assume that each server pool has its own queue, arriving customers are immediately routed to these queues, and the customers in a queue are served in a first-come first-serve basis. We use the shadow routing rule of Stolyar and Tezcan (2010) to route customers to server queues, with the routing rule parameter η set to 0.0001. We reinitialize virtual queues used in the shadow routing rule at the beginning of each time period. Customers arrive according to a time-inhomogeneous compound Poisson process, where the arrival rate in each period is randomly sampled at the beginning of each period, and then fixed during the period. Customers abandon the queue after an exponentially distributed patience time, and service times are assumed to be exponentially distributed. For patience rate of a customer class, we use the average of the service rates of the server types which can serve that customer class. Whenever a period change happens, if some busy agents have to leave due to a decrease in the staffing levels, we choose the ones with minimum work left, and assume they continue to work until that service is complete. (Service times are

small relative to the length of a period.) At the end of the day, all busy servers finish serving their customer, but any customers that are in a queue abandon.

In the e-companion, we present results of an experiment analyzing the lower and upper bounds that can be obtained with the sample average approximation approach with varying sample size. We find that a sample size of 500 is sufficient to yield high quality solutions (upper bounds), and that with a sample size of 2000 it is possible to provide lower bounds that verify these solutions are within 1% of optimality.

5.2. Value of integrated staffing and scheduling

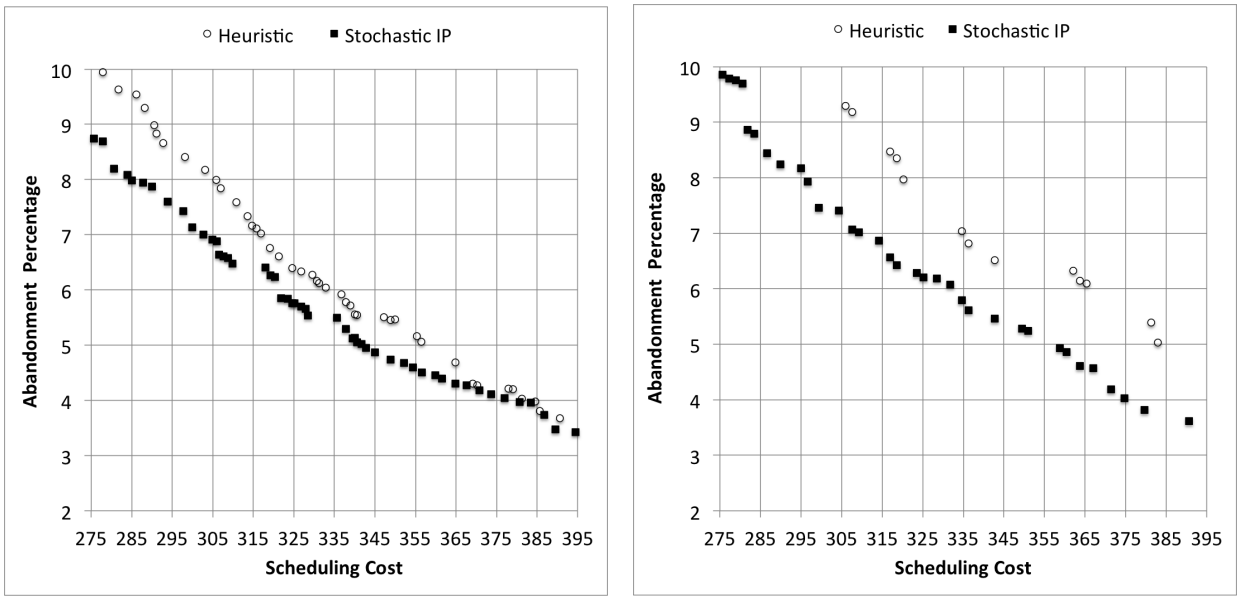
We first investigate the value of the integration of the staffing and scheduling problems. We compare the solutions obtained with the proposed stochastic integer program with the output of the following two-step heuristic. First, staffing levels for each individual time period are determined by solving a staffing problem that minimizes a weighted sum of the cost of the servers selected and the expected abandonment costs within that period. Thus, a two-stage stochastic *linear* program is solved for each time period. Given these staffing levels, we then solve an integer program to obtain the minimum cost schedules that meet these levels. We compare the solutions generated from our proposed integrated model and this two-step heuristic for the 5by5 topology, and for this experiment we fix the number of scenarios used in the optimization models at $K = 500$. (Evaluation of solutions is done via simulation using more replications.) Here, we solve the stochastic integer programs to 0.5% optimality.

