Omega 72 (2017) 25–37

Contents lists available at ScienceDirect

Omega

journal homepage: www.elsevier.com/locate/omega

Staffing and scheduling flexible call centers by two-stage robust optimization ☆

Sara Mattia [a], Fabrizio Rossi [b,*], Mara Servilio [a], Stefano Smriglio [b]

[a] Istituto di Analisi dei Sistemi ed Informatica, Consiglio Nazionale delle Ricerche. Via dei Taurini 19, 00185 Roma, Italy
[b] Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica, Università di L'Aquila, Via Vetoio I-67010 Coppito, AQ, Italy

ARTICLE INFO

Article history:
Received 13 October 2015
Accepted 8 November 2016
Available online 14 November 2016

Keywords:
Call center optimization
Shift scheduling
Two-stage robust optimization
Benders decomposition

ABSTRACT

We study the shift scheduling problem in a multi-shift, flexible call center. Differently from previous approaches, the staffing levels ensuring the desired quality of service are considered uncertain, leading to a two-stage robust integer program with right-hand-side uncertainty. We show that, in our setting, modeling the correlation of the demands in consecutive time slots is easier than in other staffing approaches. The complexity issues of a Benders type reformulation are investigated and a branch-and-cut algorithm is devised. The approach can efficiently solve real-world problems from an Italian call center and effectively support managers decisions. In fact, we show that robust shifts have very similar costs to those evaluated by the traditional (deterministic) method while ensuring a higher level of protection against uncertainty.

1. Introduction

Workforce management (WFM) is a complex process and represents a prominent issue in call centers optimization. Its general goal is to find a satisfactory trade-off between the Level of Service (LoS) provided to customers and personnel costs. LoS is mostly based on waiting times and quality of the response, while personnel costs constitute one major expense for call center companies. An insightful description of both these aspects can be found in [22]. WFM is traditionally split into a sequence of almost separate steps [2,11]: forecasting call volumes (*forecasting*); determining the *staffing levels*, defined as the number of agents required at each time period to guarantee the desired LoS (*staffing*); translating them into agents work shifts (*shift scheduling*); assigning agents to such shifts (*rostering*) and, finally, monitoring out-of-adherence situations at operational level and reacting accordingly. In practice, the arrival rate of calls is quite difficult to estimate, resulting in frequent *overstaffing* and *understaffing* situations [33,36]. It is well-known among call center professionals that overstaffing increases costs and also affects agents satisfaction, while understaffing typically lowers the LoS and impacts on revenues [11,13,14]. To overcome these challenges, call centers throughout the industry are exploring different flexibility models, ranging from recruiting temporary operators to delaying less urgent calls [15]. We focus on a specific flexible setting, investigated in [24], that we also encountered in several applied projects. In this setting managers react to front-end understaffing by moving agents from back to front office, eventually accepting some delays in the back office work (e.g., paperworks, e-mails and calls with callback). From now on, a call center implementing such a practice is referred to as *flexible* call center.

WFM poses theoretical and algorithmic challenges in operations management as well. In fact, several topics in this context are object of huge research and we refer the reader to the insightful surveys [2,15] for a comprehensive view. According to the practical decomposition, analytic queueing models or simulation models have been developed for computing staffing levels able to guarantee the desired LoS, while integer programming algorithms have been used to determine optimal shift schedules able to cover such levels. However, most of the assumptions of standard queueing models are often not valid in practice, especially in multi-queue/multi-skill environments. For instance, the *Stationary Independent Period by Period* (SIPP) model assumes the independence of the arrival processes at consecutive periods, whereas significant correlations across time periods are well-known to occur in practice. These affect the quality of the computed staffing levels, as documented in [4,17]. A new stream of research has been recently started with the purpose of getting rid of the shortcomings of the SIPP model and looking at more complex and realistic arrival processes. This naturally leads to integrate staffing and shift scheduling decisions into suitable mathematical programming formulations. For example, in [3,4] formulations are presented to

☆This manuscript was processed by Associate Editor Singh.

* Corresponding author.
E-mail addresses: sara.mattia@iasi.cnr.it (S. Mattia), fabrizio.rossi@univaq.it (F. Rossi), mara.servilio@iasi.cnr.it (M. Servilio), stefano.smriglio@univaq.it (S. Smriglio).

http://dx.doi.org/10.1016/j.omega.2016.11.001
0305-0483/

minimize staffing costs while preserving the LoS and sophisticated simulation-based cutting plane methods are developed. Here, simulation is required to compute the LoS, which is not analytically accessible. In [10,23,35] similar methodologies are applied to a staffing-scheduling problem with a single global service level constraint. Unlike previous papers, in [35] a stochastic programming formulation is introduced. The migration towards this kind of models is further developed in recent papers. In [25] stochastic programming is combined with distributionally robust optimization for multi-period, multi-shift staffing problems under service level constraints, with uncertain calls arrival rates and an intra-day seasonality. In [24] a stochastic programming approach is compared to a robust optimization based method applied to the solution of a staffing problem in a multi-period, single-shift, flexible call center. Finally, in [9] the authors present a stochastic programming model in which the LoS is approximated by a linear program instead of using simulation. This improves computational tractability at the price of reducing the accuracy. A review of the state-of-the-art literature on staffing and scheduling approaches under uncertainty can be found in [11].

The present study starts from a critical review of the WFM process and from the fact that its first step, aimed at computing staffing levels yielding the desired LoS, is challenging from both theoretical and practical perspectives. In fact, errors in the call volume forecasts and point estimate approximations often result in staffing levels which drive the call center to a poor performance when the actual demand shows up. Our proposal is then to accept that some level of uncertainty affects the staffing levels and to cope with it in the staff scheduling step. This leads to investigating a *two-stage* robust optimization model for shift scheduling in a multi-period, multi-shift, flexible call center. A preliminary version of this model has been presented in the conference paper [27], where its adherence with managers' practice in exploiting flexibility is discussed. The key issue to be addressed with this model is the uncertainty description which, however, does not involve the arrival process or the LoS computation. Two main features characterize real-life call centers: (*i*) deviations from the nominal value typically occur only in a limited number of time periods; (*ii*) deviations at consecutive time periods are often not independent [4]. We investigate three uncertainty models based on assumption (*i*), resp. (*ii*) or both. It turns out that the first two options yield an excessive level of conservatism, whereas the third one provides a much better description. This confirms that, even in our setting, the SIPP model, corresponding to option (*i*), is not adequate to address real systems. On the other hand, the third model calls for a more sophisticated methodology, which is developed in this paper. In fact, our formulation belongs to the class of two-stage problems with right-hand-side uncertainty (RHSU), typically tackled by Benders-like reformulation. The continuous version of such problems has been investigated in [29], where the separation problem associated with the Benders constraints is proved to be strongly NP-hard in general. Interestingly, we prove that, even when the uncertainty set includes the correlation between consecutive time periods, the separation problem can be solved in pseudo-polynomial time. A branch-and-cut algorithm is then devised to manage the integer case and experimented on real-world data from a large Italian call center. The favorable theoretical complexity has also borne out in practice, as the convergence of the algorithm is remarkably faster than what is typically observed from formulations of this kind. This allows to show that high level of protections is achievable with shift plans whose cost is comparable to the one computed from deterministic staffing levels. In contrast, exploiting flexibility to re-adjust deterministic plans can be remarkably more expensive.

The paper is organized as follows. In Section 2 we discuss the flexible version of the classic shift scheduling model. In Section 3 we illustrate the two-stage robust optimization approach, describe the uncertainty set and derive the Benders-like reformulation. The separation problem is addressed in Section 4. The computational experience is described and discussed in Sections 5 and 6. Finally, in Section 7, some conclusions are drawn.

2. Flexible shift scheduling

The deterministic shift scheduling model which represents our starting point is here described along with some of its basic properties. We consider a discrete planning horizon $T = \{1, \dots, m\}$ and denote by b_t the staffing level at period t, $t \in T$, i.e., the (integer) number of agents required on duty in period t. Agents are assigned to work shifts, each characterized by a starting time and a duration. Work shifts do not include breaks which are assumed to be managed at the real-time level. Let $J = \{1, \dots, n\}$ be the set of all possible shifts and c_j the cost associated to shift j. The classical shift scheduling problem consists in determining the number of agents to be assigned to each shift in order to satisfy the levels at minimum cost. A basic integer programming model for this problem was introduced in [32]. Let us define the shift matrix $\mathbf{A} \in \{0,1\}^{m \times n}$, with $a_{tj} = 1$ if shift j covers period t and 0 otherwise, and denote by x_j the number of agents assigned to shift j, $j \in J$. The problem writes

$$\begin{aligned} \min \quad & \sum_{j \in J} c_j x_j \\ & \sum_{j \in J} a_{tj} x_j \ge b_t \quad t \in T \\ & \mathbf{x} \in \mathbb{Z}^n_+ \end{aligned} \tag{1}$$

Since shifts do not include breaks, each column has consecutive ones. This property of the matrix, denoted by C1P, allows the problem to be reformulated as a minimum cost flow problem [32].

We modify model (1) so as to include flexibility, i.e., the ability of moving personnel between back and front office when (front-end) overstaffing or understaffing occur. We then introduce surplus resp. slack variables $o_t, u_t \in \mathbb{Z}_+$ to represent the number of agents in excess (overstaffing) resp. defect (understaffing) at period t. Let us also denote by $w^o_t, w^u_t > 0$ the corresponding costs. The *flexible model* reads as

$$\begin{aligned} \min \quad & \sum_{j \in J} c_j x_j + \sum_{t \in T} (w^o_t o_t + w^u_t u_t) \\ & \sum_{j \in J} a_{tj} x_j + u_t - o_t = b_t \quad t \in T \\ & \mathbf{x} \in \mathbb{Z}^n_+, \mathbf{o}, \mathbf{u} \in \mathbb{Z}^m_+ \end{aligned} \tag{2}$$

and still preserves the network structure. In fact, the constraint matrix is in the form $[A|I|-I] = [B|-I]$ with B fulfilling C1P. Also in this case the problem can be reformulated as min-cost flow in an auxiliary graph (see [1], Section 9.6).

