European Journal of Operational Research 279 (2019) 721–731

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

Analysis of models for the Stochastic Outpatient Procedure Scheduling Problem

Karmel S. Shehadeh[*], Amy E. M. Cohn, Marina A. Epelman

Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI 48109, United States

ARTICLE INFO

Article history:
Received 4 September 2018
Accepted 10 June 2019
Available online 13 June 2019

Keywords:
OR in health services
Appointment scheduling
Outpatient clinics
Stochastic programming
Mixed-integer programming

ABSTRACT

In this paper, we present a new stochastic mixed-integer linear programming model for the Stochastic Outpatient Procedure Scheduling Problem (SOPSP). In this problem, we schedule a day's worth of procedures for a single provider, where each procedure has a known type and associated probability distribution of random duration. Our objective is to minimize the expectation of a weighted sum of patient waiting time, provider idling, and clinic overtime. We present computational results to show the size and characteristics of problem instances that can be solved with our model. We also compare this model to other formulations in the literature and analyze them both empirically and theoretically, demonstrating where significant improvements in performance can be gained with our proposed model. This work is motivated by our research on developing scheduling templates for endoscopic procedures at a major medical center. More broadly, however, the SOPSP is a stochastic single-resource sequencing and scheduling problem and therefore has applications both within and outside of healthcare operations.

1. Introduction

In this paper, we address the Stochastic Outpatient Procedure Scheduling Problem (SOPSP), which arises in outpatient procedure centers (OPCs). In this problem, we consider the perspective of an OPC manager who must schedule the start times for a day's worth of procedures for a single provider, where each procedure has a known type and a random (non-negative) duration that follows a known probability distribution associated with the procedure type. Given the uncertainty in procedure durations, the goal is to minimize the expectation of a weighted sum of total patient waiting time (the time from the scheduled start of a procedure to its actual start), total provider idle time (the time from the end of one procedure to the start of the next), and clinic overtime (the time from the scheduled closing time of the clinic to the end of the last procedure of the day).

This research is motivated by our work with the University of Michigan Medical Procedures Unit, an OPC that performs a variety of endoscopic procedures such as colonoscopies. The ultimate goal of this project is to optimize daily schedule templates and policies for filling these templates, to best account for variability in patient procedure times. By building higher-quality schedules that incorporate the variability in procedure durations, it is possible to improve patient and provider satisfaction, reduce costs, and even achieve better clinical outcomes. A valuable tool in creating such templates is the ability to solve the simpler (and yet still challenging) SOPSP as an embedded sub-problem.

In addition to the value that the ability to solve the SOPSP provides to our work, it also has relevance for many other applications, including scheduling of surgeries in an operating room, ships in a port, exams in an examination facility, and more (Ahmadi-Javid, Jalali, & Klassen, 2017; Begen & Queyranne, 2011; Mancilla & Storer, 2012; Robinson & Chen, 2003; Sabria & Daganzo, 1989). For example, it is a common practice for surgeries to initially be assigned to a surgeon, date, and operating room several weeks or even months before their scheduled date. The actual scheduled start times for these surgeries, however, are typically not set until a few days in advance. It is at this point when the SOPSP can be solved to construct the final surgical schedule and notify the patients when to report to the hospital (see Denton, Miller, Balasubramanian, & Huschka, 2010; Mancilla & Storer, 2012, and references therein for more details).

The SOPSP is also computationally challenging to solve, for a number of reasons. First, it is a complex combinatorial optimization problem, given the inherent implied sequencing problem that underlies assigning appointment times to each patient (Ahmadi-Javid et al., 2017; Berg, Denton, Erdogan, Rohleder, & Huschka, 2014; Mancilla & Storer, 2012). Second, the problem is inherently stochastic due to the uncertainty in procedure durations. Finally, it is also a multi-criteria optimization problem, in which we must

[*] Corresponding author.
E-mail address: ksheha@umich.edu (K.S. Shehadeh).

https://doi.org/10.1016/j.ejor.2019.06.023
0377-2217/© 2019 Elsevier B.V. All rights reserved.

make trade-offs between longer spacing between appointments, which leads to reduced patient delays, and shorter spacing, which leads to less provider idling and overtime (Ahmadi-Javid et al., 2017; Antunes, Alves, & Clímaco, 2016; Cayirli & Veral, 2003; Denton et al., 2010; Gupta & Denton, 2008; Mancilla & Storer, 2012; Marler & Arora, 2004; T'kindt & Billaut, 2006). More broadly, the SOPSP is a single-server stochastic appointment sequencing and scheduling (SASS) problem, the underlying complexity of which has been studied by several previous authors beginning with the seminal work of Welch and Bailey (1952) and Weiss (1990) (see Ahmadi-Javid et al., 2017; Berg et al., 2014; Denton et al., 2010; Gupta, 2007; Gupta & Denton, 2008; Mancilla & Storer, 2012, and references therein).

In this paper, we present a new Stochastic Mixed-Integer Linear Program (SMILP) using Sample Average Approximation (SAA) for solving the SOPSP, with a focus both on *tractability* (i.e., being able to solve problem instances of realistic sizes in an acceptable amount of time) and *implementability* (i.e., proposing a model that can be easily translated into standard optimization software packages, not requiring customized algorithmic development or tuning). To provide context within the literature, we compare our model with those of Berg et al. (2014) (an enhancement of Denton, Viapiano, & Vogl, 2007) and Mancilla and Storer (2012), which are, to the best of our knowledge, the only SMIPs for SASS with waiting, idling, and overtime costs. We discuss the relative strengths and weaknesses of the three models and then compare them computationally under a common, straightforward software implementation.

The remainder of the paper is structured as follows. In Section 2, we present the relevant literature. In Section 3, we introduce and analyze three mathematical models of the SOPSP: two based on prior literature (Berg et al., 2014 and Mancilla & Storer, 2012), and a new model. After that, in Section 4, we compare the computational performance of the three models and provide some discussion and insights. Finally, conclusions are drawn in Section 5.

2. Literature review

Outpatient scheduling problems have been an active area of research since the seminal work of Welch and Bailey (1952). Comprehensive surveys of results obtained since then include Cayirli and Veral (2003), Gupta and Denton (2008), and Ahmadi-Javid et al. (2017). Within this literature, there are two primary approaches to stochastic appointment scheduling. The first is to develop and evaluate scheduling heuristics, often through the use of simulation (see, for example, Ahmadi-Javid et al., 2017; Ho & Lau, 1992; Klassen & Rohleder, 1996; Rohleder & Klassen, 2000; Vissers & Wijngaard, 1979). The second is to construct models and design algorithms to find optimal schedules through the use of queueing theory (see, for example, Bosch & Dietz, 2000; Jansson, 1966; Mercer, 1960; Sabria & Daganzo, 1989; Soriano, 1966; Vanden Bosch & Dietz, 2001, and references therein), stochastic programming (see, for example, Berg et al., 2014; Denton & Gupta, 2003; Mancilla & Storer, 2012; Robinson & Chen, 2003, and references therein), and, more recently, robust and distributionally robust optimization (RO and DRO, respectively; see, for example, Jiang, Shen, & Zhang, 2017; Mak, Rong, & Zhang, 2014, and references therein).

Herein, we present studies that are most relevant to us: papers that use SMILP models to address offline single-resource stochastic appointment sequencing and scheduling (SASS) problems that are similar to the SOPSP ("offline" in the sense that sequencing and scheduling decisions are all made ahead of time). We are interested in generating optimal solutions to the SOPSP assuming knowledge of the distributions of appointment durations (a classic SASS assumption, Ahmadi-Javid et al., 2017; Berg et al., 2014; Deceuninck, Fiems, & De Vuyst, 2018), which rules out both the heuristic approach (due to sub-optimality and lack of performance guarantees, Ahmadi-Javid et al., 2017; Ho & Lau, 1992; Klassen & Rohleder, 1996; Rohleder & Klassen, 2000; Vissers & Wijngaard, 1979) and the RO and DRO-based approaches (which assume distributional ambiguity). Finally, as pointed out by Robinson and Chen (2003), queueing theory-based results and algorithms are not appropriate for the SOPSP and other OPC scheduling problems which involve serving a finite number of patients within fixed service hours (i.e., the queue never reaches a steady state).