Because the ultimate objective of this model is to obtain a set of solutions on the efficient frontier between scheduling cost and quality-of-service (as measured by number of abandonments), we compare these two methods by comparing the efficient frontier that we are able to obtain with them. Thus, for each method, we generated a large number of different solutions by varying the parameter α used to weight the abandonment cost in the combined objective function, and then for each solution we evaluate its scheduling cost and estimate the quality-of-service via simulation. We measure the quality-of-service by the abandonment percentage, calculated as the ratio of total abandonments (over all customer types, time periods, and replications) to total number of customers. We generate 10000 samples of arrival times throughout the day (each sample represents the whole sequence of arrival times for each customer) and use these to perform 10000 replications of simulation by using this sample set and estimate of quality-of-service. We use the same sample set of arrivals to evaluate all the solutions from both methods. Similarly, each solution is evaluated using the same streams of random service and patience times in the simulation replications.

Figure 2 displays scatter plots of the scheduling cost and abandonment percentage for the solutions obtained using the integrated model, labeled "Stochastic IP", and those obtained using the

two-step heuristic, labeled "Heuristic" for two different assumptions on the maximum number of part-time schedules that were allowed. The scheduling costs are on the x-axis and the estimated abandonment percentages are on the y-axis. The upper bound for part-time schedule usage is denoted by "PT". The plot 2(a) shows the results of our base-case which allows up to four part-time schedules (PT= 4), and the plot 2(b) shows the results in which no part-time schedules are allowed (PT= 0). We also experimented with allowing more than four part-time schedules, but the results were very similar to the base case of four. Only the solutions which are not dominated by any of the others are shown in the figure. The half-widths of the confidence intervals on the estimated abandonment percentages are very small compared to the means (1-2% of the means), so these are not displayed.

Figure 2 Efficient frontiers of heuristic and integrated stochastic model with and without part-time schedules.



(a) PT=4 (b) PT=0

For the stochastic integer programming case, the solutions are obtained by first varying the parameter α and solving the model (3). Additional solutions are obtained by solving a model in which the expected abandonment costs are minimized, while the scheduling cost is constrained to be less than a given upper bound in order fill in large gaps in the efficient frontier. For the two-step heuristic method, the parameter α is used to balance staffing cost and expected abandonments within a stochastic *linear* program. Thus, all solutions that define extreme points of the efficient frontier in the objective space can be obtained by varying the weight α in the objective. Unfortunately, when such solutions are then passed to the scheduling integer program step of the heuristic,

this can leave large gaps in the efficient frontier, which is especially the case when part-time schedules are not allowed (PT= 0). Thus, one potential drawback of the two-step heuristic is a difficulty to find solutions in the desired trade-off range between the two objectives.

The efficient frontier obtained using the integrated model is strictly below the efficient frontier of the heuristic, and the gap is substantial over much of the frontier. For instance, for our baseline case in which four part-time schedules are allowed, at a scheduling cost of 305 the integrated model finds a schedule with abandonment percentage 6.9%, as compared to 8.0% from the heuristic. At a scheduling cost of 365, the integrated model finds a schedule with abandonment percentage about 0.4% smaller than with the heuristic. Alternatively, at an abandonment percentage of 8%, the integrated model obtained a schedule that is 6.8% less costly than the heuristic. Holding the abandonment percentage at 5%, the integrated model yields scheduling cost savings of around 4.5%. Another observation is that when the abandonment percentages are low, the difference in scheduling cost between the integrated model and the two-step heuristic is relatively smaller. This suggests that the two-step heuristic may be more adequate for systems in which very high quality-of-service is sought.

When we eliminate the option of part-time schedules (the case PT= 0), we see that the efficient frontiers from both methods shift up, which means that obtaining the same abandonment percentage is more costly, illustrating the value of having the more flexible part-time schedules available. We also observe that the difference between the efficient frontiers obtained from the two methods is greater when part-time schedules are not allowed, suggesting that the integrated model is especially valuable when there is less flexibility in the scheduling decisions. For example, at a scheduling cost of about 307, the integrated model finds a schedule with abandonment percentage of 7.1%, compared to 9.2% for the two-step heuristic. Alternatively, the least expensive schedule obtained from the two-step heuristic that obtains an abandonment percentage of 6% has cost 365, whereas the integrated model is able to obtain a schedule with the same abandonment percentage at a cost of about 332, a savings of 9%.