Definition 1. Let f_j (l_j) be the first (last) slot such that $a_{tj} = 1$, for every $j \in J$. The auxiliary graph $H(N, F)$ is defined as:

$N = T \cup \{m+1\}$, with supply/demands $b'_t = b_{t+1} - b_t$, $t \in T$ and $b'_{m+1} = -b_m$.

$F = F_o \cup F_u \cup F_x$, where

$F_u = \{(t, t+1), t \in T\}$, *with costs* $c_{t,t+1} = w^u_t$;
$F_o = \{(t+1, t), t \in T\}$, *with costs* $c_{t+1,t} = w^o_t$;
$F_x = \{(f_j, l_j+1), j \in J\}$, *with costs* c_j.

Graph H is drawn in Fig. 1. Since $\mathbf{b}$ is integer, $\mathbf{b}'$ is integer and the optimal flow will be integral. The flow on arcs in F_u (F_o) corresponds to the level of understaffing (overstaffing); the flow on

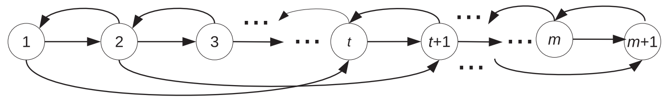

Fig. 1. Graph G of Theorem 2.

arcs in F_x represents the number of agents assigned to the shifts. As a consequence, problem (2) can be solved in strongly polynomial time.

Theorem 2. *Model (2) can be solved in time $O((m+n)^2 \log m + m(n+m)\log^2 m)$.*

Proof. The auxiliary graph has $O(m)$ nodes and $O(n+m)$ arcs. As stated in [30], the time complexity of solving a min-cost flow problem in graph $H(N,F)$ is $O(|F|^2 \log |N| + |F||N| \log^2 |N|)$. Hence problem (2) can be solved in time $O((m+n)^2 \log m + m(n+m)\log^2 m)$. □

It is straightforward to observe that, at any time slot, overstaffing and understaffing cannot show up simultaneously.

Property 3. *For every optimal solution to model (2), $o_t u_t = 0$, for all $t \in T$.*

From now on model 2 will be referred to as *nominal problem*. We now start investigating the case when uncertain staffing levels $\mathbf{b}$ come into play.

3. The robust optimization approach

Robust optimization methods can be classified into single-stage and multi-stage approaches. In the former, the solution is computed entirely before the realization of the uncertainty and the same solution is applied to any realization. In the latter, the solution is computed in stages (usually two): part of the solution is computed before the realization of the uncertainty and part after. Both methods have advantages and disadvantages: single stage solutions are in general easier to compute but often too conservative, whereas multi-stage solutions are less conservative (the solution can be adapted to the actual values), but the corresponding problem is harder to solve. RHSU single stage optimization reduces to solve a nominal problem with suitable right-hand-side values [28]. In our case, a single stage reformulation cannot even be applied, as the nominal problem is defined by equality constraints, which cause the robust single stage problem to be infeasible. Two-stage problems are in general NP-hard and they may require an exponential number of constraints and/or variables (non-compact formulations). This difficulty may be addressed by restricting the second stage variables to be affine functions of the uncertain data [5]. The resulting models, known as affinely adjustable robust (AAR) models, lead to tractable compact reformulations under some assumptions, although they do not provide the full flexibility of a general two-stage model. Unfortunately, we will show that these are not valid in our case. Therefore, we resort to what is known in the literature as two-stage with unrestricted recourse.

Interestingly, a two-stage formulation corresponds to a widespread call center practice: in a first stage, typically at the beginning of the week, a daily shift schedule is computed; in a second stage (at the operational level), personnel reallocation between front and back office is implemented, according to the actual period by period needs. In this framework, $\mathbf{x}$ include the first stage variables, whose values are computed in advance and kept fixed for any realization of the uncertainty, whereas $\mathbf{o}$ and $\mathbf{u}$ are second stage variables, whose values depend on the actual $\mathbf{b}$, along with $\mathbf{x}$.

The two-stage robust version of model (2) is then

$$\min_{\mathbf{x} \in X} \left\{ \sum_{j \in J} c_j x_j + W_{\mathbf{x}}(U) \right\} \tag{3}$$

where X defines the feasible shift schedules, including the integrality of the $\mathbf{x}$ variables; $W_{\mathbf{x}}(U)$ is the minimum reallocation cost for a fixed schedule $\mathbf{x}$ (see Section 4) over the set of all possible realizations of the staffing levels, i.e., the uncertainty set U (see Section 3.2).

The solution to model (3) provides a double information: the value of first stage variables $\mathbf{x}$ and an upper bound on the amount of personnel reallocation cost $W_{\mathbf{x}}(U)$. However, it neither provides the real personnel reallocation cost $R_{\mathbf{xb}}$ nor the value of $\mathbf{o}$ and $\mathbf{u}$ variables, which must be recomputed ex-post, as they depend on both first stage variables $\mathbf{x}$ and the actual realization of $\mathbf{b}$. We call *personnel reallocation* problem the problem of computing $\mathbf{o}$, $\mathbf{u}$ and the corresponding $R_{\mathbf{xb}}$.

3.1. The second stage decision: personnel reallocation

The personnel reallocation problem corresponds to model (2) with fixed $\mathbf{x}$, that is, the problem:

$$\begin{aligned} R_{\mathbf{xb}} &= \min \sum_{t \in T} (w_t^o o_t + w_t^u u_t) \\ u_t - o_t &= b_t - \sum_{j \in J} a_{tj} x_j \quad t \in T \\ \mathbf{o}, \mathbf{u} &\in \mathbb{Z}_+^m \end{aligned} \tag{4}$$

Model (4) is still a min-cost flow problem and then the integrality requirements on the variables can be relaxed. Since (4) is always feasible and bounded (for bounded $\mathbf{b}$), using linear duality, the problem can be equivalently formulated as

$$\begin{aligned} R_{\mathbf{xb}} &= \max \sum_{t \in T} \Big(b_t - \sum_{j \in J} a_{tj} x_j\Big) y_t \\ &-w_t^o \le y_t \le w_t^u \quad t \in T \end{aligned} \tag{5}$$

$R_{\mathbf{xb}}$ can be decomposed by time period and solved in linear time $O(m)$ and, by Property 3, the optimal primal solution is

$$o_t^* = \max \left\{ 0, \sum_{j \in J} a_{tj} x_j - b_t \right\}$$

$$u_t^* = \max \left\{ 0, b_t - \sum_{j \in J} a_{tj} x_j \right\}$$

and the optimal dual solution is

$$y_t^* = \begin{cases} w_t^u & \text{if } b_t - \sum_{j \in J} a_{tj} x_j \ge 0 \\ -w_t^o & \text{otherwise} \end{cases}$$

Since $\mathbf{b}$ is given, the personnel reallocation problem is easy, independent of the uncertainty set. However, the uncertainty set plays a fundamental role in solving the robust problem (3), therefore in the next section we present the description of the uncertainty sets relevant to our application.

3.2. The uncertainty set

The uncertainty set U consists of all the realizations of the uncertain parameters, and traditionally it is expressed by a polyhedron or a convex set [6,8]. In our case, the nominal staffing levels $\tilde{b}_t$ are supposed to be affected by deviations δ_t ranging in intervals $[-D_t, D_t]$. Advanced staffing procedures [19] provide both $\tilde{b}_t$ and D_t. Two further observations contribute to the definition of the uncertainty set, summarized in the following property.

Property 4. *Properties of the uncertainty set*:

(i) *cardinality*: *deviations typically occur only in a limited number Γ of time periods, while in the others they can be considered negligible*;
(ii) *correlation*: *deviations at consecutive time periods are often not independent* [4].

Condition (i) naturally requires the well-known cardinality constrained approach [8]. Formally, let z_t be the percentage deviation in period t and let ζ_t indicate whether a deviation occurs in period t or not. The corresponding set is:

$$U_\Gamma = \left\{ \mathbf{b} \in \mathbb{Z}^m : b_t = \tilde{b}_t + D_t z_t; \sum_{t \in T} \zeta_t \le \Gamma, |z_t| \right.$$
$$\left. \le \zeta_t, \zeta_t \in \{0,1\}; z_t \in \mathbb{R}, t \in T \right\}$$

Notice that, since U_Γ does not include correlation, it corresponds to the SIPP model. In order to model condition (ii), we assume that the difference between the deviations of two consecutive periods is limited by a parameter $\Delta(t)$, obtaining:

$$U_\Delta = \{ \mathbf{b} \in \mathbb{Z}^m : b_t = \tilde{b}_t + D_t z_t; |D_t z_t - D_{t-1} z_{t-1}| \le \Delta(t), t \in T \backslash \{1\};$$
$$z_t \in [-1, 1], t \in T \}$$

When both conditions are enforced, we obtain $U_{\Gamma\Delta} = U_\Gamma \cap U_\Delta$, which is the set that better represents the uncertainty in real-life call-centers. Notice that b_t is integer, as it represents the number of agents on duty at period t. The integrality of b_t, $t \in T$, has some consequences. First, $\tilde{b}_t$ is integer in order to guarantee the feasibility of the nominal scenario, that is, the one with $\mathbf{b} = \tilde{\mathbf{b}}$. If so, $D_t z_t$ has to be integer as well. Therefore, w.l.o.g., we assume D_t and $\Delta(t)$ to be integer. Clearly, all sets are bounded and non-empty and U_Γ, U_Δ are relaxations of $U_{\Gamma\Delta}$. We will demonstrate that taking correlation into account is crucial to obtain an appropriate level of conservatism (see Section 6.1).