Papers that present models and algorithms for optimizing SASS decisions using SMILP fall into two groups: those that focus on determining the optimal start times (or, equivalently, the inter-arrival times) assuming that the sequence of patients (customers) is already fixed (e.g., through the use of a heuristic, see, for example, Bosch & Dietz, 2000; Denton & Gupta, 2003; Erdogan & Denton, 2013; Ge, Wan, Wang, & Zhang, 2013; Robinson & Chen, 2003; Vanden Bosch & Dietz, 2001, and references therein), and those that focus on optimizing the sequencing and scheduling decisions simultaneously. Since we consider both sets of decisions, we further limit the scope of this review to the latter category. We refer the reader to the following studies: Ahmadi-Javid et al. (2017), Berg et al. (2014), Cayirli, Veral, and Rosen (2006), Cayirli, Veral, and Rosen (2008), Creemers, Beliën, and Lambrecht (2012a), Creemers, Colen, and Lambrecht (2012b), Gupta and Denton (2008), Rohleder and Klassen (2000), Salzarulo, Mahar, and Modi (2016), and references therein, which demonstrate the benefit of sequencing heterogeneous patient appointments based on their characteristics for improving clinic performance and reducing costs compared to fixed sequence approaches. To the best of our knowledge, and according to the recent review of outpatient appointment systems by Ahmadi-Javid et al. (2017), papers by Denton et al. (2007), Berg et al. (2014), and Mancilla and Storer (2012) are the ones most closely related to our work, addressing similar SASS problems with waiting, idling, and overtime costs using SMILP.

Denton et al. (2007) formulated the stochastic surgery scheduling problem in an operating room (OR) as a two-stage SMILP with binary precedence variables and continuous time allowance variables in the first stage, and continuous waiting, idling, and overtime variables in the second stage. They used the sample-average approximation approach (i.e., a scenario-based approach) to replace the continuous distributions of surgery durations with approximate discrete distributions by considering a sample of N randomly generated scenarios. Since it was difficult to solve instances with more than 4 surgeries, they proposed several sequencing heuristics and then obtained the optimal surgery start times, for a fixed sequence, via the L-shaped algorithm (Birge & Louveaux, 2011) described in Denton and Gupta (2003). Their results showed substantial potential reductions in surgeon waiting, OR idling, and overtime costs by sequencing surgeries based on variances of their durations compared to the schedule of the OR that the study considered.

In a slightly different setting, Berg et al. (2014) considered the problem of optimizing the booking (number of patients to schedule) and appointment time decisions for outpatient procedures under no-show and procedure durations uncertainties. The goal was to maximize profit, i.e., the difference between the expected revenue and the expected variable cost of patient waiting time, provider idle time, and overtime associated with scheduling patients. Since the revenue was straightforward to compute, the paper focused on minimizing the expected variable cost determined by sequencing and scheduling decisions (a SASS problem which is, to some extent, similar to the SOPSP). To that end, the paper extended and enhanced the SMILP model of Denton et al. (2007) by including heterogeneous no-show probabilities and using both precedence and assignment variables to strengthen the earlier model, and employed three exact solution methods:

L-shaped, hybrid multi-cut L-shaped with scenario aggregation and ranking (to overcome the computational burden of the original multi-cut method, see Birge & Louveaux, 1988), and branch-and-bound with progressive hedging as a primal heuristic (Rockafellar & Wets, 1991). While these methods were computationally competitive (relative to each other) in solving small instances (≤ 5 patients), it was challenging to solve larger instances (10 patients), primarily due to the stochastic and combinatorial elements of the problem. Therefore, they proposed six sequencing heuristics based on standard deviations of procedure durations and no-show probabilities, and illustrated the conditions under which some of these provided a near-optimal solution to the problem.

Mancilla and Storer (2012) formulated the surgery sequencing and scheduling problem in a single operating room at a local hospital as a stochastic mixed-integer program with sample average approximation. The model differs from that of Denton et al. (2007) in the following two ways. First, they replaced binary precedence variables with binary sequence position assignment variables (previously proposed in Wagner, 1959). Second, they replaced continuous job time allowance variables with continuous appointment (start) time variables. Additionally, using concepts from Garey, Johnson, and Sethi (1976), they proved that for two scenarios and equal idling costs but different waiting costs for each job, the finite scenario SAA problem is NP-complete. Therefore, to overcome the computational burden of the sequencing decisions, they developed an algorithm to generate a near-optimal sequence, with the resulting linear subproblem of determining appointment times solved within their algorithm using the CPLEX barrier method. Given that the SMILP studied in Mancilla and Storer (2012) is a variation of the one in Denton et al. (2007), and the one in Berg et al. (2014) is stronger than Denton et al. (2007), in this paper, we focus our analysis on the models of Mancilla and Storer (2012) and Berg et al. (2014).

Finally, we point out the similarities and differences between single provider stochastic appointment sequencing and scheduling and single machine scheduling (SMS). At the outset, they look similar: the provider can be thought of as a single machine, and procedures and their durations as jobs and their processing times, respectively (see Forst, 1993; Lawler, Lenstra, Kan, & Shmoys, 1993; Pinedo, 2016 for machine scheduling literature). Nevertheless, SASS is materially different from SMS. In SMS problems, each job release time (the time at which the job becomes available for processing) is typically exogenous (i.e., a parameter). In contrast, the appointment time in SASS, which can be thought of as a release time at which the scheduled patient is presumably available for the procedure, is a decision variable. Furthermore, in the classic SMS problem, one scheduling criterion that has received the most attention over the years is minimizing makespan (i.e., completing the last job at the earliest possible time), which trivially minimizes overtime but does not consider patient waiting time nor provider idle time. Our SMILP model, as well as those of Mancilla and Storer (2012) and Berg et al. (2014), however, improve on some ideas from the seminal work of Wagner (1959) and Pinto and Grossmann (1998) in the domain of deterministic single-machine jobs/tasks sequencing and scheduling.

3. Stochastic mixed-integer linear programming models of the SOPSP

In this section, we present and analyze three SMILP formulations for the SOPSP. First, we define the problem formally. Then, we present our SMILP formulation and the conditions under which it is equivalent to two closely-related stochastic appointment sequencing and scheduling SMILPs in the literature, those of Mancilla and Storer (2012) and Berg et al. (2014), which are also presented for completeness.

3.1. Formal statement of the problem

We consider the problem of sequencing a set of procedures for a single provider (where each procedure has a known type and a random, non-negative, duration that follows a known probability distribution associated with the procedure type) and determining the associated scheduled start time for each procedure. The performance metric is the weighted sum of three components, total patient waiting time (the time from the scheduled start of a procedure to its actual start), total provider idle time (the time from the end of one procedure to the start of the next), and overtime (the time from the scheduled closing time of the clinic to the end of the last procedure of the day). Given a set of procedures, their sequence, their scheduled start times, and the distributions of their durations, the expected value of this weighted sum can be estimated by averaging over finitely many realizations (a sample) of procedure durations. This sample average is the objective function of the forthcoming optimization problems. We make the following assumptions:

A1. A procedure is not permitted to start before its scheduled start time nor the completion time of the previous procedure.
A2. Although patients may fail to show up to their appointments, we assume that those who do show up are punctual, i.e., available at the scheduled start times of their procedures.
A3. The provider is always available at the start of the day, and immediately after each procedure.
A4. There is no opportunity to modify the schedule on the day of service, i.e., rescheduling during the day or adding procedures (to accommodate walk-ins or emergencies) is not permitted.

The problem can be formulated as a two-stage SMILP with binary (for *sequencing*) and continuous (for *scheduling*, i.e., start times) first-stage variables and continuous second-stage variables representing what happens for each realization of procedure durations (waiting time, idle time, and overtime), given the sequence of appointment times decided in the first stage. To incorporate procedure duration uncertainty into the model, we use a Sample Average Approximation (SAA) approach as in Robinson and Chen (2003), Denton et al. (2010), and Mancilla and Storer (2012). That is, we generate a sample of N scenarios (each scenario consists of a vector of realizations of procedure durations which are drawn independently from the distributions corresponding to each patient's type; a no-show patient can be represented by a realized procedure duration of 0), and then optimize the sample average of the weighted sum of the three metrics using the stored sample. (The technical details of sample average approximation approach are out of the scope of this paper, and we refer the reader to Kim, Pasupathy, & Henderson, 2015; Kleywegt, Shapiro, & Homem-de Mello, 2002; Mak, Morton, & Wood, 1999; Molina-Pariente, Hans, & Framinan, 2016; Shapiro & Homem-de Mello, 2000, and references therein, for a thorough discussion.)

3.2. Formulations of the problem

Table 1 summarizes notation and some terminology used in our sample-average SMILP formulation of the SOPSP. Note, in particular, that we use the term "appointment" to refer to a position in the sequence, and use the terms "patient" and "procedure" interchangeably. Using this notation, the problem can be formulated as

Table 1
Notation.