Next, we compare the abandonment percentages obtained via simulation with the ones obtained via the LP approximation given in (2). We estimate the abandonments obtained by each solution of our stochastic model used for Figure 2 by simulation and by the LP approximation. To eliminate the bias introduced by optimization with respect to a given sample, we estimate the abandonment percentage by using a new sample of size 10000. Specifically, we generate 10000 samples of daily arrivals in order to make 10000 simulation replications and create a sample of size 10000 for customer counts (generated from the same arrival rates) which are used in the LP.

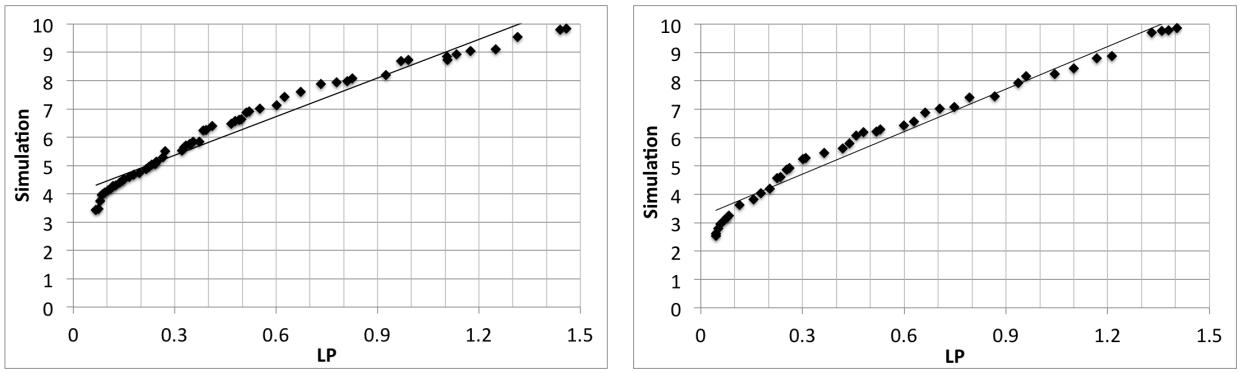
As mentioned before, for simulation the abandonment percentage is calculated as the ratio of total abandonments (over all customer types, time periods, and replications) to total number of

customers. For each solution, the LP approximation of abandonment percentage is obtained by solving the LP (2) for each period and each scenario in the new sample to obtain the abandonment amounts (w_{jtk}) for customer class j in time period t and scenario k. The abandonment percentage estimate is then

$$\frac{\sum_{j\in\mathcal{J}}\sum_{t\in\mathcal{T}}\sum_{k\in\mathcal{K}} w_{jtk}}{\sum_{j\in\mathcal{J}}\sum_{t\in\mathcal{T}}\sum_{k\in\mathcal{K}} \lambda^k_{jt}}.$$

Figure 3 shows scatter plots (with trendlines) of the abandonment percentages obtained via simulation and the LP approximation, for the two different assumptions on the upper bound for part-time schedule usage. Each point in these plots corresponds to a single solution, with its x coordinate given by the abandonment percentage as estimated via the LP approximation, and its y coordinate given by the abandonment percentage as estimated via simulation.

Figure 3 Scatter plots of the abandonment percentages estimated by simulation and by the LP approximation, with part-time schedules (a) and without (b).



(a) PT=4 (b) PT=0

Since the LP approximation allows fractional assignment of customers to servers and ignores the timing of customer arrivals and service completions within a period, it yields optimistic estimates of the abandonment percentages. Indeed, as observed from Figure 3, the LP approximation significantly and consistently underestimates the abandonment percentage. Therefore, we find that it is indeed important to evaluate solutions by using simulation to obtain a more accurate estimate of the quality-of-service. However, we also find there is a very strong positive correlation between the quality-of-service estimates obtained via simulation and the LP approximation. In particular, a linear regression analysis between these factors yields coefficient of determination (R^2) values of 0.95 and 0.97 for the PT=4 and PT=0 cases, respectively, indicating a very good fit of the linear model. Also, the correlation coefficient for both cases is 0.98. These results suggest that the LP approximation is a very good proxy for the actual (simulation estimated) abandonment percentage, in the sense that it can be effectively used to *rank* solutions according to their abandonment

percentage. In other words, if simulation finds one solution to have better quality-of-service than another, then it is usually the case that the LP approximation will have the same preference. This explains why, despite the very biased estimates of abandonment percentage given by the LP approximation, using this approximation within an approach that generates the efficient frontier of feasible solutions can yield very good solutions, as we saw in the example. Thus, by using the LP approximation combined with the efficient frontier approach, we are able to obtain high quality solutions and exploit the computational benefits of using the LP approximation.