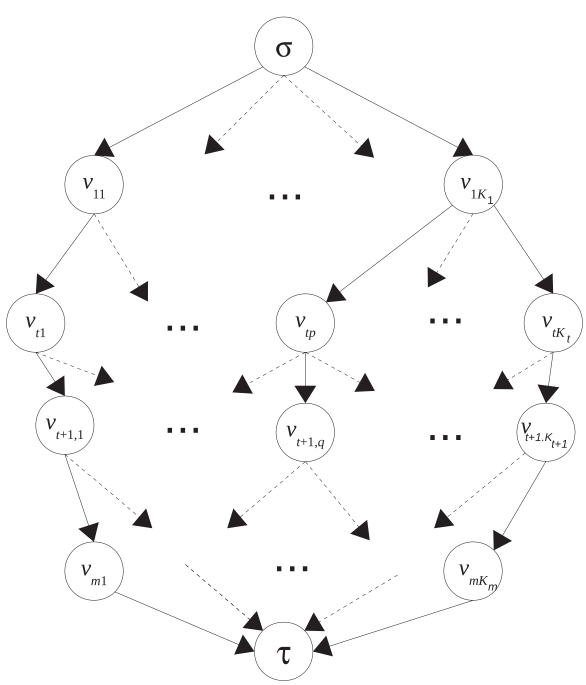

Fig. 2. The Δ-uncertainty graph.

3.3. Benders reformulation

We now illustrate a Benders like reformulation [7] of problem (3) and discuss the related algorithmic issues. Problem (3) can be rewritten as:

$$\begin{aligned} \min \quad & \sum_{j \in J} c_j x_j + \lambda \\ & \lambda \ge W_{\mathbf{x}}(U) \\ & \mathbf{x} \in X \end{aligned} \tag{6}$$

where $\sum_{j \in J} c_j x_j$ represents the total cost of work shifts and $W_{\mathbf{x}}(U)$ the worst-case personnel reallocation cost:

$$W_{\mathbf{x}}(U) = \max_{\mathbf{b} \in U} \{ R_{\mathbf{xb}} \} \tag{7}$$

Using the expression (5) for $R_{\mathbf{xb}}$, $W_{\mathbf{x}}(U)$ is computed as:

$$\begin{aligned} W_{\mathbf{x}}(U) = \max & \sum_{t \in T} \left(b_t - \sum_{j \in J} a_{tj} x_j \right) y_t \\ & -w_o^t \le y_t \le w_u^t \quad t \in T \\ & \mathbf{b} \in U \end{aligned} \tag{8}$$

Therefore, problem (6) becomes:

$$\min \quad \sum_{j \in J} c_j x_j + \lambda \tag{9}$$

$$\lambda \ge \sum_{t \in T} \left(b_t - \sum_{j \in J} a_{tj} x_j \right) y_t \quad \mathbf{b} \in U, \mathbf{y} \in Y \tag{10}$$

$$\mathbf{x} \in X \tag{11}$$

where $\mathbf{y}$ are the dual variables defining $R_{\mathbf{xb}}$ and $Y = \{ \mathbf{y} : -w_o^t \le y_t \le w_u^t, t \in T \}$. Formulation (9)-(11) is non-compact, as it may have an exponential number of constraints.

In principle, one could use the analytic expression of the optimal dual solution $\mathbf{y}$ given in Section 3.1 to derive an equivalent version of constraints (10) as sum of maxima, which are usually solved either by vertex enumeration [31] or by approximating the problem using the AAR framework [16]. As we discuss in Section 4, AAR models cannot be applied when correlation is considered. On the other hand, dropping correlation leads to an excessive level of conservatism that results in too expensive plans (see Section 6.1). Therefore, we devise an algorithm able to efficiently solve large real-life instances with correlation using the Benders-like formulation (9)–(11) presented above.

A standard algorithmic framework for non-compact formulations is the Kelley cutting plane method, originally introduced in [21]. This method starts with a relaxed formulation including a suitable subset of constraints (10). Then, additional constraints are dynamically

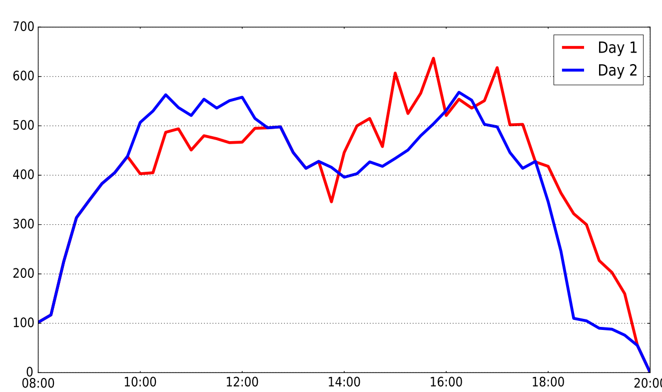

Fig. 3. Staffing levels of *Day1* and *Day2*.

Table 1
Branch-and-cut statistics for U_Γ, Day 1.

	Γ	Obj. value	Root primal	Root LP rel.	Root % gap	B&B nodes	Total time	Sep. time	# cuts	Heur. time
$dev\% = 5$	5	80,021	80,068	80,013.3	0.07	73	6	5	111	1
	10	80,742	80,742	80,732.3	0.01	0	5	5	126	<1
	15	81,393	81,575	81,389.0	0.23	49	6	6	147	<1
	20	82,017	82,122	82,014.0	0.13	25	7	7	158	<1
	25	82,628	82,690	82,613.0	0.09	73	7	6	142	<1
	30	83,174	83,251	83,173.3	0.09	30	8	5	151	3
$dev\% = 10$	5	80,984	80,984	80,976.0	0.01	0	6	6	126	<1
	10	82,419	82,511	82,406.0	0.13	2	10	9	149	1
	15	83,709	83,771	83,702.0	0.08	153	15	14	165	1
	20	84,962	84,994	84,950.3	0.05	71	9	9	131	<1
	25	86,151	86,222	86,137.7	0.10	106	19	18	166	1
	30	87,286	87,286	87,243.7	0.05	0	9	9	153	<1
$dev\% = 20$	5	82,953	82,953	82,930.0	0.03	0	10	9	101	1
	10	85,786	85,876	85,778.0	0.11	38	18	17	120	1
	15	88,350	88,422	88,347.7	0.08	42	21	20	145	1
	20	90,835	90,895	90,834.3	0.07	324	22	20	128	2
	25	93,218	93,298	93,204.0	0.10	401	23	22	141	1
	30	95,422	95,642	95,411.0	0.01	2	19	18	149	<1

Table 2
Branch-and-cut statistics for U_Γ, Day 2.

	Γ	Obj. value	Root primal	Root LP rel.	Root % gap	B&B nodes	Total time	Sep. time	# cuts	Heur. time
$dev\% = 5$	5	75,014	75,014	74,940.1	0.10	0	6	5	51	<1
	10	75,718	75,718	75,632.7	0.11	0	5	5	50	<1
	15	76,357	76,357	76,308.4	0.06	0	6	5	49	1
	20	76,996	76,996	76,953.8	0.05	0	6	5	51	<1
	25	77,624	77,624	77,534.8	0.11	0	6	6	50	<1
	30	78,118	78,118	78,081.8	0.05	0	5	4	49	<1
$dev\% = 10$	5	75,964	75,964	75,915.3	0.06	0	7	7	50	<1
	10	77,355	77,355	77,287.5	0.09	0	8	7	49	<1
	15	78,695	78,695	78,611.4	0.11	0	8	7	50	<1
	20	79,953	79,953	79,851.1	0.13	0	7	7	48	<1
	25	81,076	81,076	81,033.8	0.05	0	9	8	52	<1
	30	82,168	82,168	82,110.6	0.07	0	7	7	50	<1
$dev\% = 20$	5	77,942	77,942	77,796.7	0.19	0	7	7	50	<1
	10	80,648	80,648	80,562.9	0.11	0	7	7	50	<1
	15	83,376	83,376	83,137.1	0.29	0	8	8	52	<1
	20	85,759	85,759	85,714.5	0.05	0	7	6	49	<1
	25	88,081	88,081	88,024.9	0.06	0	8	7	47	<1
	30	90,203	90,203	90,189.3	0.02	0	9	8	52	1

Table 3
Branch-and-cut statistics for U_Δ, Day 1.

	Δ	Obj. value	Root primal	Root LP rel.	Root % gap	B&B nodes	Total time	Sep. time	# cuts	Heur. time
$dev\% = 5$	22	83,961	84,008	83,946.0	0.07	1031	10	10	146	<1
	45	84,438	84,487	84,427.7	0.07	40	7	7	143	<1
$dev\% = 10$	22	88,237	88,294	88,219.9	0.08	163	60	60	121	<1
	45	88,924	88,924	88,901.3	0.03	0	15	15	126	<1
$dev\% = 20$	22	97,340	97,450	97,311.8	0.14	248	29	27	66	2
	45	98,128	98,595	98,113.3	0.49	97	22	22	73	<1

Table 4
Branch-and-cut statistics for U_{Δ}, Day 2.