Indices	
p	index of patients, or procedures, to be scheduled, $p = 1, \ldots, P$
i	index of positions in the sequence, or appointments, $i = 1, \ldots, P$
n	index of scenarios to be considered, $n = 1, \ldots, N$
Parameters	
λ_i^w	waiting time penalty for appointment i
λ_i^g	penalty for idle time between appointments i and $i+1$
λ^o	overtime penalty
$\mathcal{L}$	planned length of clinic day
d_p^n	duration of procedure p in scenario n
Scenario-independent (first-stage) variables	
$x_{i,p}$	binary assignment variable indicating whether procedure p is assigned to appointment i
t_i	scheduled start time of appointment i
Scenario-dependent (second-stage) variables	
s_i^n	actual start time of appointment i in scenario n
g_i^n	idle time after appointment i in scenario n
o^n	overtime in scenario n

Table 2
Additional notation (Mancilla & Storer, 2012).

Parameters	
λ_p^w	waiting time penalty for procedure p
λ_p^g	idle time penalty for procedure p
Scenario-dependent (second-stage) variables	
$w_{i,p}^n$	waiting time of procedure p in scenario n, if it is assigned to appointment i (0 otherwise)
$g_{i,p}^n$	idle time after procedure p in scenario n, if it is assigned to appointment i (0 otherwise)
e^n	slack variable measuring early completion of the schedule in scenario n

follows:

$$\text{(S) minimize } \frac{1}{N}\sum_{n=1}^{N}\left[\sum_{i=1}^{P}\lambda_i^w \cdot (s_i^n - t_i) + \sum_{i=1}^{P}\lambda_i^g \cdot g_i^n + \lambda^o \cdot o^n\right] \tag{1a}$$

$$\text{subject to } \sum_{i=1}^{P} x_{i,p} = 1 \qquad \forall p \tag{1b}$$

$$\sum_{p=1}^{P} x_{i,p} = 1 \qquad \forall i \tag{1c}$$

$$s_i^n \geq t_i \qquad \forall i, n \tag{1d}$$

$$s_i^n \geq s_{i-1}^n + \sum_{p=1}^{P} d_p^n \cdot x_{i-1,p} \qquad \forall (i \geq 2, n) \tag{1e}$$

$$g_i^n = s_{i+1}^n - \left(s_i^n + \sum_{p=1}^{P} d_p^n \cdot x_{i,p}\right) \qquad \forall (i < P, n) \tag{1f}$$

$$o^n \geq \left(s_P^n + \sum_{p=1}^{P} d_p^n \cdot x_{P,p}\right) - \mathcal{L} \qquad \forall n \tag{1g}$$

$$(g_i^n, s_i^n) \geq 0 \qquad \forall (i, n) \tag{1h}$$

$$o^n \geq 0 \qquad \forall n \tag{1i}$$

$$t_i \geq 0 \qquad \forall i \tag{1j}$$

$$x_{i,p} \in \{0, 1\} \qquad \forall (i, p) \tag{1k}$$

In the above formulation, the objective function in (1a) is the sample average of the weighted linear combination of the total waiting time, total idle time, and overtime cost. Constraints (1b) and (1c) ensure that each procedure is assigned to one appointment and each appointment is assigned one procedure. For every scenario n, constraints (1d) and (1e) require the actual start time, s_i^n, of the ith appointment to be no smaller than the scheduled start time, t_i, and than the completion time of the preceding appointment, i.e., the $(i-1)$st appointment's actual start time, s_{i-1}^n, plus the duration of the procedure assigned to it, $\sum_{p=1}^{P} d_p^n \cdot x_{i-1,p}$. The ith appointment waiting time is the difference between its actual and scheduled start time (i.e., $s_i^n - t_i$), which we include in the objective function directly. Constraints (1f) define the idle time between two consecutive appointments as the gap between the actual start time of an appointment and the completion time of the preceding one. Constraints (1g) and (1i) define overtime (if any) as the positive difference between the completion time of the last appointment and the clinic scheduled closing time, $\mathcal{L}$. Finally, the remaining constraints specify feasible ranges of the decision variables.

The formulation of Mancilla and Storer (2012) uses additional notation presented in Table 2. Note that components of g are indexed differently in this model than in our formulation (1a)–(1k), but this slight abuse of notation allows us to emphasize the relationship between two sets of variables representing idling times in the two models. The formulation of Mancilla and Storer (2012) is as follows:

$$\text{(M) minimize } \frac{1}{N}\sum_{n=1}^{N}\left[\sum_{i=1}^{P}\sum_{p=1}^{P}\lambda_p^w \cdot w_{i,p}^n + \sum_{i=1}^{P}\sum_{p=1}^{P}\lambda_p^g \cdot g_{i,p}^n + \lambda^o \cdot o^n\right] \tag{2a}$$

$$\text{subject to } \sum_{i=1}^{P} x_{i,p} = 1 \qquad \forall p \tag{2b}$$

$$\sum_{p=1}^{P} x_{i,p} = 1 \qquad \forall i \tag{2c}$$

Table 3
Additional notation (Berg et al., 2014).

Indices	
p, p'	indices for procedures, $p, p' = 1, \dots, P+1$
i	index for appointments, $i = 1, \dots, P+1$
Parameters	
$\lambda^w_{p,p'}$	sequence-dependent waiting cost for procedure p' following procedure p
$\lambda^g_{p,p'}$	sequence-dependent cost of idling between procedures p and p'
A^n_p	binary attendance indicator for patient p in scenario n ($A^n_p = 0$ if and only if p is a no-show)
Scenario-independent (first-stage) variables	
$r_{p,p'}$	binary precedence variable; equals 1 if and only if procedure p is followed by procedure p'
y_p	time allotted to procedure p
Scenario-dependent (second-stage) variables	
$w^n_{p,p'}$	sequence-dependent waiting time for procedure p' when preceded by procedure p in scenario n
$g^n_{p,p'}$	sequence-dependent idle time between procedures p and p' in scenario n
e^n	slack variable measuring early completion of the schedule in scenario n

$$t_i - t_{i+1} - \sum_{p=1}^{P} w^n_{i+1,p} + \sum_{p=1}^{P} g^n_{i,p} + \sum_{p=1}^{P} w^n_{i,p} = -\sum_{p=1}^{P} d^n_p \cdot x_{i,p} \qquad \forall (i < P, n) \tag{2d}$$

$$t_P + \sum_{p=1}^{P} w^n_{P,p} - o^n + e^n = -\sum_{p=1}^{P} d^n_p \cdot x_{P,p} + \mathcal{L} \qquad \forall n \tag{2e}$$

$$w^n_{i,p} \le M^i_1 \cdot x_{i,p} \qquad \forall (i,p,n) \tag{2f}$$

$$g^n_{i,p} \le M_2 \cdot x_{i,p} \qquad \forall (i,p,n) \tag{2g}$$

$$(w^n_{i,p}, g^n_{i,p}, o^n, e^n) \ge 0 \qquad \forall (i,p,n) \tag{2h}$$

$$t_i \ge 0 \qquad \forall i \tag{2i}$$

$$x_{i,p} \in \{0,1\} \qquad \forall (i,p) \tag{2j}$$

As described in Mancilla and Storer (2012), the objective function in (2a) is the sample average of the weighted linear combination of the total waiting cost, total idling cost, and overtime cost. Constraints (2b) and (2c) ensure that each procedure is assigned to one appointment, and each appointment is assigned one procedure. Constraints (2d) define, for each scenario, the waiting and idle time for every appointment. Constraints (2e) define overtime in scenario n. Constraints (2f) and (2g) are logical constraints that enforce the relationship between variables $w^n_{i,p}$, $g^n_{i,p}$, and $x_{i,p}$ (here, M^i_1, $i = 1, \dots, P$, and M_2 are sufficiently large constants). Finally, the remaining constraints specify feasible ranges of the decision variables.

It is well known that, in order to strengthen the formulation, the values of "Big-M" constants in constrains such as (2f) and (2g) should be as small as possible without loss of optimality. Mancilla and Storer (2012) recommend setting

$$M^i_1 = \sum_{j=1}^{i-1} \delta_j, \;\; i = 1, \dots, P,$$

where δ_j corresponds to the jth largest value of $\max_{n=1,\dots,N} d^n_r - \min_{n=1,\dots,N} d^n_r$ over $r = 1, \dots, P$, and

$$M_2 = \max_{p=1,\dots,P} \left\{ \max_{n=1,\dots,N} d^n_p - \min_{n=1,\dots,N} d^n_p \right\}.$$

We followed this suggestion in our computational experiments in Section 4.