Lastly, we compare the heuristic and the integrated model at three different standard deviation levels of the arrival rates. The case with standard deviation $\sigma = 1$ corresponds to the original data where we used the mean vector and variance-covariance matrix estimated from the data in the arrival rate generation procedure. We also increased and decreased the estimated standard deviation by 20% and then generated arrival rates using the modified variance-covariance matrix (keeping the original mean vector), yielding two cases referred to as $\sigma = 1.2$ and $\sigma = 0.8$, respectively.

Figure 4 **Efficient frontiers of heuristic and integrated model for varying arrival rate standard deviations.**

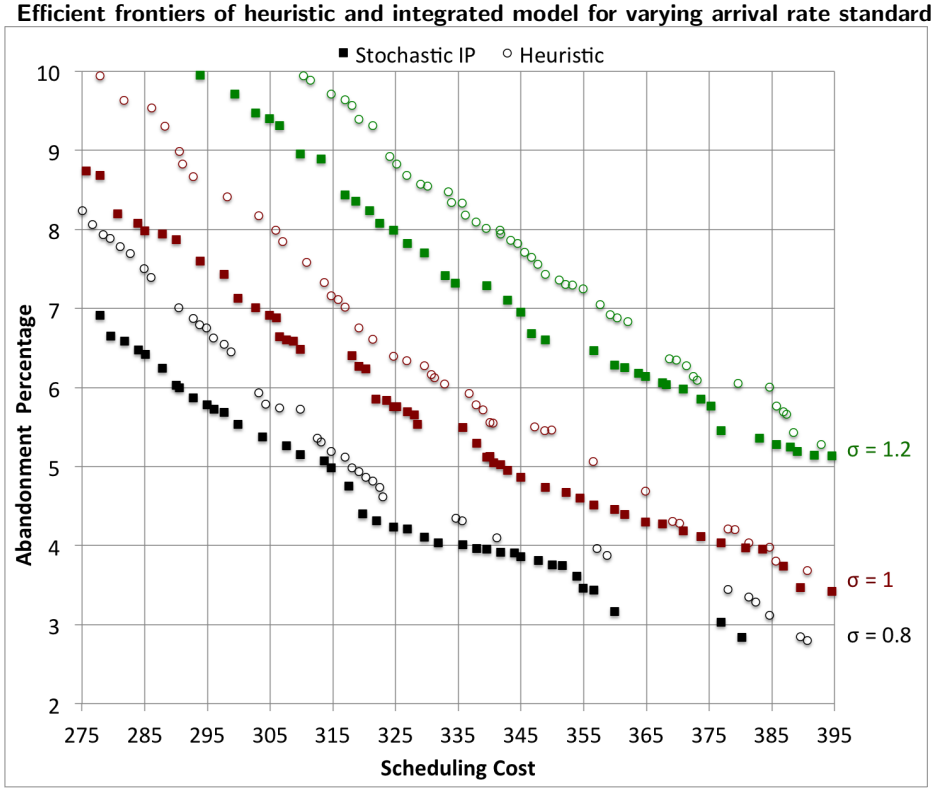


Figure 4 shows the efficient frontiers of the two-step heuristic and the integrated model for these three arrival rate standard deviation cases. The two frontiers closest to the top-right corner of the plot are obtained in the $\sigma = 1.2$ case and the two frontiers closest to the bottom-left corner of the plot are obtained in the $\sigma = 0.8$ case. The first observation is that higher standard deviation in the

arrival rates causes the efficient frontiers for both the heuristic and the integrated model to shift significantly to the right. Thus, obtaining a given abandonment percentage requires significantly higher scheduling costs when the standard deviation is higher, regardless of the method used. This suggests that there is a very large potential benefit to improving arrival rate forecasts. The second observation is that changes in the standard deviation do not affect the relative performances of the heuristic and the integrated model. The efficient frontier of the heuristic is strictly above that of the integrated model with similar magnitude of the difference in all cases. Thus, the integrated model can provide significant benefits over a range of standard deviation levels.

5.3. Comparison of algorithmic performance

We next investigate the computational efficiency of the proposed method for solving the two-stage stochastic integer programming formulation that arises in our approach. Specifically, we compare the use of MIR inequalities to classical alternatives.