	Δ	Obj. value	Root primal	Root LP rel.	Root % gap	B&B nodes	Total time	Sep. time	# cuts	Heur. time
dev% = 5	16	78,659	78,659	78618.4	0.05	28	9	8	47	< 1
	32	79,201	79,201	78811.7	0.49	0	5	4	32	< 1
dev% = 10	16	83,149	83,149	83049.3	0.13	0	5	4	26	< 1
	32	83,490	83,490	83383.8	0.13	25	8	7	38	< 1
dev% = 20	16	91,389	91,389	91314.8	0.08	0	13	11	26	2
	32	93,010	93,010	92631.3	0.40	0	9	8	26	< 1

Table 5
Branch-and-cut statistics for $U_{\Gamma\Delta}$, Day 1.

		Γ	Obj. value	Root primal	Root LP rel.	Root % gap	B&B nodes	Total time	Sep. time	# cuts	Heur. time
dev% = 5	$\Delta = 22$	5	79,909	79,967	79,902.7	0.08	485	64	63	135	< 1
		10	80,611	80,662	80,586.0	0.09	2	27	25	149	2
		15	81,258	81,258	81,240.3	0.02	0	31	31	151	< 1
		20	81,847	81,847	81,832.3	0.02	0	20	20	126	< 1
		25	82,456	82,456	82,416.1	0.05	0	17	15	126	2
		30	82,995	82,995	82,960.3	0.04	0	19	18	151	1
	$\Delta = 45$	5	80,043	80,128	80,010.7	0.15	770	12	12	139	< 1
		10	80,751	80,751	80,714.2	0.05	0	6	5	127	< 1
		15	81,374	81,434	81,360.0	0.09	113	16	16	132	< 1
		20	81,992	82,021	81,974.7	0.06	137	16	15	144	1
		25	82,577	82,611	82,548.0	0.08	390	16	16	116	< 1
		30	83,129	83,129	83,087.7	0.05	0	9	9	139	< 1
dev% = 10	$\Delta = 22$	5	80,043	80,043	80,018.8	0.03	0	65	62	125	3
		10	81,336	81,354	81,303.8	0.06	2	171	160	199	11
		15	82,572	82,572	82,530.9	0.05	0	133	118	151	15
		20	83,634	83,697	83,612.4	0.10	129	108	102	114	6
		25	84,674	84,941	84,656.6	0.34	109	131	126	121	5
		30	85,766	85,891	85,764.7	0.15	63	144	136	140	8
	$\Delta = 45$	5	80,813	80,813	80,805.3	0.01	0	10	8	100	2
		10	82,163	82,261	82,159.7	0.12	58	29	28	109	1
		15	83473	83,473	83,453.0	0.02	0	40	38	126	2
		20	84,663	84,663	84,639.5	0.03	0	41	39	126	2
		25	85,846	85,884	85,815.7	0.08	545	220	217	146	1
		30	86,909	86,909	86,887.3	0.02	0	54	50	176	4
dev% = 20	$\Delta = 22$	5	80,091	80,134	80,081.3	0.07	231	576	564	119	12
		10	82,267	82,397	82,264.7	0.16	39	804	763	150	41
		15	84,899	85,195	84,897.6	0.35	39	301	287	105	14
		20	87,319	87,537	87,284.8	0.29	813	1420	1406	127	11
		25	88,730	88,779	88,692.7	0.10	121	1254	1222	99	30
		30	90,962	91,035	90,944.6	0.10	60	271	249	89	22
	$\Delta = 45$	5	81,137	81,364	81,127.6	0.29	94	111	108	123	3
		10	83,780	83,844	83,754.7	0.11	120	255	248	122	7
		15	86,399	86,435	86,387.5	0.05	2	136	125	124	11
		20	88,584	88,772	88,567.7	0.23	75	106	101	93	5
		25	90,765	91,242	90,727.5	0.57	2	100	94	99	6
		30	93,052	93,261	93,010.8	0.27	359	171	165	125	6

generated by a *separation oracle*. Let $(\overline{\mathbf{x}}, \overline{\lambda})$ be a solution to the current problem. The *Separation Problem* (SEP) consists of finding a realization $\overline{\mathbf{b}}$ and a vector $\overline{\mathbf{y}} \in Y$ such that $\overline{\lambda} < (\overline{b}_t - \sum_{j \in J} a_{tj}\overline{x}_j)\overline{y}_t$ or prove that none exists. In the former case, the corresponding (violated) inequality (10) is added to the formulation. The complexity of the separation problem for the sets described in Section 3.2 is investigated in the next section.

4. The separation problem

In general, separation problems arising from two-stage RHSU models are non-convex and strongly NP-*hard* [29]. Luckily, it turns out that complexity is more favorable for the cases of our interest. It is worthwhile to start with recalling the complexity of SEP for U_{Γ}.

Table 6
Branch-and-cut statistics for $U_{\Gamma\Delta}$, Day 2.

		Γ	Obj. value	Root primal	Root LP rel.	Root % gap	B&B nodes	Total time	Sep. time	# cuts	Heur. time
$dev\% = 5$	$\Delta = 16$	5	74,717	74,717	74,709.9	0.01	2	27	26	49	1
		10	75,381	75,381	75,336.0	0.06	0	34	32	51	2
		15	76,306	76,306	75,925.6	0.50	0	15	13	28	2
		20	76,621	76,621	76,586.5	0.04	0	35	33	51	2
		25	77,232	77,232	77,207.0	0.03	0	37	34	51	2
		30	77,769	77,769	77,752.8	0.02	0	25	24	51	1
	$\Delta = 32$	5	75,014	75,014	74,940.1	0.10	0	11	10	51	1
		10	75,683	75,683	75,645.1	0.05	0	12	11	51	1
		15	76,377	76,377	76,305.4	0.09	0	13	12	51	1
		20	76,968	76,968	76,944.8	0.03	0	13	12	51	1
		25	77,545	77,545	77,511.0	0.04	0	15	14	51	1
		30	78,076	78,076	78,033.7	0.05	0	17	16	51	1
$dev\% = 10$	$\Delta = 16$	5	74,868	74,868	74,857.7	0.01	253	103	101	50	1
		10	76,107	76,107	76,105.7	0.00	2	56	50	49	3
		15	77,396	77,396	77,395.8	0	82	58	54	53	4
		20	78,547	78,547	78,543.0	0.01	41	73	67	50	4
		25	79,482	79,482	79,476.2	0.01	105	46	43	52	2
		30	80,625	80,625	80,613.1	0.01	18	57	55	45	2
	$\Delta = 32$	5	75,471	75,471	75,436.6	0.05	0	25	23	51	2
		10	76,986	76,986	76,645.3	0.44	0	23	20	32	3
		15	78,103	78,103	78,016.3	0.11	0	47	44	51	3
		20	79,206	79,206	79,196.9	0.01	54	58	57	49	1
		25	80,445	80,445	80,404.7	0.05	0	51	49	51	2
		30	81,674	81,674	81,308.8	0.45	0	16	15	27	1
$dev\% = 20$	$\Delta = 16$	5	74,871	74,871	74,867.2	0.01	0	63	57	51	6
		10	77,005	77,005	76,625.2	0.49	0	86	83	42	3
		15	79,152	79,152	79,140.0	0.02	0	115	106	51	9
		20	81,671	81,671	81,669.7	0.00	1	103	97	50	6
		25	83,691	83,691	83,600.9	0.11	0	50	46	27	4
		30	85,864	85,864	85,703.0	0.19	0	43	40	26	3
	$\Delta = 32$	5	75,900	75,900	75,828.4	0.09	0	67	64	51	3
		10	78,537	78,537	78,448.7	0.11	248	60	58	41	2
		15	81,032	81,032	80,806.1	0.28	0	35	33	26	2
		20	83,202	83,202	83,189.8	0.01	0	76	71	51	5
		25	85,449	85,449	85,338.2	0.13	0	36	33	26	3
		30	87,741	87,741	87,599.9	0.16	0	36	33	26	3

Theorem 5. *[34] Given* $\mathbf{x}$*, the corresponding worst case realization* $\mathbf{b}$ *is:*

$$b_t = \begin{cases} \tilde{b}_t + D_t & \text{if } t \in I \quad \text{and} \quad \left(\tilde{b}_t + D_t - \sum_{j \in J} a_{tj} x_j\right) w_t^u \geq \left(\sum_{j \in J} a_{tj} x_j - \tilde{b}_t\right) w_t^o \\ \tilde{b}_t - D_t & \text{if } t \in I \quad \text{and} \quad \left(\tilde{b}_t + D_t - \sum_{j \in J} a_{tj} x_j\right) w_t^u < \left(\sum_{j \in J} a_{tj} x_j - \tilde{b}_t\right) w_t^o \\ \tilde{b}_t & \text{if } t \notin I \end{cases}$$

where set $I \subseteq T$ is the set of the first Γ time periods according to non decreasing values of τ defined below.

$$\tau_t = \max\left\{ \left(\tilde{b}_t + D_t - \sum_{j \in J} a_{tj} x_j\right) w_t^u, \left(\sum_{j \in J} a_{tj} x_j - \tilde{b}_t - D_t\right) w_t^o \right\}$$
$$- \max\left\{ \left(\tilde{b}_t - \sum_{j \in J} a_{tj} x_j\right) w_t^u, \left(\sum_{j \in J} a_{tj} x_j - \tilde{b}_t\right) w_t^o \right\} \quad t \in T$$

Based on this result, we state the complexity of SEP for U_Γ.