The formulation of Berg et al. (2014) uses additional notation defined in Table 3, and is as follows:

$$\text{(B) minimize } \frac{1}{N} \sum_{n=1}^{N} \left[\sum_{p=1}^{P+1} \sum_{p'=1}^{P} \lambda^w_{p,p'} \cdot A^n_{p'} w^n_{p,p'} + \sum_{p=1}^{P+1} \sum_{p'=1}^{P+1} \lambda^g_{p,p'} \cdot g^n_{p,p'} + \lambda^o \cdot o^n \right] \tag{3a}$$

$$\text{subject to } \sum_{p'=1}^{P+1} r_{p,p'} \le 1 \qquad \forall p \tag{3b}$$

$$\sum_{p=1}^{P+1} \sum_{p'=1}^{P+1} r_{p,p'} = P \tag{3c}$$

$$x_{i,p} + x_{i+1,p'} - 1 \le r_{p,p'} \qquad \forall (p, p', i \le P) \tag{3d}$$

$$\sum_{i=1}^{P+1} x_{i,p} = 1 \qquad \forall p \tag{3e}$$

$$\sum_{p=1}^{P+1} x_{i,p} = 1 \qquad \forall i \tag{3f}$$

$$\sum_{p=1}^{P+1} r_{p,P+1} = 1 \tag{3g}$$

$$\sum_{p=1}^{P+1} r_{P+1,p} = 0 \tag{3h}$$

$$x_{P+1,P+1} = 1 \tag{3i}$$

$$w^n_{p,p'} \le M^n_1 r_{p,p'} \qquad \forall (p, p', n) \tag{3j}$$

$$g^n_{p,p'} \le M_2 r_{p,p'} \qquad \forall (p, p', n) \tag{3k}$$

$$-\sum_{p'=1}^{P+1} w^n_{p',p} + \sum_{p'=1}^{P+1} w^n_{p,p'} - \sum_{p'=1}^{P+1} g^n_{p,p'} = A^n_p d^n_p - y_p \qquad \forall (p : p \le P, n) \tag{3l}$$

$$\sum_{p=1}^{P+1} \sum_{p'=1}^{P} g^n_{p,p'} - o^n + e^n = \mathcal{L} - \sum_{p=1}^{P+1} A^n_p d^n_p \quad \forall n \tag{3m}$$

$$r_{p,p'}, x_{i,p} \in \{0,1\}, \; y_p \geq 0 \quad \forall (p,p',i) \tag{3n}$$

$$(w^n_{p,p'}, g^n_{p,p'}, o^n, e^n \geq 0) \quad \forall (p,p',n) \tag{3o}$$

As described in Berg et al. (2014), this formulation uses a dummy procedure $P+1$ that has zero duration and is always assigned to the appointment slot $P+1$. The objective function in (3a) is the sample average of the weighted linear combination of the total waiting cost, total idling cost, and overtime cost. Constraints (3b) ensure that each procedure precedes at most one other procedure. Constraints (3c) ensure that every procedure, except for the dummy procedure and the first procedure, is included in exactly two precedence relationships. Constraints (3d) state that a precedence relationship can only exist if that same relationship is defined by the appointment assignment decisions. Constraints (3e) and (3f) require that each procedure is assigned to one appointment, and each appointment is assigned one procedure. Constraints (3g)–(3i) ensure that the dummy procedure will be the last procedure as defined by the binary precedence variables and the appointment slot assignment variables. If procedure p does not precede procedure p', the associated sequence-dependent waiting and idle times will be 0 by constraints (3j) and (3k), where M^n_1 and M_2 are sufficiently large constants. Constraints (3l) calculate the waiting and idle times associated with each procedure based on the waiting time for the preceding procedure. The clinic's overtime is defined by (3m). Finally, the remaining constraints specify feasible ranges of the decision variables. Berg et al. (2014) set $M^n_1 = \sum_{p=1}^{P} d^n_p, n = 1, \ldots, N$, and $M_2 = \mathcal{L}$, which we also used in our computation experiments in Section 4.

In the following discussion, we will refer to formulation (1) proposed in this paper as (S) (for Shehadeh et al.), and to formulations (2) of Mancilla and Storer (2012) and (3) of Berg et al. (2014) as (M) and (B), respectively.

Note that each of the three models has different capabilities in handling various waiting and idling cost structures. Our model (S) can handle situations where the costs are appointment-specific, model (M) can handle situations where the costs are patient-specific, and model (B) can handle situations where the costs depend on the sequence of patients in the schedule.

We also note that the models take different approaches to calculating waiting times and costs in the presence of no-shows: both in model (M) and our model (S), waiting cost is incurred if an appointment runs late, even if the patient assigned to the following appointment does not show (indeed, a no-show patient is treated as a procedure with duration 0), while in model (B) no waiting cost is incurred in this situation.

In the remainder of the paper, we will consider the SOPSP under the following additional assumptions: (i) zero no-show rate (i.e., $A^n_p = 1 \; \forall (p,n)$); (ii) identical waiting costs across appointments and procedures, i.e., $\lambda^w_i = \lambda^w \; \forall i, \; \lambda^w_p = \lambda^w \; \forall p$, and $\lambda^w_{p,p'} = \lambda^w \; \forall (p,p')$; and (iii) identical idling costs across appointments and procedures, i.e., $\lambda^g_i = \lambda^g \; \forall i, \; \lambda^g_p = \lambda^g \; \forall p$, and $\lambda^g_{p,p'} = \lambda^g \; \forall (p,p')$. Under these assumptions, models (S), (M), and (B) are SMILP formulations of the same SOPSP and are, therefore, equivalent. Table 4 presents the respective sizes, in terms of number of variables and constraints, of the three formulations under these assumptions.

4. Computational experiments

In this section, we present computational experiments that explore the size and characteristics of the SOPSP instances that can be solved with the three SMILP formulations presented in Section 3.2. In Section 4.1, we describe the set of the SOPSP instances that we constructed for our experiments, explain how we generated a testbed of sample average approximations (SAAs) for each instance, and discuss other experimental settings. We then present results in Section 4.2, comparing the computational performance of the three formulations.

4.1. Description of experiments

Due to data privacy policies at the collaborating OPC preventing us from using real patient data directly, and in order to study the impact of a variety of problem characteristics on computational performance, we developed a set of diverse SOPSP instances, in part based on prior literature, summarized in Table 5. Each of the 14 instances is characterized by the number of procedures to be scheduled, the types of procedures, and the number of procedures of each type (for example, Instance 1 involves scheduling 4 procedures: two of type A, one of type C, and one of type J). Probability distributions of procedure durations by type are contained in Table 6.

Instances 1–8, 10, and 11 were based on the data set provided as part of the AIMMS-MOPTA 5th Optimization Modeling Competition (http://coral.ise.lehigh.edu/mopta2013/competition). For each procedure type, we used all procedure duration realizations provided in the data set to fit all valid parametric distributions using the open source Matlab function allfitdist (Sheppard, 2012), selecting the distribution with the best combination of the reported Goodness of Fit metrics (e.g., Akaike Information Criterion, Bayesian Information Criterion, Negative of the Log Likelihood). Instance 9 was based on the problem studied by Berg et al. (2014), which includes procedures of two types: colonoscopies (CL) and upper endoscopies (U). Instances 12–14 were based on the problem studied in Deceuninck et al. (2018), where 75% of the patients are newly referred (N) and the remaining 25% are follow-up return (R) patients. Accordingly, we constructed instances with up to 20 procedures, since this is by far the maximum number of patients a single provider can see in a clinic session. In each instance, we set $\mathcal{L}$ equal to the expected total duration of the P procedures, as is done in Mancilla and Storer (2012), Berg et al. (2014), and others.

We considered three different sets of weights for the multi-criteria objective function: (i) $\lambda^w = \lambda^g = \lambda^o$; (ii) $\lambda^w = 1, \;\; \lambda^g = 0, \;\; \lambda^o = 10$; and (iii) $\lambda^w = 1, \;\; \lambda^g = 5, \;\; \lambda^o = 7.5$. For the first set of weights, each of the three objectives is equally important. The second set comes from Berg et al. (2014), where it was motivated by the argument that instances with $\lambda^g \neq 0$ proved to be computationally easier. The third set comes from Deceuninck et al. (2018), where the authors assumed that the overtime cost is 50% higher than the regular idling cost based on the OPC literature and practice (Cayirli et al., 2006; Deceuninck et al., 2018). Note that, with these sets of weights, and assuming zero no-show rate, formulations (S), (M), and (B) are equivalent.

We added symmetry-breaking constraints (see Berg et al., 2014; Denton et al., 2010; Ostrowski, Linderoth, Rossi, & Smriglio, 2011) to all three models, recognizing that the durations of procedures of the same type are identically distributed. In particular, let P_q be the set of procedures of type q, $q = 1, \ldots, Q$. Without loss of generality, we can assume that procedures within each P_q are numbered sequentially. We added the following symmetry-breaking constraints

Table 4
Sizes of formulations of the SOPSP with P procedures and N scenarios.