Implementation details. We implemented all algorithms in C++ using Cplex 12.4. We performed all experiments on a Linux workstation with 24, 2.67 GHz Intel Xeon CPUs and 132 GB memory. We set the number of threads to one, although we note that the Benders decomposition algorithm (with and without MIR inequalities) can naturally be implemented in parallel computing environments. We solved the deterministic equivalent extensive form of our model with default Cplex settings, while we turned off presolve for the Benders algorithms. For convenience in conducting these tests, we used a solution time limit of one hour. We also repeated these experiments with a three-hour time limit, and found similar results (which we do not report for brevity).

In order to limit the total number of cuts added to the master problem, we generate user cuts only at nodes with depth up to five in the branch-and-bound tree and at every 100 nodes for the rest. We always apply MIR strengthening for these cuts if the MIR option is selected. Also, at every node except the root node, we generate at most one round of user cuts. When we do attempt to generate cuts, we add all of the identified cuts that are violated at the current solution by a minimum relative violation, defined as the absolute violation of the cut divided by the 2-norm of the cut coefficients. User cuts are added if they have relative violation at least 10^{-4} at the root node and at least 5×10^{-4} at other nodes in the tree. Lazy cuts are added if they have relative violation at least 10^{-5}. Unless otherwise stated, we set the relative MIP optimality tolerance to 0.5%.

Computational comparison. We compare the computational performance of solving the integrated model using three methods: solving the extensive form using default Cplex; using pure Benders decomposition within a branch-and-cut algorithm; and using the Benders decomposition method enhanced with MIR inequalities. We denote them as "Ext", "Ben" and "+MIR", respectively.

In Table 1 we present the results for three different topologies and varying number of scenarios. In all of these instances we fixed $\alpha = 1.5$[1]. In the columns labeled as "Time/Gap", we report the time spent to solve the instance in the case that the instance is solved in the one-hour time limit. For the instances that are not solved within the time limit, the "Time/Gap" column reports the percentage optimality gap (calculated as $100 * (z_{UB} - z_{LB})/z_{UB}$ where z_{UB} denotes the upper bound and z_{LB} denotes the lower bound) obtained at the end of the hour. We also report the number of branch-and-bound nodes executed by each algorithm and the LP relaxation gap ("LP Gap (%)") with and without MIR inequalities. The LP gap is calculated as $100 * (z_{IP} - z_{LP})/z_{IP}$, where z_{IP} denotes the optimal value of the two-stage stochastic program and z_{LP} denotes the LP relaxation optimal value obtained after processing the root node with or without MIR inequalities.

Table 1 Comparison of algorithms to solve the two-stage stochastic integer program.

		Time/Gap			# Nodes			LP Gap (%)	
Topology	K	Ext	Ben	+MIR	Ext	Ben	+MIR	Ben	+MIR
4by4	500	1.4%	**2114**	**72**	371	19259	118	4.0	0.6
	1000	11.6%	1.1%	**184**	0	28224	202	4.0	0.6
	2000	12.9%	1.2%	**855**	0	15108	5120	4.5	1.0
5by5	500	11.3%	1.7%	**644**	47	29200	5040	4.8	1.4
	1000	10.1%	1.7%	**777**	0	20803	3970	4.9	1.3
	2000	13.8%	1.6%	**1229**	0	18132	3883	5.3	1.5
6by6	500	16.6%	6.1%	**3345**	0	18712	19170	6.4	2.1
	1000	10.6%	5.7%	**2517**	0	15957	9260	6.1	1.8
	2000	38.0%	5.8%	**2930**	0	13088	6640	6.5	2.0

We observe that the extensive form cannot solve any of the instances, mainly because it can only process a small number of nodes in the time limit due to the formulation size. The basic Benders algorithm can solve only one of the instances within one hour, although it ends with a much smaller optimality gap than the extensive form, and the ending optimality gaps do not vary significantly with number of scenarios in the instances. This appears to be due to the ability of the Benders decomposition algorithm to solve many nodes in the time limit. Finally, we find that strengthening the Benders cuts with mixed-integer rounding leads to a significant reduction in the number of branch-and-bound nodes that need to be processed, and hence enables all of the instances to be solved in the time limit.

We also see that using MIR to strengthen the Benders cuts closes significantly more of the root LP relaxation gap than using Benders cuts alone. We provide more details about the processing

[1] After observing α values used to obtain the points in plot 2(a), we conducted these experiments with values $\alpha = 1$, $\alpha = 1.5$ and $\alpha = 2$. We found similar results for all three values, and so for brevity we report only the case where $\alpha = 1.5$.

of the root node in the e-companion, including solution times, the total number of cuts added, and the number of master iterations while solving the root via the two Benders algorithms. We find that even though generating MIR inequalities takes time and leads to the addition of more total cuts, the total time spent solving the root node with and without the MIR strengthening is comparable because fewer iterations are required when the MIR strengthened cuts are used.