Corollary 6. *SEP for U_Γ can be solved in time $O(m \log m)$.*

Proof. Given $\mathbf{x}$, compute worst-case realization $\mathbf{b}$ as in Theorem 5. Vectors $\mathbf{o}, \mathbf{u}$ and $\mathbf{y}$ can be obtained solving the personnel reallocation problem for the given $\mathbf{x}$ and $\mathbf{b}$, which can be done in time $O(m)$ (see Section 3.1). Therefore, the time complexity reduces to the one of ordering the τ values, that is, $O(m \log m)$.□

When U_Δ is considered, SEP can be solved in pseudo-polynomial time by reducing it to computing paths on a suitable Δ-uncertainty graph $G(V, E)$. Let us denote by K_t the set of integers in $[-D_t, D_t]$, $t \in T$. The graph can be formally defined as follows (see Fig. 2).

Definition 7. The Δ-uncertainty graph $G(V, E)$ is defined as:

$V = \{\sigma, \tau\} \cup \{v_{tk}, t \in T, k \in K_t\}$;

$E = E_\sigma \cup E_\tau \cup E_\Delta$, *where:*

$E_\sigma = \{(\sigma, v_{1k})$ *for each* $k \in K_1\}$;

$E_\tau = \{(v_{mk}, \tau)$ *for each* $k \in K_m\}$;

$E_\Delta = \{(v_{tp}, v_{(t+1)q})$, *for each* $p \in K_t, q \in K_{t+1}, \quad t \in T \setminus \{m\}$ *and*

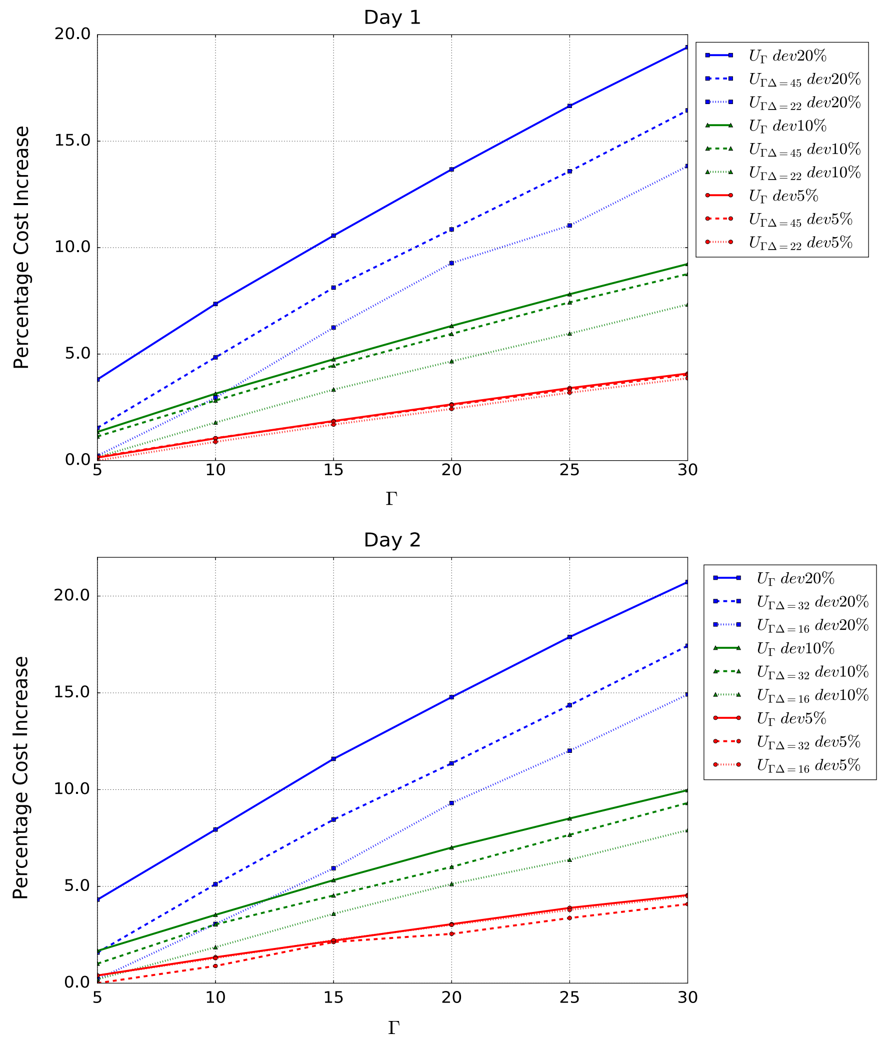

Fig. 4. Percentage cost increase for U_Γ and $U_{\Gamma\Delta}$.

$|p-q| \le \Delta(t+1)\}.$

Note that $G(V,E)$ is acyclic, $|V| = \sum_{t\in T}|K_t|+2$ and $|E| \le \sum_{t=1}^{m-1}|K_t|(2\Delta(t+1)+1)+|K_m|+|K_1|$. Furthermore, $|K_t| = 2D_t$. If we let $\overline{D} = \max_{t\in T} D_t$, then $|V| = O(\overline{D}m)$ and $|E| = O(\overline{D}^2 m)$. We show that solving the problem with U_Δ reduces to computing a longest-path in $G(V,E)$, which can be done in pseudo-polynomial time. For any $t \in T$ and $k \in K_t$, let c_{tk} be the minimum reallocation cost for time slot t when $b_t = \tilde{b}_t + k$ (see Section 3.1).

Theorem 8. *SEP for U_Δ can be solved in time $O(\overline{D}^2 m)$.*

Proof. Let us define arc weights as follows: 0 for $(\sigma, v_{1k}) \in E_\sigma$; c_{mk} for $(v_{mk}, \tau) \in E_\tau$; c_{tp} for $(v_{tp}, v_{(t+1)q}) \in E_\Delta$. By construction, SEP amounts to computing a longest $\sigma-\tau$ path on $G(V,E)$. Since G is acyclic, the problem can be solved in $O(|E|) = O(\overline{D}^2 m)$ time. □

We are now ready to address the uncertainty set $U_{\Gamma\Delta}$ of interest to our application. $U_{\Gamma\Delta}$ is described by integer variables and its continuous relaxation has exponentially many vertices which are not integer valued. The complexity of SEP for $U_{\Gamma\Delta}$ is open. However, the Δ-uncertainty graph allows us to prove that it can be solved in pseudo-polynomial time. In fact, when both cardinality and correlation are enforced, the longest path problem defined in Theorem 8 turns into a resource constrained longest-path problem on G.

Theorem 9. *SEP for $U_{\Gamma\Delta}$ can be solved in time $O(\Gamma\overline{D}^2 m)$.*

Proof. Let us define arc weights as in the proof of Theorem 8 and, for each arc, a further binary weight r_e equal to 1 for $e = (v_{tk}, j) \in E_\Delta \cup E_\tau, k \ne 0$ and 0 otherwise. That is, $r_e = 1$ only for arcs $e = (v_{tk}, j)$ such that node v_{tk} corresponds to some deviation from the nominal value. Since we have a bound Γ on number of possible deviations, any feasible path P must also satisfy the additional requirement $\sum_{e\in P} r_e \le \Gamma$. Therefore, SEP reduces to a single-resource constrained longest path problem, which can be solved in in time $O(\Gamma\overline{D}^2 m)$ by dynamic programming techniques (see e.g. [20]). □

These results have implications on the complexity of the continuous version of problem (3).

Corollary 10. *The LP relaxation of (3) is solvable in:*

1. *polynomial time for U_Γ;*
2. *pseudo-polynomial time for U_Δ and $U_{\Gamma\Delta}$.*

Proof. The results follow from the equivalence between optimization and separation [18].□

Thanks to the strong polynomiality of the corresponding SEP, U_Γ opens the way to a compact AAR reformulation. Unfortunately, using relaxation U_Γ leads to an unsatisfactory description of the uncertainty, suffering from overconservatism. Since SEP for $U_{\Gamma\Delta}$ is

Table 7
Robust vs traditional, Day 1.

	Δ	Γ	Robust cost			Traditional cost			% difference	
			Staff.	Reall.	Overall	Staff.	Reall.	Overall	Staff.	Overall
$dev\% = 5$	$\Delta = 22$	5	65,624	14,285	79,909	66,288	13,855	80,143	−1.01	−0.29
		10	65,736	14,875	80,611	66,288	14,945	81,233	−0.84	−0.77
		15	66,128	15,130	81,258	66,288	15,825	82,113	−0.24	−1.05
		20	66,272	15,575	81,847	66,288	16,405	82,693	−0.02	−1.03
		25	66,936	15,520	82,456	66,288	16,965	83,253	0.98	−0.97
		30	66,760	16,235	82,995	66,288	17,475	83,763	0.71	−0.93
	$\Delta = 45$	5	65,488	14,555	80,043	66,288	14,065	80,353	−1.22	−0.39
		10	65,256	15,495	80,751	66,288	15,195	81,483	−1.58	−0.91
		15	65,544	15,830	81,374	66,288	16,050	82,338	−1.14	−1.18
		20	65,792	16,200	81,992	66,288	16,670	82,958	−0.75	−1.18
		25	65,432	17,145	82,577	66,288	17,250	83,538	−1.31	−1.16
		30	66,344	16,785	83,129	66,288	17,795	84,083	0.08	−1.15
$dev\% = 10$	$\Delta = 22$	5	66,528	13,515	80,043	66,288	14,445	80,733	0.36	−0.86
		10	67,336	14,000	81,336	66,288	15,865	82,153	1.58	−1.00
		15	68,272	14,300	82,572	66,288	17,115	83,403	2.99	−1.01
		20	68,184	15,450	83,634	66,288	18,130	84,418	2.86	−0.94
		25	68,344	16,330	84,674	66,288	19,240	85,528	3.10	−1.01
		30	69,136	16,630	85,766	66,288	20,375	86,663	4.30	−1.05
	$\Delta = 45$	5	64,928	15,885	80,813	66,288	15,080	81,368	−2.09	−0.69
		10	65,328	16,835	82,163	66,288	17,355	83,643	−1.47	−1.80
		15	66,008	17,465	83,473	66,288	19,215	85,503	−0.42	−2.43
		20	66,128	18,535	84,663	66,288	20,375	86,663	−0.24	−2.36
		25	65,736	20,110	85,846	66,288	21,500	87,788	−0.84	−2.26
		30	66,624	20,285	86,909	66,288	22,560	88,848	0.51	−2.23
$dev\% = 20$	$\Delta = 22$	5	66,416	13,675	80,091	66,288	14,695	80,983	0.19	−1.11
		10	66,792	15,475	82,267	66,288	17,205	83,493	0.76	−1.49
		15	68,624	16,275	84,899	66,288	20,520	86,808	3.52	−2.25
		20	68,384	18,935	87,319	66,288	23,705	89,993	3.16	−3.06
		25	66,400	22,330	88,730	66,288	27,135	93,423	0.17	−5.29
		30	66,032	24,930	90,962	66,288	28,435	94,723	−0.39	−4.13
	$\Delta = 45$	5	66,312	14,825	81,137	66,288	16,215	82,503	0.04	−1.68
		10	67,280	16,500	83,780	66,288	19,490	85,778	1.50	−2.38
		15	67,304	19,095	86,399	66,288	22,575	88,863	1.53	−2.85
		20	67,224	21,360	88,584	66,288	25,740	92,028	1.41	−3.89
		25	67,040	23,725	90,765	66,288	28,500	94,788	1.13	−4.43
		30	67,392	25,660	93,052	66,288	30,840	97,128	1.67	−4.38