	(B)	(M)	(S)
# Binary variables	$2P^2+4P+2$	P^2	P^2
# Continuous variables	$P+1+N(2P^2+4P+4)$	$P+N(2P^2+2)$	$P+N(2P+1)$
# First-stage constraints	$P^3+5P^2+11P+10$	P^2+3P	P^2+3P
# Second-stage constraints	$N(4P^2+9P+5)$	$N(4P^2+P+2)$	$5NP$

Table 5
Characteristics of SOPSP instances.

Instance	# of Procedures	# of Types	Procedures to be scheduled (by type)
1	4 procedures	3 types	(2A, 1C, 1J)
2	5 procedures	4 types	(2A, 1G, 1H, 1J)
3	5 procedures	4 types	(1A, 1D, 2G, 1J)
4	6 procedures	5 types	(1A, 1B, 1F, 2G, 1H)
5	7 procedures	5 types	(1C, 1D, 1F, 1H, 3J)
6	7 procedures	6 types	(1A, 1B, 1D, 1E, 2G, 1J)
7	10 procedures	6 types	(3A, 1C, 1D, 1G, 1I, 3J)
8	10 procedures	6 types	(2A, 1B, 1D, 2G, 2I, 2J)
9	10 procedures	2 types	(6CL, 4U)
10	11 procedures	8 types	(2A, 1C, 2E, 1F, 1G, 1H, 2I, 1J)
11	11 procedures	6 types	(2A, 2F, 1G, 2H, 2I, 2J)
12	12 procedures	2 types	(9R, 3N)
13	16 procedures	2 types	(12R, 4N)
14	20 procedures	2 types	(15R, 5N)

Table 6
Distribution information for procedure duration, by type.

Procedure type	Mean	Variance	Distribution
A	9.83	12.08	Lognormal
B	81.46	804.56	Normal
C	59.75	652.69	Lognormal
D	34.53	303.94	Lognormal
E	120.84	2.38e+3	Lognormal
F	47.76	232.06	Lognormal
G	43.94	469.86	Gamma
H	39.90	129.28	Lognormal
I	95.13	2.430e+3	Lognormal
J	19.51	99.36	Lognormal
U	12.05	188.57	Weibull
CL	30.96	58.75	Weibull
R	20.00	256.00	Lognormal
N	30.00	576.00	Lognormal

to all three models:

$$x_{i,p} - \sum_{j>i}^{P} x_{j,p+1} \leq 0 \ \forall i = 1,\dots,P,\ \ \forall p : p, p+1 \in P_q,\ \ q = 1,\dots,Q, \tag{4}$$

indicating that, if procedures p and $p+1$ are of the same type, p is scheduled before $p+1$.

For each of the 14 SOPSP instances and 3 sets of objective function weights, we generated 10 SAAs, for a total of 420 SAA instances, each with $N = 1000$ scenarios. Our choice of the sample size N was motivated by the trade-off between the computational effort required to solve the resulting mixed-integer linear programs (MILPs) and the quality of approximation of the expected value objective of SOPSP by its sample average. On the one hand, the sizes of MILP instances of (S), (M), and (B) increase with N (see Table 4), and their solution times increase as well. As demonstrated in Section 4.2, using formulation (S), we were able to solve all the SAAs associated with the SOPSP instances described in Table 5 with $N = 1000$ in a reasonable time.

On the other hand, optimal solutions of SAA instances with larger values of N are likely to be closer to optimality with respect to the expected value objective of SOPSP. The research literature on sample average approximation methods in stochastic optimization provides theoretical insights as well as guidance for selecting a sample size from this perspective. In particular, the so-called Monte Carlo Optimization procedure can be used to calculate statistical lower and upper bounds on the optimal value of SOPSP based on an optimal solution to its SAA approximation, which in turn provide a statistical estimate of the relative approximation gap between the optimal value of SOPSP and its SAA approximation (see Homem-de Mello & Bayraksan, 2014 and Kleywegt et al., 2002 and references therein for the description of the MCO methodology and other technical details.) Applying the MCO procedure to the formulation (S) with $N = 1000$, we estimated the relative approximation gaps for the SOPSP instances described in Table 5 to range between 0.004% and 0.9%, whereas larger sample sizes resulted in longer solution times without consistent and significant improvements in the relative approximation gaps. Based on the above considerations, we selected $N = 1000$ for our computational experiments.

We represented and solved the 420 SAA instances using the AMPL modeling language and IBM ILOG CPLEX Optimization Studio (version 12.6.2). We used the default settings of the solver since our experiments showed no consistent benefits of any parameter or settings tuning. We imposed a solver time limit of 7200 seconds (2 hours) for each SAA instance. We performed all experiments on an HP workstation running Windows Server 2012 with two 2.10 gigahertz Intel E5-2620-v4 processors, each with 8 cores (16 total) and 128 gigabyte shared RAM.

4.2. Discussion of results

Recall that formulation (1) proposed in this paper is designated by (S), and formulations (2) of Mancilla and Storer (2012) and (3) of Berg et al. (2014) are designated by (M) and (B), respectively. Henceforth, we will assume that constraints (4) are included in each of the models.

Using our proposed model (S), we were able to solve all 420 instances of the SAAs associated with the SOPSP instances described in Table 5 within the imposed time limit of two hours. In fact, solution times of the SAAs that correspond to Instances 1–9, 10 and 11 under the second and third weight sets, and 12–13 were less than 10 minutes (see Table 7 for details). Moreover, solution times of the SAAs that correspond to the largest (in terms of the number of procedures) and the most complex SOPSP instance (which is somewhat less commonly encountered in practice), Instance 14, were less than 25 minutes. These solution times are sufficient for real-world implementation of model (S). Below, we compare the computational performance of model (S) with models (M) and (B).

4.2.1. Comparison with model (B) of Berg et al. (2014)

Using model (B), we were able to solve 160 of the 420 SAA instances to optimality within two hours, namely, all 60 SAAs that correspond to SOPSP Instances 1–6 and the first weight set, and all 100 SAAs that correspond to Instances 1–5 with the second and third weight sets. We present a comparison of solution times of these 160 SAAs by models (S) and (B) in Table 8. Observe that model (B) takes from 6 to 138 times longer than model (S). We attribute the difference in solution times to two primary reasons. First, as shown in Table 4, model (B) has significantly more variables and constraints. As argued by Artigues, Koné, Lopez,

Table 7
Solution times (in seconds) using model (S).

SOPSP	$\lambda^w = \lambda^g = \lambda^o$			$\lambda^w = 1, \lambda^g = 0, \lambda^o = 10$			$\lambda^w = 1, \lambda^g = 5, \lambda^o = 7.5$		
Instance	Min	Avg ± stdv	Max	Min	Avg ± stdv	Max	Min	Avg ± stdv	Max
1	2	3 ± 0.34	3	3	3 ± 1	7	3	3 ± 0.2	7
2	10	13 ± 2	17	8	11 ± 3	17	4	5 ± 0.9	7
3	8	9 ± 0.9	11	5	5 ± 0.4	6	5	6 ± 0.6	7
4	33	41 ± 6	55	21	23 ± 2	26	23	25 ± 2	28
5	53	65 ± 9	77	44	51 ± 6	60	41	49 ± 5	57
6	99	111 ± 7	122	52	58 ± 8	80	57	70 ± 8	79
7	215	276 ± 46	334	153	176 ± 36	276	168	197 ± 28	248
8	237	284 ± 24	310	140	170 ± 29	242	205	226 ± 18	269
9	57	70 ± 8	85	44	55 ± 6	61	46	53 ± 4	58
10	588	769 ± 105	937	178	226 ± 37	293	233	270 ± 33	342
11	660	770 ± 37	987	254	357 ± 61	460	251	326 ± 43	375
12	83	107 ± 12	123	70	78 ± 5	86	100	116 ± 11	130
13	363	466 ± 59	551	242	297 ± 35	349	455	512 ± 55	602
14	862	1218 ± 164	1464	930	1189 ± 193	1500	461	549 ± 76	703

Table 8
Ratios of solution times of models (B) and (S) on SAAs solved by both.

$\lambda^w = \lambda^g = \lambda^o$(a)			$\lambda^w = 1, \lambda^g = 0, \lambda^o = 10$(b)			$\lambda^w = 1, \lambda^g = 5, \lambda^o = 7.5$(b)		
Min	Avg ± stdv	Max	Min	Avg ± stdv	Max	Min	Avg ± stdv	Max
6	31 ± 29	116	4	33 ± 27	107	8	51 ± 35	138

[*a*] SOPSP Instances 1–6, 10 SAA instances each.
[*b*] SOPSP Instances 1–5, 10 SAA instances each.

Table 9
Ratios of optimal objective values of LP relaxations of (S) and (B).