5.4. Increasing system volume

We next investigate empirically the effect of scaling up the volume in these test instances, as inspired by Propositions 1 and 2. Since we have upper bounds on the scheduling variables and the schedules include breaks, our test instance does not satisfy the assumptions of either of these propositions, although the constraint matrix may be considered to be "nearly" totally unimodular because it closely resembles an interval matrix. We therefore investigate whether integrality gap decreases as the volume increases in these instances. For this experiment we fix $\alpha = 1.5$ and $K = 2000$. Table 2 shows the results of this experiment. We *scale* the original system by a scale factor given in the column labeled as "ScaleBy" in Table 2, i.e., we multiply the original arrival rates and upper bounds of the constraints on the scheduling variables by the scaling factor, keeping everything else the same. So, the case with one as the scaling factor corresponds to the original data.

Table 2 Comparison of algorithms to solve integrated stochastic program for larger volume systems ($\alpha = 1.5, K = 2000$).

		Time/Gap (%)			# Nodes			LP Gap (%)	
Topology	ScaleBy	Ext	Ben	+MIR	Ext	Ben	+MIR	Ben	+MIR
4by4	1	12.9%	1.2%	**855**	0	15108	5120	4.5	1.0
	3	1.9%	**738**	**577**	1	1360	1460	1.0	0.4
	5	1.1%	**622**	**248**	4	1740	10	0.3	0.1
5by5	1	13.8%	1.6%	**1229**	0	18132	3883	5.3	1.5
	3	3.7%	**1731**	**446**	0	6900	160	0.9	0.4
	5	2.5%	**536**	**412**	0	270	40	0.4	0.2
6by6	1	38.0%	5.8%	**2930**	0	13088	6640	6.5	2.0
	3	8.1%	0.9%	**2107**	0	14324	6190	1.5	0.8
	5	2.5%	**2071**	**779**	0	8100	230	0.6	0.3

We find that when the scale factor is increased the root LP relaxation gap of the standard Benders decomposition (which equals the root LP relaxation of the extensive form) decreases dramatically, leading to drastic reductions in ending optimality gap for the extensive form and making the standard Benders algorithm much more competitive. For systems with high volumes and a modest number of scenarios, even the extensive form might become a viable option. At the

scale factor of three the standard Benders algorithm is now able to solve two instances, although using MIR to strengthen the Benders cuts still leads to significant improvement in computation time. At scale factor five the LP relaxation of the standard Benders algorithm is now small enough that the benefit from using MIR inequalities is almost eliminated. These results suggest that even when the assumptions of Propositions 1 or 2 are not strictly satisfied, the integrality gap of the proposed stochastic integer program may still be small for systems with high volumes, and for such systems using standard Benders cuts within a branch-and-cut algorithm, or even the extensive form, may be sufficient for solving the integrated staffing and scheduling problem.

6. Concluding remarks

In this paper, we consider multiskill multiclass service systems with uncertain arrival rates. We introduce a two-stage stochastic integer program that integrates the staffing and scheduling steps of the workforce management process into a single optimization model. In the first stage the staffing levels and schedules are decided, while in the second stage, after the customer counts are assumed known, a linear program is solved to approximate the effect of the resulting staffing levels on the quality-of-service.

We propose a technique for using mixed-integer rounding to improve the cuts used in a Benders decomposition procedure by making use of the integrality information of the first-stage decision variables. To the best of our knowledge, this is the first general purpose application of the MIR procedure within a Benders decomposition algorithm for solving a two-stage stochastic integer program. This technique may be useful for any class of stochastic integer programs having integer first-stage decision variables. We also show that under certain conditions the integrality gap of the proposed model approaches zero as the system volume increases.

Using test instances based on data from a bank call center, we find that using the integrated stochastic integer programming model can yield staffing and scheduling decisions with significant savings in cost at the same quality-of-service, or that yield significant improvement in quality-of-service at the same cost, relative to solving the staffing and scheduling problems separately. Finally, we find that while it may be difficult to solve the proposed model using standard Benders decomposition or the deterministic equivalent extensive form, the Benders decomposition algorithm enhanced with our procedure for adding MIR inequalities is able to solve optimally, or within a small optimality gap, reasonable size instances within an hour of computation time.

References

Aksin, O. Z., M. Armony, V. Mehrotra. 2007. The modern call center: A multi-disciplinary perspective on operations management research. *Production Oper. Management* **16**(6) 665–688.