not polynomial in the strong sense, the same approach cannot be applied, because such reformulation requires the uncertainty set to be computationally tractable [5]. Solving it by the Δ-uncertainty graph as in Theorem 9 also has a few drawbacks: the problem size depends on the deviation, as the graph has $O(\overline{D}m)$ nodes and $O(\overline{D}^2 m)$ arcs, and arc weights $\mathbf{c}$ must be updated at each iteration of the cutting plane algorithm.

An alternative approach consists in formulating SEP as a Mixed Integer Program (MIP). For a given solution $\mathbf{x}$, we must compute $W_{\mathbf{x}(U)} = \max_{\mathbf{b} \in U}\{R_{\mathbf{xb}}\}$. Using expression (4) for $R_{\mathbf{xb}}$, we obtain a maxmin problem. The inner min can be removed, while preventing the unboundedness of the outer max problem, by using binary variables α to enforce optimality conditions for $R_{\mathbf{xb}}$, namely, $u_t o_t = 0$ for any time slot $t \in T$. The MIP formulation of SEP reads

$$\max \sum_{t \in T} (w_t^o o_t + w_t^u u_t)$$

$$u_t - o_t = b_t - \sum_{j \in J} a_{tj} x_j \quad t \in T$$

$$o_t \le M\alpha_t \quad t \in T$$

$$u_t \le M(1-\alpha_t) \quad t \in T$$

$$\mathbf{b} \in U, \mathbf{o}, \mathbf{u} \ge 0, \quad \boldsymbol{\alpha} \in \{0,1\}^{|T|} \tag{12}$$

It has $O(m)$ constraints and variables and, unlike the Δ-uncertainty graph, only m coefficients require to be updated at each iteration.

In a preliminary computational experience we compared two algorithms for SEP with $U_{\Gamma\Delta}$. The first is the algorithm proposed in [12] to solve the constrained shortest path problem on the Δ-uncertainty graph. The second consists of model (12) solved by the commercial solver IBM Cplex 12.6, with default settings. We observed that the two algorithms have similar performance for small $\overline{D}$ values, while the MIP performs better as $\overline{D}$ increases. For these reasons, we carry out our experiments by the MIP-based separation. It is also worth mentioning that the higher flexibility of the MIP approach could be valuable in order to take into account additional operational constraints (or a different uncertainty set).

5. Computational experience

In this section we investigate the application of the robust optimization approach in practical settings. We first give the

Table 8
Robust vs traditional, Day 2.

	Δ	Γ	Robust cost			Traditional cost			% difference	
			Staff.	Reall.	Overall	Staff.	Reall.	Overall	Staff.	Overall
$dev\% = 5$	$\Delta = 16$	5	58,152	16,565	74,717	58,456	16,365	74,821	−0.52	−0.14
		10	58,376	17,005	75,381	58,456	17,130	75,586	−0.14	−0.27
		15	57,736	18,570	76,306	58,456	17,955	76,411	−1.25	−0.14
		20	58,096	18,525	76,621	58,456	18,335	76,791	−0.62	−0.22
		25	58,272	18,960	77,232	58,456	18,920	77,376	−0.32	−0.19
		30	58,704	19,065	77,769	58,456	19,500	77,956	0.42	−0.24
	$\Delta = 32$	5	57,304	17,710	75,014	58,456	16,750	75,206	−2.01	−0.26
		10	57,728	17,955	75,683	58,456	17,680	76,136	−1.26	−0.60
		15	57,912	18,465	76,377	58,456	18,375	76,831	−0.94	−0.59
		20	57,528	19,440	76,968	58,456	19,020	77,476	−1.61	−0.66
		25	57,800	19,745	77,545	58,456	19,580	78,036	−1.13	−0.63
		30	57,936	20,140	78,076	58,456	20,025	78,481	−0.90	−0.52
$dev\% = 10$	$\Delta = 16$	5	58,448	16,420	74,868	58,456	16,720	75,176	−0.01	−0.41
		10	58,312	17,795	76,107	58,456	17,800	76,256	−0.25	−0.20
		15	58,576	18,820	77,396	58,456	18,955	77,411	0.21	−0.02
		20	59,152	19,395	78,547	58,456	20,250	78,706	1.19	−0.20
		25	58,752	20,730	79,482	58,456	21,145	79,601	0.51	−0.15
		30	59,520	21,105	80,625	58,456	22,320	80,776	1.82	−0.19
	$\Delta = 32$	5	57,896	17,575	75,471	58,456	17,240	75,696	−0.97	−0.30
		10	58,616	18,370	76,986	58,456	18,920	77,376	0.27	−0.51
		15	58,368	19,735	78,103	58,456	20,160	78,616	−0.15	−0.66
		20	58,256	20,950	79,206	58,456	21,475	79,931	−0.34	−0.92
		25	58,520	21,925	80,445	58,456	22,510	80,966	0.11	−0.65
		30	58,464	23,210	81,674	58,456	23,620	82,076	0.01	−0.49
$dev\% = 20$	$\Delta = 16$	5	58,496	16,375	74,871	58,456	16,770	75,226	0.07	−0.47
		10	59,480	17,525	77,005	58,456	19,080	77,536	1.75	−0.69
		15	58,712	20,440	79,152	58,456	21,355	79,811	0.44	−0.83
		20	58,696	22,975	81,671	58,456	23,650	82,106	0.41	−0.53
		25	57,856	25,835	83,691	58,456	26,335	84,791	−1.04	−1.31
		30	58,584	27,280	85,864	58,456	27,605	86,061	0.22	−0.23
	$\Delta = 32$	5	58,360	17,540	75,900	58,456	18,110	76,566	−0.16	−0.88
		10	58,632	19,905	78,537	58,456	21,210	79,666	0.30	−1.44
		15	58,432	22,600	81,032	58,456	23,570	82,026	−0.04	−1.23
		20	58,592	24,610	83,202	58,456	26,265	84,721	0.23	−1.83
		25	58,904	26,545	85,449	58,456	28,605	87,061	0.77	−1.89
		30	58,576	29,165	87,741	58,456	30,400	88,856	0.21	−1.27

details of our implementation and describe the test-bed. Then, the algorithm performance is discussed. Finally, we analyze the benefits of the robust methodology from managers' perspective. The experiments are run on 2 Intel Xeon 5150 processors clocked at 2.6 GHz with 8 GB of RAM in 4-thread mode. The commercial framework IBM Cplex 12.6 is used to implement a branch-and-cut algorithm in which we integrated our primal heuristic and separation routine. Computations are stopped either by 1 hour time limit or 0.05% optimality tolerance. A preliminary experience showed that the best performance is obtained by Cplex default settings with cutting plane generation (for all families of cuts) and MIP heuristics turned off. Separation is performed on all integer solutions, while fractional ones are tested only at the root node. The primal heuristic consists of rounding the current LP solution and computing the associated worst case uncertainty cost $W_{\mathbf{x}}(U)$, again by solving SEP. A time limit of 50 s is imposed to the heuristic: if an optimal solution to SEP has been obtained, then solution $(\mathbf{x}, W_{\mathbf{x}}(U))$ is returned, otherwise the heuristic fails.

5.1. Test-bed

The instances are based on real data, gathered during year 2008 from a large, distributed call center of an Italian Public Agency. The call center is on duty on working days from 7:45 a.m. to 8:00 p.m. corresponding to $T=49$ time slots (15 min). Three labor contracts are used, with 4, 6 or 8 h shifts. The costs ascribed to these shifts are 72, 96 and 112 € respectively. There are 34 4-h shifts, 26 6-h shifts and 18 8-h shifts. Agents are skilled for both front and back office and job flexibility is implemented by call center managers, who dynamically allocate agents to different activities according to operational needs. However, personnel reallocation requires some organizational set-up and should be limited. Inbound calls are prioritized over back office. Therefore, (front office) understaffing is considered more critical than overstaffing and $\max_{j \in J}\{c_j/s_j\} < w_t^o < w_t^u$, $t \in T$, where c_j is the cost of the shift and s_j the number of its time periods. Here, we considered as fair values $w_t^u = 10€$ and $w_t^o = 5€$ for every $t \in T$.