$\lambda^w = \lambda^g = \lambda^o$			$\lambda^w = 1, \lambda^g = 0, \lambda^o = 10$			$\lambda^w = 1, \lambda^g = 5, \lambda^o = 7.5$		
Min	Avg ± stdv	Max	Min	Avg ± stdv	Max	Min	Avg ± stdv	Max
1.95	2.62 ± 0.41	3.48	1.11	1.38 ± 0.26	2.08	1.27	1.64 ± 0.33	2.49

Table 10
Relative MIP gap at termination for SAAs not solved by (B) in two hours.

$\lambda^w = \lambda^g = \lambda^o$(a)			$\lambda^w = 1, \lambda^g = 0, \lambda^o = 10$(b)			$\lambda^w = 1, \lambda^g = 5, \lambda^o = 7.5$(b)		
Min	Avg ± stdv	Max	Min	Avg ± stdv	Max	Min	Avg ± stdv	Max
41%	54 ± 0.08%	70%	19%	34 ± 0.09%	53%	16%	40 ± 0.09%	52%

[*a*] SOPSP Instances 7–12, 10 SAA instances each.
[*b*] SOPSP Instances 6–11, 10 SAA instances each.

and Mongeau (2015), Catanzaro, Gouveia, and Labbé (2015), Fortz, Oliveira, and Requejo (2017), Jünger et al. (2009), Keha, Khowala, and Fowler (2009), Klotz and Newman (2013), Morales-España, Correa-Posada, and Ramos (2016), Pochet and Wolsey (2006), this increase in model size often suggests an increase in solution time for the linear programming (LP) relaxations. Second, as shown in Table 9, for all 420 SAAs, the LP relaxations obtained using model (S) were strictly tighter than using model (B), by a factor of 1.11 to 3.48.

Finally, for the 260 SAAs that were not solved by model (B) in two hours, we report the relative MIP (relMIP) gap, calculated as relMIP gap $= \frac{\text{UB - LB}}{\text{UB}} \times 100\%$, where UB is the best upper bound and LB is the linear programming relaxation-based lower bound obtained at termination after 2 hours. Of the 260 SAAs in question, 180 terminated with a relMIP gap between 16 and 70% (see Table 10 for details), while the remaining 80 SAAs terminated without any feasible MIP solutions (and thus no upper bound).

4.2.2. Comparison with model (M) of Mancilla and Storer (2012)

Using model (M), we solved 340 of the 420 SAAs to optimality within the two hour time limit. We present performance comparisons for these instances in Table 11. Table 12 identifies the SOPSP instances that gave rise to the remaining 80 SAAs.

In exploring the difference in solution times between the two models, we first observe that they have the same first-stage formulation. Furthermore, as we prove in Theorem 1 in Appendix A, the LP relaxations of the two models have the same optimal objective values. In fact, using the same proof techniques, we can show that, given any set of values of variables $x_{i,p}$ $\forall(i, p)$ that satisfy constraints (1b) and (1c) (which are identical to constraints (2b) and (2c)) and $0 \leq x_{i,p} \leq 1$ $\forall(i, p)$, the optimal objective value obtained by optimizing the remaining (continuous) variables will be the same for either model. This suggests that a branch-and-bound algorithm would perform similarly on both models in terms of the number of nodes explored (recognizing that there will be variability due to CPLEX preprocessing and implementation of branch-and-cut instead of a traditional branch-and-bound). The ratios between the number of nodes explored by CPLEX for the two models for the 340 SAAs solved by both are, indeed, on average equal to 1 for each of the weight sets, as reported in Table 11.

Clearly, then, the difference in solution times between models (S) and (M) is primarily due to differences in time spent exploring each node. This is supported further by Table 11 which reports the ratios in the numbers of simplex iterations required to solve each instance using the two models. The number of iterations is typically much larger for model (M), presumably as a result of the significantly larger second-stage formulation (see Table 4).

Finally, for the 80 SAAs that were not solved by model (M) in 2 hours, the relMIP gap at termination was 15% on average, with the maximum of 25%.

Table 11
Comparison of performance of models (M) and (S) on SAAs solved by both: solution time, number of nodes, simplex iterations.

	$\lambda^w = \lambda^g = \lambda^o$			$\lambda^w = 1, \lambda^g = 0, \lambda^o = 10$			$\lambda^w = 1, \lambda^g = 5, \lambda^o = 7.5$		
Ratio	Min	Avg ± stdv	Max	Min	Avg ± stdv	Max	Min	Avg ± stdv	Max
$\frac{\text{(M) sol. time}}{\text{(S) sol. time}}$	1.2	7 ± 4	21	2	13 ± 9	43	1.1	7 ± 5	27
$\frac{\text{(M) nodes}}{\text{(S) nodes}}$	0.5	1 ± 0.2	1.4	0.2	1 ± 0.3	1.9	0.4	1 ± 0.2	1.4
$\frac{\text{(M) iterations}}{\text{(S) iterations}}$	1	11 ± 15	119	1	12 ± 19	133	1	16 ± 22	113

Table 12
Number of SAA instances that were not solved to optimality in the two hours by model (M).

SOPSP Instance #	$\lambda^w =$ $\lambda^g = \lambda^o$	$\lambda^w = 1, \lambda^g =$ $0, \lambda^o = 10$	$\lambda^w = 1, \lambda^g =$ $5, \lambda^o = 7.5$
10	10	5	4
11	10	10	0
13	6	2	3
14	10	10	10

5. Conclusion

In this paper, we presented a new stochastic mixed-integer linear programming model for the Stochastic Outpatient Procedure Scheduling Problem (SOPSP) using a sample-average approximation. This problem considers the perspective of an OPC manager who must schedule the start times for a day's worth of procedures (patients) for a single provider, where each procedure has a known type and a random (non-negative) duration that follows a known probability distribution associated with the procedure type. Given the uncertainty in procedure duration, the goal is to minimize the expectation of a weighted sum of patient waiting time, provider idle time, and clinic overtime. Our model allows for appointment-dependent waiting and idling costs, and treats patient no-shows as procedures with duration 0.

The SOPSP is a basic (yet still challenging) offline single-resource stochastic sequencing and scheduling problem that has been studied in various forms by several previous authors. Therefore, we compared our model with two closely-related models by Mancilla and Storer (2012) and Berg et al. (2014) under assumptions that ensure their equivalence, and analyzed them both empirically and theoretically. Computational results demonstrated where significant improvements in performance could be gained with our proposed model.

In addition to empirical tractability, our modeling approach has the advantage of implementability. Indeed, our proposed model performed well in the computational experiments that were performed using commonly available computer resources, a standard optimization modeling tool, and a commercial MILP solver with default settings — in other words, it did not require development of any specialized algorithms or a time-consuming search for beneficial software parameter settings. This is in contrast to previously-studied models of Mancilla and Storer (2012) and Berg et al. (2014), which were used in conjunction with specially-developed algorithms or heuristics in the original papers, but did not perform as well as our model with straightforward implementation. Implementability in the above sense is necessary for an optimization-based decision support tool to gain wide adoption in OPCs and other healthcare systems that do not have ongoing access to support staff with optimization expertise, and thus is a valuable feature of our proposed model.

We suggest three areas for future research. First, we would like to extend our approach to include additional sources of uncertainty, particularly variability in patient arrival time. In addition, we are interested in studying trade-offs between "day-of" metrics such as provider idling and patient delay and access delays, i.e., the length of time a patient has to wait before a scheduled appointment is available to them. Finally, our model assumes static scheduling, i.e., scheduling of a fixed number of patients whose procedure types are known in advance. We seek to use the results of this research to develop templates and policies for scheduling patients dynamically as they randomly request future appointments.

Acknowledgments

We are grateful for the insights of Prof. Ruiwei Jiang, Dr. Jacob Kurlander, and Dr. Sameer Saini, and for the contributions of many students affiliated with the Center for Healthcare Engineering and Patient Safety (CHEPS). We would like to thank the anonymous referees and the associate editor for their suggestions that allowed us to improve the paper. We also gratefully acknowledge the financial support of CHEPS and the Seth Bonder Foundation. Ms. Shehadeh's efforts were also supported by grant number P30HS024385 from the Agency for Healthcare Research and Quality.

Appendix A. Comparison of linear programming relaxations of models (S) of (1) and (M) of (2)

In this section, we compare the LP relaxations of models (S) of (1) and (M) of (2) under the assumption that waiting and idling costs are identical across appointments and procedures, i.e., that $\lambda^w_i = \lambda^w$ and $\lambda^g_i = \lambda^g$ $\forall i$, and $\lambda^w_p = \lambda^w$ and $\lambda^g_p = \lambda^g$ $\forall p$. Since these two models take the same approach to waiting time and cost calculations in case of patient no-shows (see discussion in Section 3), we allow for no-shows, which would be represented as procedures with duration 0.