Aksin, O. Z., F. Karaesmen, E.L. Örmeci. 2010. Workforce cross training in call centers from an operations management perspective. *Workforce Cross Training* 211.

Atar, Rami, et al. 2005. Scheduling control for queueing systems with many servers: Asymptotic optimality in heavy traffic. *The Annals of Applied Probability* **15**(4) 2606–2650.

Atlason, J., M. A. Epelman, S. G. Henderson. 2004. Call center staffing with simulation and cutting plane methods. *Ann. Oper. Res.* **127** 333–358.

Atlason, J., M. A. Epelman, S. G. Henderson. 2008. Optimizing call center staffing using simulation and analytic center cutting-plane methods. *Management Sci.* **54**(2) 295–309.

Avramidis, A. N., W. Chan, M. Gendreau, P. L'Ecuyer, O. Pisacane. 2010. Optimizing daily agent scheduling in a multiskill call center. *European J. Oper. Res.* **200**(3) 822–832.

Avramidis, A. N., W. Chan, P. L'Ecuyer. 2009. Staffing multi-skill call centers via search methods and a performance approximation. *IIE Trans.* **41**(6) 483–497.

Bassamboo, A., J. M. Harrison, A. Zeevi. 2006. Design and control of a large call center: Asymptotic analysis of an lp-based method. *Oper. Res.* **54**(3) 419–435.

Bassamboo, A., A. Zeevi. 2009. On a data-driven method for staffing large call centers. *Oper. Res.* **57** 714–726.

Batun, Sakine, Brian T Denton, Todd R Huschka, Andrew J Schaefer. 2011. Operating room pooling and parallel surgery processing under uncertainty. *INFORMS journal on Computing* **23**(2) 220–237.

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

Bhulai, S., G. Koole, A. Pot. 2008. Simple methods for shift scheduling in multiskill call centers. *Manufacturing Service Oper. Management* **10**(3) 411–420.

Birge, J. R., F. Louveaux. 1997. *Introduction to stochastic programming*. Springer, New York.

Borst, S., A. Mandelbaum, M. Reiman. 2004. Dimensioning large call centers. *Oper. Res.* **52** 17–34.

Brown, L., N. Gans, A. Mandelbaum, A. Sakov, H. Shen, S. Zeltyn, L. Zhao. 2005. Statistical analysis of a telephone call center: A queueing-science perspective. *J. Amer. Statist. Assoc.* **100**(469) 36–50.

Cezik, M. T., P. L'Ecuyer. 2008. Staffing multiskill call centers via linear programming and simulation. *Management Sci.* **54** 310–323.

Dai, JG, Tolga Tezcan. 2008. Optimal control of parallel server systems with many servers in heavy traffic. *Queueing Systems* **59**(2) 95–134.

Dai, L, C. H. Chen, J. R. Birge. 2000. Convergence properties of two-stage stochastic programming. *J. Optim. Theory Appl.* **106**(3) 489–509.

Ehrgott, Matthias. 2005. *Multicriteria Optimization*. 2nd ed. Springer, Berlin.

Feldman, Z., A. Mandelbaum. 2010. Using simulation-based stochastic approximation to optimize staffing of systems with skills-based-routing. *Proc. Winter Simulation Conf.*. 3307–3317.

Feldman, Z., A. Mandelbaum, W. A. Massey, W. Whitt. 2008. Staffing of time-varying queues to achieve time-stable performance. *Management Sci.* **54**(2) 324–338.

Gans, N., G. Koole, A. Mandelbaum. 2003. Telephone call centers: Tutorial, review, and research prospects. *Manufacturing Service Oper. Management* **5**(2) 79–141.

Garnett, O., A. Mandelbaum, M. Reiman. 2002. Designing a call center with impatient customers. *Manufacturing Service Oper. Management* **4**(3) 208–227.

Guedj, Ilan, Avi Mandelbaum. 2000. "anonymous bank" call-center data. `http://iew3.technion.ac.il/serveng/callcenterdata/index.html`.

Gurvich, I., J. Luedtke, T. Tezcan. 2010. Staffing call centers with uncertain demand forecasts: A chance-constrained optimization approach. *Management Sci.* **56**(7) 1093–1115.

Gurvich, Itay, Ward Whitt. 2007. Service-level differentiation in many-server service systems: A solution based on fixed-queue-ratio routing. *Operations Research* **29** 567–588.

Harrison, J. M., A. Zeevi. 2005. A method for staffing large call centers based on stochastic fluid models. *Manufacturing Service Oper. Management* **7**(1) 20–36.