Staffing levels of two reference days, *Day 1* and *Day 2*, were provided us by the managers and are represented in Fig. 3.

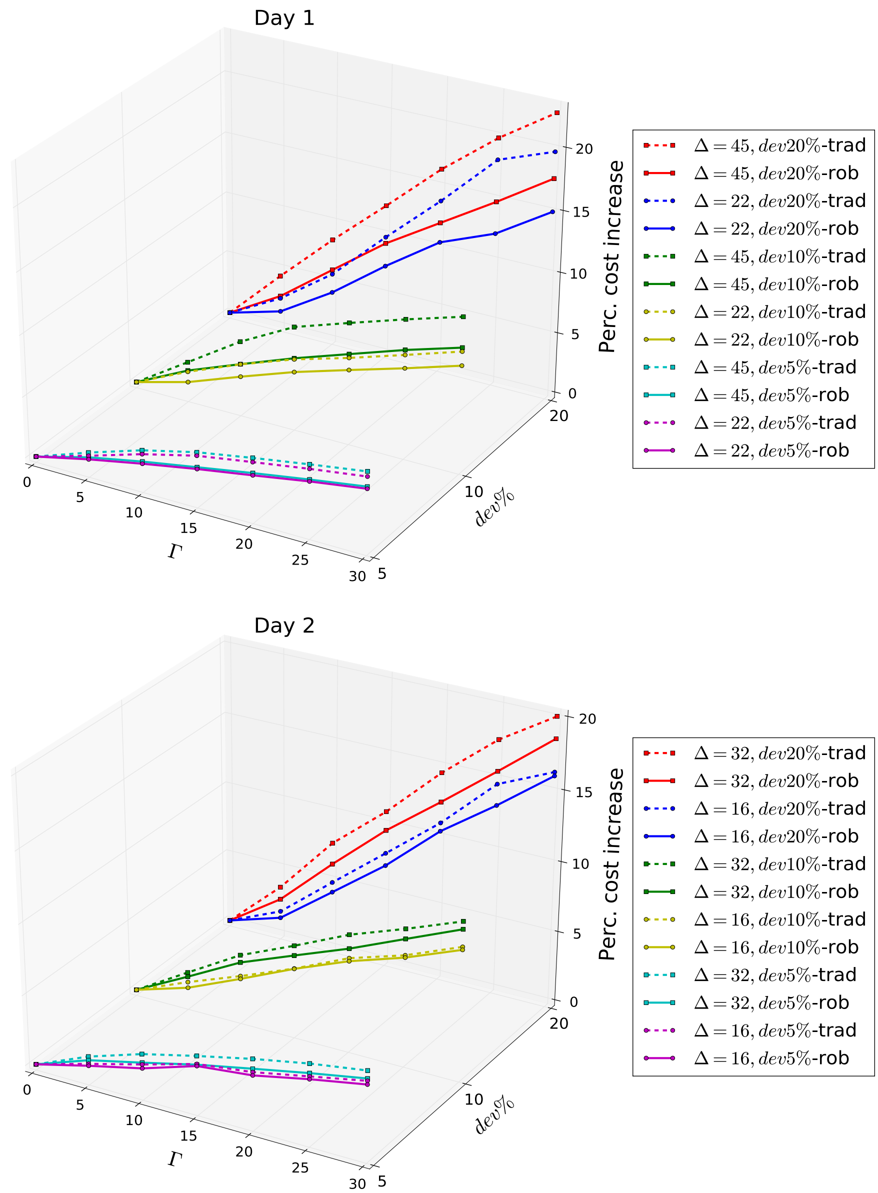

Fig. 5. Cost increase as a function of Γ and $dev\%$.

Starting from this data, we generated 60 instances for each reference day (namely, 36 for $U_{\Gamma\Delta}$, 18 for U_{Γ} and 6 for U_{Δ}) by systematically varying the percentage deviation $dev\%$ of the actual staffing level with respect to the nominal value, the number Γ of time slots affected by uncertainty and the allowed deviation difference Δ between two consecutive time periods, supposing it the same for every pair. We consider the values $dev\% \in \{5, 10, 20\}$, $\Gamma \in \{5, 10, 15, 20, 25, 30\}$ and $\Delta \in \{0.5\theta, \theta\}$, where θ is the average difference between two consecutive nominal staffing levels. For Day 1 we measured $\theta = 45$, yielding $\Delta \in \{22, 45\}$, for Day 2 we measured $\theta = 32$, yielding $\Delta \in \{16, 32\}$. The variation of the $dev\%$ parameter models the confidence of the managers in the nominal values, ranging from reliable estimates (e.g. days with standard demand patterns) to weak confidence caused by critical situations, like strikes in the sector of Agency's activity. Parameter Γ controls the trade-off between the robustness and the corresponding cost. Parameter Δ captures the correlation between the variations of consecutive time periods, intrinsic in call center dynamics.

5.2. Branch-and-cut performance

Traditionally, Benders reformulations may suffer from the weakness of the LP relaxation as well as from numerical difficulties due to nasty coefficients. In fact, specialized techniques have been recently proposed to overcome these drawbacks [9]. Interestingly, our formulation turns out to be not significantly affected by such problems. In Tables 1–6 the branch-and-cut statistics are reported for U_{Γ}, U_{Δ} and $U_{\Gamma\Delta}$. Besides the instance parameters, the tables contain: objective value, upper bound found by the primal heuristic at the root node (Root primal), value of the LP relaxation and the percentage gap at the root node, number of B&B nodes, total CPU time, separation time, number of generated cuts, time used by the heuristic. All times are expressed in seconds.

All the instances are solved in reasonable CPU time and limited number of branch-and-bound nodes (only in one case over 1000). These computing times are fully compatible with operational practice. No significant differences have been observed among the different uncertainty sets. Two major evidences explain such a nice behavior. The first deals with the quality of the LP relaxation. In fact, the gap at the root node never exceeds 1%. This is particularly valuable, as the results of Section 4 show that solving the LP relaxation is computationally accessible. The second evidence concerns the cuts. In our case:

1. integrality of b_t, w_t^u, w_t^o implies that all coefficients involved in inequalities (10) turn out to be integer (see Section 3.1);
2. figures m, b_t, w_t^u, w_t^o arising from real-world instances always produce cuts with limited *coefficient dynamism* [26].

This mitigates known numerical difficulties of Benders reformulation and the number of generated cuts remains small, guaranteeing a good convergence of the algorithm. Looking at the effect of the parameters, we observe that CPU times tend to increase as $dev\,\%$ gets larger, while it is not significantly affected by Γ. Conversely, times increase as Δ decreases. *Day1* and *Day2* show a similar behavior. Thanks to this nice computational behavior, we could evaluate the economical impact of robustness in a real-world setting, as described in the next section.

6. Economical analysis

Here we discuss how using a robust model instead of the traditional approach affects the overall costs. We first investigate the effect of including correlation between consecutive time periods in the model. Afterwards we compare the proposed robust approach with the current practice.

6.1. Uncertainty set comparison: U_Γ vs $U_{\Gamma\Delta}$

Graphs in Fig. 4 illustrate the cost variations induced by uncertainty sets U_Γ and $U_{\Gamma\Delta}$. Precisely, each line in the picture corresponds to a deviation $dev\,\% = \{5, 10, 20\}$ of the actual staffing level and to an uncertainty set. The chart describes the cost percentage increase with respect to the lowest cost value ($U_{\Gamma\Delta}$, $dev\ \%=5$, $\Delta = 0.5\theta$). In general, costs increase both with larger values of Δ (that is, by diminishing the correlation between slots) and with the deviation $dev\,\%$. When comparing cost increases under different uncertainty sets, one can observe that U_Γ implies consistently the highest costs. Variations are small (under 1%) for $dev\,\% = 5$, but significantly rise for $dev\,\% = \{10, 20\}$, reaching about 5%. This corresponds to a cost increase of about 5000 €/day, perceived as quite relevant in our context. Interestingly, correlation can act as the key control parameter of the level of conservatism, allowing to discard realizations which are well-known to be very unlikely to occur in practice.

6.2. Robust model vs traditional approach

We investigate the trade-off between level of protection and personnel cost when the uncertainty is modeled using $U_{\Gamma\Delta}$. Both the robust model presented in Section 3 and the traditional approach are considered.

Recall that the personnel cost has two components: the staffing cost, that is, the cost of the shifts associated with the $\mathbf{x}$ variables, and the personnel reallocation cost, associated with $\mathbf{o}$ and $\mathbf{u}$ variables. When the staffing levels are subject to uncertainty, personnel reallocation cost depends on the realization and its worst case value $W_{\mathbf{x}}(U)$ must be considered so as to be protected against all realizations. In the robust approach, both cost components are included in the solution to problem (9)–(11), since $W_{\mathbf{x}^*}(U) = \lambda^*$. In the traditional approach $\mathbf{x}$ values, along with the associated staffing cost, come from the solution of model (2) evaluated on nominal staffing levels. Therefore, the personnel reallocation cost for the worst case realization on the uncertainty set $U_{\Gamma\Delta}$ must be computed ex-post, that is, by solving problem (12) for the optimal solution $\mathbf{x}$ of (2).