Theorem 1. *Suppose $\lambda^w > 0$, and $\lambda^g > 0$ and/or $\lambda^o > 0$. The linear programming relaxations of models (S) of (1) and (M) of (2) are equivalent. In particular, given an optimal solution to the LP relaxation of (S), we can construct a feasible solution to the LP relaxation of (M) with the same objective function value, and vice versa.*

Proof. Suppose $(\hat{x}, \hat{t}, \hat{s}, \hat{g}, \hat{o})$ (with appropriately indexed components) is an optimal solution to the LP relaxation of (S), which is obtained by replacing constraint (1k) with $0 \le \hat{x}_{i,p} \le 1$ $\forall (i,p)$. Below, we construct a feasible solution $(\bar{x}, \bar{t}, \bar{w}, \bar{g}, \bar{o}, \bar{e})$ to the LP relaxation of (M) with the same objective value. (Recall that components of $\hat{g}$ are indexed differently than those of $\bar{g}$.)

- Let $\bar{x} = \hat{x}$ and $\bar{t} = \hat{t}$. Since $\hat{x}$ satisfies constraints (1b) and (1c), and $0 \le \hat{x}_{i,p} \le 1$ $\forall (i,p)$, $\bar{x}$ satisfies (2b) and (2c), and $0 \le \bar{x}_{i,p} \le 1$ $\forall (i,p)$. Similarly, since $\hat{t}$ satisfies (1j) then $\bar{t}$ satisfies (2i). Moreover, if symmetry-breaking constraints (4) are included in both models, they will be satisfied by both $\hat{x}$ and $\bar{x}$.
- Let $\bar{w}^n_{i,p} = (\hat{s}^n_i - \hat{t}_i) \cdot \hat{x}_{i,p}$ $\forall (i,p,n)$. Due to constraints (1d), and since $\hat{x}_{i,p} \ge 0$, $\bar{w}_{i,p} \ge 0$ and thus satisfies constraints (2h). By construction, $\bar{w}_{i,p} = 0$ whenever $\hat{x}_{i,p} = 0$. Moreover, in an optimal solution of the LP relaxation of (S), $\hat{t}$ and $\hat{s}$ will be chosen to ensure that the values of $\hat{s}^n_i - \hat{t}_i$ will not be excessive

for any n as long as $\lambda^w > 0$ (otherwise, one would be able to reduce the waiting component of the cost of the solution). Therefore, constraints (2f) will be satisfied for sufficiently large $M_1^i,\ i = 1, \ldots, P$.

- Let $\bar{g}^n_{i,p} = \hat{g}^n_i \cdot \hat{x}_{i,p}\ \forall (i, p, n)$, which clearly satisfies (2h). By construction, $\bar{g}^n_{i,p} = 0$ whenever $\hat{x}_{i,p} = 0$. Moreover, in an optimal solution of the LP relaxation of (S), $\hat{t}$ and $\hat{s}$ will be chosen to ensure that the values of $\hat{g}^n_i$ will not be excessive for any n as long as $\lambda^w > 0$, or $\lambda^g > 0$ or $\lambda^o > 0$ (otherwise, one will be able to reduce the waiting or idling/overtime component of the cost of the solution). Therefore, constraints (2g) will be satisfied for sufficiently large M_2.
- Let $\bar{o}^n = \hat{o}^n\ \forall n$ (which satisfies (2h)), and define $\bar{e}^n$ to satisfy Eq. (2e) $\forall n$.

It remains to verify that the vector $(\bar{x}, \bar{t}, \bar{w}, \bar{g}, \bar{o}, \bar{e})$ defined above satisfies constraints (2d), and $\bar{e}^n \geq 0\ \forall n$.

First, we derive several helpful algebraic expressions. Given the formulae defining $\bar{w}^n_{i,p}$ and $\bar{g}^n_{i,p}$, we have:

$$\sum_{p=1}^{P} \bar{w}^n_{i,p} = \sum_{p=1}^{P} (\hat{s}^n_i - \hat{t}_i) \cdot \hat{x}_{i,p} = (\hat{s}^n_i - \hat{t}_i) \cdot \sum_{p=1}^{P} \hat{x}_{i,p} = \hat{s}^n_i - \hat{t}_i\ \forall (i, n) \tag{A.1}$$

and

$$\sum_{p=1}^{P} \bar{g}^n_{i,p} = \sum_{p=1}^{P} \hat{g}^n_i \cdot \hat{x}_{i,p} = \hat{g}^n_i \cdot \sum_{p=1}^{P} \hat{x}_{i,p} = \hat{g}^n_i\ \forall (i, p), \tag{A.2}$$

where the last equality, in both cases, is due to (1c). Using (A.1) and (A.2) and the definition of $\bar{t}$, the left-hand side of (2d) can be re-written as

$$\begin{aligned} \hat{t}_i - \hat{t}_{i+1} - (\hat{s}^n_{i+1} - \hat{t}_{i+1}) + \hat{g}^n_i + (\hat{s}^n_i - \hat{t}_i) &= -\hat{s}^n_{i+1} + \hat{g}^n_i + \hat{s}^n_i \\ = -\sum_{p=1}^{P} d^n_p \hat{x}_{i,p} &= -\sum_{p=1}^{P} d^n_p \bar{x}_{i,p}, \end{aligned} \tag{A.3}$$

where the second equality follows from (1f), and the third one – from the definition of $\bar{x}$. This verifies constraints (2d).

Finally, using the definition of $\hat{e}^n$ via (2e) and expression (A.1), we derive:

$$\begin{aligned} \bar{e}^n &= \bar{o}^n + \mathcal{L} - \sum_{p=1}^{P} d^n_p \bar{x}_{P,p} - \bar{t}_P - \sum_{p=1}^{P} \bar{w}^n_{P,P} \\ &= \hat{o}^n + \mathcal{L} - \sum_{p=1}^{P} d^n_p \hat{x}_{P,p} - \hat{s}^n_P \geq 0 \end{aligned}$$

by (1g).

We conclude that $(\bar{x}, \bar{t}, \bar{w}, \bar{g}, \bar{o}, \bar{e})$ defined above is a feasible solution to the LP relaxation of (M), with objective function value

$$\begin{aligned} &\frac{1}{N} \sum_{n=1}^{N} \left[\sum_{i=1}^{P} \sum_{p=1}^{P} \lambda^w \bar{w}^n_{i,p} + \sum_{i=1}^{P} \sum_{p=1}^{P} \lambda^g \bar{g}^n_{i,p} + \lambda^o \bar{o}^n \right] \\ =&\frac{1}{N} \sum_{n=1}^{N} \left[\sum_{i=1}^{P} \sum_{p=1}^{P} \lambda^w (\hat{s}^n_i - \hat{t}_i) \cdot \hat{x}_{i,p} + \sum_{i=1}^{P} \sum_{p=1}^{P} \lambda^g \hat{g}^n_i \cdot \hat{x}_{i,p} + \lambda^o \hat{o}^n \right] \\ =&\frac{1}{N} \sum_{n=1}^{N} \left[\sum_{i=1}^{P} \lambda^w (\hat{s}^n_i - \hat{t}_i) \cdot \sum_{p=1}^{P} \hat{x}_{i,p} + \sum_{i=1}^{P} \lambda^g \hat{g}^n_i \cdot \sum_{p=1}^{P} \hat{x}_{i,p} + \lambda^o \hat{o}^n \right] \\ =&\frac{1}{N} \sum_{n=1}^{N} \left[\sum_{i=1}^{P} \lambda^w (\hat{s}^n_i - \hat{t}_i) + \sum_{i=1}^{P} \lambda^g \hat{g}^n_i + \lambda^o \hat{o}^n \right], \end{aligned}$$

i.e., equal to the optimal value of the LP relaxation of (S).

Conversely, suppose $(\bar{x}, \bar{t}, \bar{w}, \bar{g}, \bar{o}, \bar{e})$ is an optimal solution to the LP relaxation of model (M) of (2). We will construct a feasible solution $(\hat{x}, \hat{t}, \hat{s}, \hat{g}, \hat{o})$ to the LP relaxation of (S) with the same objective value.

- Let $\hat{x} = \bar{x}$, $\hat{t} = \bar{t}$, and $\hat{o} = \bar{o}$, which satisfy constraints (1b), (1c), (1i), (1j), and $0 \leq \hat{x}_{i,p} \leq 1\ \forall (i, p)$. Moreover, if symmetry-breaking constraints (4) are included in both models, they will be satisfied by both $\bar{x}$ and $\hat{x}$.
- Let $\hat{s}^n_i = \sum_{p=1}^{P} \bar{w}^n_{i,p} + \bar{t}_i$ and $\hat{g}^n_i = \sum_{p=1}^{P} \bar{g}^n_{i,p}\ \forall (i, n)$. Due to (2h), $\hat{s}$ and $\hat{g}$ satisfy (1h), and $\hat{s}$ satisfies (1d).