Henderson, S. G., A. J. Mason. 1998. Rostering by iterating integer programming and simulation. *Proc. 30th Winter Simulation Conf.*. IEEE Computer Society, 677–684.

Ingolfsson, A., F. Campello, X. Wu, E. Cabral. 2010. Combining integer programming and the randomization method to schedule employees. *European J. Oper. Res.* **202**(1) 153–163.

Jennings, O., A. Mandelbaum, W. Massey, W. Whitt. 1996. Server staffing to meet time-varying demand. *Management Sci.* **42**(10) 1383–1394.

Jordan, William C, Stephen C Graves. 1995. Principles on the benefits of manufacturing process flexibility. *Management Science* **41**(4) 577–594.

Kim, K., S. Mehrotra. 2014. A two-stage stochastic integer programming approach to integrated staffing and scheduling with application to nurse management. `http://www.optimization-online.org/DB_FILE/2014/01/4200.pdf`.

Kleywegt, A. J., A. Shapiro, T. Homem-de Mello. 2002. The sample average approximation method for stochastic discrete optimization. *SIAM J. Optim.* **12**(2) 479–502.

Kolesar, P. J., L. V. Green. 1998. Insights on service system design from a normal approximation to erlang's delay formula. *Production Oper. Management* **7** 282–293.

Koole, G., A. Mandelbaum. 2002. Queueing models of call centers: An introduction. *Ann. Oper. Res.* **113**(1-4) 41–59.

Koole, G., A. Pot. 2006. An overview of routing and staffing algorithms in multi-skill customer contact centers. Working paper, Vrije Universiteit Amsterdam, Netherlands.

L'Ecuyer, P. 2006. Modeling and optimization problems in contact centers. *Proc. Third International Conf. on Quantitative Evaluation of Systems*. IEEE Computer Society, 145–156.

Mandelbaum, A., A. Sakov, S. Zeltyn. 2000. Empirical analysis of a call center. Tech. rep., Technion, Israel Institute of Technology.

Mandelbaum, A., S. Zeltyn. 2009. Staffing many-server queues with impatient customers: constraint satisfaction in call centers. *Oper. Res.* **57**(5) 1189–1205.

Mandelbaum, Avishai, Alexander L Stolyar. 2004. Scheduling flexible servers with convex delay costs: Heavy-traffic optimality of the generalized cμ-rule. *Operations Research* **52**(6) 836–855.

Nemhauser, G. L., L. A. Wolsey. 1988. *Integer and combinatorial optimization*, vol. 18. Wiley New York.

Pot, A., S. Bhulai, G. Koole. 2008. A simple staffing method for multiskill call centers. *Manufacturing Service Oper. Management* **10**(3) 421–428.

Robbins, T. R., T. P. Harrison. 2008. A simulation based scheduling model for call centers with uncertain arrival rates. *Proc. 40th Conf. on Winter Simulation*. 2884–2890.

Robbins, T. R., T. P. Harrison. 2010. A stochastic programming model for scheduling call centers with global service level agreements. *European J. Oper. Res.* **207**(3) 1608–1619.

Shen, H., J. Z. Huang. 2008a. Interday forecasting and intraday updating of call center arrivals. *Manufacturing Service Oper. Management* **10**(3) 391–410.

Shen, Haipeng, Jianhua Z Huang. 2008b. Forecasting time series of inhomogeneous poisson processes with application to call center workforce management. *The Annals of Applied Statistics* 601–623.

Stolyar, Alexander L, Tolga Tezcan. 2010. Control of systems with flexible multi-server pools: a shadow routing approach. *Queueing Systems* **66**(1) 1–51.

Thompson, G. M. 1997. Labor staffing and scheduling models for controlling service levels. *Naval Res. Logist.* **44**(8) 719–740.

Wallace, R. B., W. Whitt. 2005. A staffing algorithm for call centers with skill-based routing. *Manufacturing Service Oper. Management* **7**(4) 276–294.

Weinberg, J., L. D. Brown, J. R. Stroud. 2007. Bayesian forecasting of an inhomogeneous poisson process with applications to call center data. *J. Amer. Statist. Assoc.* **102**(480) 1185–1198.

Whitt, W. 2006. Staffing a call center with uncertain arrival rate and absenteeism. *Production Oper. Management* **15** 88–102.

Wolsey, Laurence A. 1998. *Integer programming*, vol. 42. Wiley New York.

Ye, H., J. Luedtke, H. Shen. 2014. Forecasting and staffing call centers with multiple interdependent uncertain arrival streams. Submitted for publication.