Tables 7 and 8 report staffing, reallocation and overall costs for the robust and traditional approach, as well as staffing and overall percentage cost variations (last two columns). Fig. 5 summarizes the overall cost variation as a function of both Γ ($\Gamma = 0$ represents the overall nominal cost) and *dev%*. Dashed lines represent the traditional approach, solid ones the robust approach. The graphs give evidence to the fact that the traditional approach cost grows more than the robust one with both deviation *dev%* and Γ. Such an increase is more noticeable in *Day1*, whereas in *Day2* costs are closer. This issue is often underestimated as the urgency of covering understaffing situations at the front office is perceived more critical at real-time level. In general, the traditional approach can be by far more expensive than the corresponding robust method in case of uncertainty. The overall difference rises up to 5% in *Day1* and about 2% in *Day2*, corresponding to ~5000 and 2000 €/day, respectively.

Management major concern about robust planning is the staffing cost increase. Recall that, in our setting staffing cost is paid by the company whatever realization occurs. The analysis of the disaggregated costs shows that the robust model not only leads to cheaper overall costs, but also to possibly cheaper staffing costs (15 cases out of 36 in *Day1* and 19 out of 36 in *Day2*). In general, there may exist different staffing allocations with similar nominal cost and trend, but with substantially different ability to cope with uncertainty. Among them, the robust model is able to choose the best option to handle uncertainty, whereas the traditional approach is not guaranteed to do the same. Finally, Fig. 6 shows a qualitative comparison between the staffing allocations produced by the robust and the traditional approach for problem with *dev%=10*, $\Gamma = 20$, $\Delta = \theta$. In both days the robust staffing cost is smaller than the corresponding traditional cost. Both models prefer understaffing versus overstaffing, but the robust method favors a small level of understaffing at the beginning of the day and a larger one at the end, whereas the traditional approach does the opposite. We mention that in our application managers recognized this trend as a good operational practice, because it

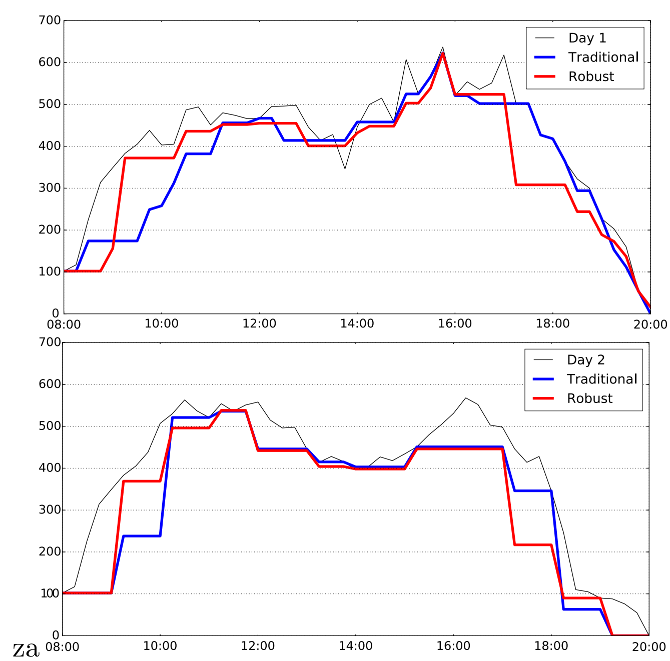

Fig. 6. Staffing comparison, *dev%=10*, $\Gamma = 20$, $\Delta = \theta$ (45 for *Day1* and 32 for *Day2*).

gives them more time to organize the back office work before moving people to front office.

7. Conclusions

We investigated a two-stage robust optimization model for shift scheduling with uncertain staffing levels (i.e., right-hand-sides). In contrast to the general case, we showed that the separation problem associated with the constraints of a Benders-like reformulation can be solved rather efficiently, even when correlation between consecutive time periods is taken into account. This is a key issue as not including correlation in the model leads to over-conservatism. Therefore, our study opens the way to practical algorithms able to improve the WFM process. In fact, our method interfaces with any staffing procedure and benefits from more accurate staffing estimates, as the more the confidence the cheaper the protection cost.

Acknowledgements

We are grateful to James Luedtke for his helpful comments.

References

[1] Ahuja R, Magnanti T, Orlin J. Network flows: theory, algorithms and applications. Prentice Hall, Upper Saddle River, New Jersey 07458; 1993.
[2] Aksin Z, Armony M, Mehrotra V. The modern call center: a multi-disciplinary perspective on operations management research. Production and Operations Management 2007;16(6):665–88.
[3] Atlason J, Epelman M, Henderson S. Optimizing call center staffing using simulation and analytic center cutting-plane methods. Management Science 2008;54(2):295–309.
[4] Avramidis A, Deslauriers A, L'Ecuyer P. Modeling daily arrivals to a telephone call center. Manag Sci 2004;50(7):896–908.
[5] Ben-Tal A, Goryashko A, Guslitzer E, Nemirovski A. Adjusting robust solutions of uncertain linear programs. Mathematical Programming 2004;99:351–76.
[6] Ben-Tal A, Nemirowski A. Robust convex optimization. Mathematics of Operations Research 1998;23:769–805.
[7] Benders JF. Partitioning procedures for solving mixed-variables programming problems. Numerische Mathematik 1962:238–52.
[8] Bertsimas D, Sim M. The price of robustness. Operations Research 2004;52(1):35–53.
[9] Bodur M, Luedtke J. Mixed-integer rounding enhanced Benders decomposition for multiclass service system staffing and scheduling with arrival rate uncertainty. Management Science 2016. http://dx.doi.org/10.1287/mnsc.2016.2455.
[10] Cezik T, L'Ecuyer P. Staffing multiskill call centers via linear programming and simulation. Management Science 2008;54(2):310–23.
[11] Defraeye M, Van Nieuwenhuyse I. Staffing and scheduling under nonstationary demand for service: a literature review. Omega 2016;58:4–25.
[12] Dumitrescu I, Boland N. Improved preprocessing, labeling and scaling algorithms for the weight-constrained shortest path problem. Networks 2003;42:135–53.
[13] Durante A. The costs of call center overstaffing and understaffing. ⟨https://blog.thecallcenterschool.com/the-costs-of-call-center-overstaffing-and-understaffing?gtld=us⟩, 2015.
[14] Fluss D. The workforce management overstaffing problem. ⟨http://blog.intradiem.com/workforce-management-overstaffing-problem/⟩, 2012.
[15] Gans N, Koole G, Mandelbaum A. Telephone call centers: tutorial, review and research prospects. Manufacturing and Service Operations Management 2003;5(2):79–141.
[16] Gorissen B, den Hertog D. Robust counterparts of inequalities containing sums of maxima of linear functions. European Journal of Operations Research 2013;227:30–43.
[17] Green L, Kolesar P, Soares J. Improving the sipp approach for staffing service systems that have cyclic demands. Operations Research 2001;49(4):549–64.
[18] Grötschel M, Lovász L, Schrijver A. The ellipsoid method and its consequences in combinatorial optimization. Combinatorica 1981;1(2):169–97.
[19] Gurvich I, Luedtke J, Tezcan T. Staffing call centers with uncertain demand forecasts: a chance-constrained optimization approach. Management Science 2010;56(7):1093–115.
[20] Hassin R. Approximation schemes for the restricted shortest path problem. Mathematics of Operations Research 1992;17(1):36–42.
[21] Kelley J. The cutting-plane method for solving convex programs. Journal of the Society for Industrial and Applied Mathematics 1960;8(4):703–12.
[22] Koole G. Call center mathematics. A scientific method for understanding and improving contact centers. MG Books 2013.
[23] Koole G, van der Sluis E. Optimal shift scheduling with a global service level constraint. IIE Transactions (Institute of Industrial Engineers) 2003;35(11):1049–55.
[24] Liao S, Koole G, van Delft C, Jouini O. Staffing a call center with uncertain non-stationary arrival rate and flexibility. OR Spectrum 2012;34(3):691–721.
[25] Liao S, van Delft C, Vial J-P. Distributionally robust workforce scheduling in call centres with uncertain arrival rates. Optimization Methods and Software 2013;28(3):501–22.
[26] Margot F. Testing cut generators for mixed-integer linear programming. Mathematical Programming Computation 2009;1(1):69–95.
[27] Mattia S, Rossi F, Servilio M, Smriglio S. Robust shift scheduling in call centers. In: Combinatorial Optimization - Proceedings of the Third International Symposium, ISCO 2014, Lisbon, Portugal, March 5–7, 2014, volume 8596 of LNCS, pp. 336–346, 2014.
[28] Minoux M. Robust linear programming with right-hand-side uncertainty, duality and applications. in: C. Floudas and P. Pardalos, editors, Encyclopedia of Optimization, pp. 3317–3327. Springer US, 2009.
[29] Minoux M. On 2-stage robust LP with RHS uncertainty: complexity results and applications. Journal of Global Optimization 2011;49(3):521–37.
[30] Orlin J. A faster strongly polynomial algorithm for the minimum cost flow problem. Operations Research 1993;41:338–50.
[31] Özbay N, Bienstock D. Computing robust basestock levels. Discrete Optimization 2008;5:389–414.
[32] Segal M. The operator-scheduling problem: a network-flow approach. Oper Res 1974;22(4):808–23.
[33] Steckley S, Henderson S, Mehrotra V. Forecast errors in service systems. Probability in the Engineering and Informational Sciences 2009;23(2):305–32.
[34] Thiele A, Terry T, Epelman M. Robust linear optimization with recourse. Rapport Technique 2009:4–37.
[35] Thomas R, Harrison T. A stochastic programming model for scheduling call centers with global service level agreements. European Journal of Operational Research 2010;207(3):1608–19.
[36] Whitt W. Staffing a call center with uncertain arrival rate and absenteeism. Production Operation and Management 2006;15(1):88–102.