With the above definitions, (1f) and (1e) readily follow from (2d) and (2h), and (1g) follows from (2e) and nonnegativity of $\bar{e}$. Therefore, $(\hat{x}, \hat{t}, \hat{s}, \hat{g}, \hat{o})$ is a feasible solution to the LP relaxation of model (S), with objective function value

$$\begin{aligned} &\frac{1}{N} \sum_{n=1}^{N} \left[\sum_{i=1}^{P} \lambda^w (\hat{s}^n_i - \hat{t}_i) + \sum_{i=1}^{P} \lambda^g \hat{g}^n_i + \lambda^o \hat{o}^n \right] \\ &= \frac{1}{N} \sum_{n=1}^{N} \left[\sum_{i=1}^{P} \sum_{p=1}^{P} \lambda^w \bar{w}^n_{i,p} + \sum_{i=1}^{P} \sum_{p=1}^{P} \lambda^g \bar{g}^n_{i,p} + \lambda^o \bar{o}^n \right], \end{aligned}$$

i.e., equal to the optimal value of the LP relaxation of (M). This complete the proof. □

Similar analysis techniques can be used to show that the linear programming relaxation of model (S) of (1) (and therefore (M) of (2)) is at least as tight as the linear programming relaxation of model (B) of (3) under the additional assumption that the are no patient no-shows, which needs to be made to account for different approaches to waiting time and cost calculations in these models. Moreover, as illustrated in Table 9, linear relaxations of model (S) had larger optimal values, i.e., were tighter, than linear relaxations of model (B) on all test instances in our computational experiments.

References

Ahmadi-Javid, A., Jalali, Z., & Klassen, K. J. (2017). Outpatient appointment systems in healthcare: A review of optimization studies. *European Journal of Operational Research, 258*(1), 3–34.

Antunes, C. H., Alves, M. J., & Clímaco, J. (2016). *Multiobjective linear and integer programming*. Springer.

Artigues, C., Koné, O., Lopez, P., & Mongeau, M. (2015). Mixed-integer linear programming formulations. In C. Schwindt, & J. Zimmermann (Eds.), *Handbook on project management and scheduling vol. 1* (pp. 17–41). Springer.

Begen, M. A., & Queyranne, M. (2011). Appointment scheduling with discrete random durations. *Mathematics of Operations Research, 36*(2), 240–257.

Berg, B. P., Denton, B. T., Erdogan, S. A., Rohleder, T., & Huschka, T. (2014). Optimal booking and scheduling in outpatient procedure centers. *Computers & Operations Research, 50*, 24–37.

Birge, J. R., & Louveaux, F. (2011). *Introduction to stochastic programming*. Springer Science & Business Media.

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

Bosch, P. M. V., & Dietz, D. C. (2000). Minimizing expected waiting in a medical appointment system. *IIE Transactions, 32*(9), 841–848.

Catanzaro, D., Gouveia, L., & Labbé, M. (2015). Improved integer linear programming formulations for the job sequencing and tool switching problem. *European Journal of Operational Research, 244*(3), 766–777.

Cayirli, T., & Veral, E. (2003). Outpatient scheduling in health care: a review of literature. *Production and Operations Management, 12*(4), 519–549.

Cayirli, T., Veral, E., & Rosen, H. (2006). Designing appointment scheduling systems for ambulatory care services. *Health Care Management Science, 9*(1), 47–58.

Cayirli, T., Veral, E., & Rosen, H. (2008). Assessment of patient classification in appointment system design. *Production and Operations Management, 17*(3), 338–353.

Creemers, S., Beliën, J., & Lambrecht, M. (2012a). The optimal allocation of server time slots over different classes of patients. *European Journal of Operational Research, 219*(3), 508–521.

Creemers, S., Colen, P., & Lambrecht, M. (2012b). Evaluation of appointment scheduling rules: a multi-performance measures approach. Available at SSRN. https://doi.org/10.2139/ssrn.2086264.

Deceuninck, M., Fiems, D., & De Vuyst, S. (2018). Outpatient scheduling with unpunctual patients and no-shows. *European Journal of Operational Research, 265*(1), 195–207.

Denton, B., & Gupta, D. (2003). A sequential bounding approach for optimal appointment scheduling. *IIE Transactions, 35*(11), 1003–1016.

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

Denton, B. T., Miller, A. J., Balasubramanian, H. J., & Huschka, T. R. (2010). Optimal allocation of surgery blocks to operating rooms under uncertainty. *Operations Research, 58*(4-part-1), 802–816.

Erdogan, S. A., & Denton, B. (2013). Dynamic appointment scheduling of a stochastic server with uncertain demand. *INFORMS Journal on Computing, 25*(1), 116–132.

Forst, F. G. (1993). Stochastic sequencing on one machine with earliness and tardiness penalties. *Probability in the Engineering and Informational Sciences, 7*(2), 291–300.

Fortz, B., Oliveira, O., & Requejo, C. (2017). Compact mixed integer linear programming models to the minimum weighted tree reconstruction problem. *European Journal of Operational Research, 256*(1), 242–251.

Garey, M. R., Johnson, D. S., & Sethi, R. (1976). The complexity of flowshop and jobshop scheduling. *Mathematics of Operations Research, 1*(2), 117–129.

Ge, D., Wan, G., Wang, Z., & Zhang, J. (2013). A note on appointment scheduling with piecewise linear cost functions. *Mathematics of Operations Research, 39*(4), 1244–1251.

Gupta, D. (2007). Surgical suites' operations management. *Production and Operations Management, 16*(6), 689–700.

Gupta, D., & Denton, B. (2008). Appointment scheduling in health care: Challenges and opportunities. *IIE Transactions, 40*(9), 800–819.

Ho, C.-J., & Lau, H.-S. (1992). Minimizing total cost in scheduling outpatient appointments. *Management Science, 38*(12), 1750–1764.

Homem-de Mello, T., & Bayraksan, G. (2014). Monte Carlo sampling-based methods for stochastic optimization. *Surveys in Operations Research and Management Science, 19*(1), 56–85.

Jansson, B. (1966). Choosing a good appointment system-a study of queues of the type (*D*, *M*, 1). *Operations Research, 14*(2), 292–312.

Jiang, R., Shen, S., & Zhang, Y. (2017). Integer programming approaches for appointment scheduling with random no-shows and service durations. *Operations Research, 65*(6), 1638–1656.

Jünger, M., Liebling, T. M., Naddef, D., Nemhauser, G. L., Pulleyblank, W. R., Reinelt, G., & Wolsey, L. A. (2009). *50 years of integer programming 1958–2008: From the early years to the state-of-the-art*. Springer Science & Business Media.

Keha, A. B., Khowala, K., & Fowler, J. W. (2009). Mixed integer programming formulations for single machine scheduling problems. *Computers & Industrial Engineering, 56*(1), 357–367.

Kim, S., Pasupathy, R., & Henderson, S. G. (2015). A guide to sample average approximation. In M. C. Fu (Ed.), *Handbook of simulation optimization* (pp. 207–243). Springer.

Klassen, K. J., & Rohleder, T. R. (1996). Scheduling outpatient appointments in a dynamic environment. *Journal of Operations Management, 14*(2), 83–101.

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

Klotz, E., & Newman, A. M. (2013). Practical guidelines for solving difficult linear programs. *Surveys in Operations Research and Management Science, 18*(1–2), 1–17.

Lawler, E. L., Lenstra, J. K., Kan, A. H. R., & Shmoys, D. B. (1993). Sequencing and scheduling: Algorithms and complexity. *Handbooks in Operations Research and Management Science, 4*, pp.445–522.

Mak, H.-Y., Rong, Y., & Zhang, J. (2014). Appointment scheduling with limited distributional information. *Management Science, 61*(2), 316–334.

Mak, W.-K., Morton, D. P., & Wood, R. K. (1999). Monte Carlo bounding techniques for determining solution quality in stochastic programs. *Operations Research Letters, 24*(1), 47–56.

Mancilla, C., & Storer, R. (2012). A sample average approximation approach to stochastic appointment sequencing and scheduling. *IIE Transactions, 44*(8), 655–670.

Marler, R. T., & Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization, 26*(6), 369–395.

Mercer, A. (1960). A queueing problem in which the arrival times of the customers are scheduled. *Journal of the Royal Statistical Society. Series B (Methodological), 22*(1), 108–113.

Molina-Pariente, J. M., Hans, E. W., & Framinan, J. M. (2016). A stochastic approach for solving the operating room scheduling problem. *Flexible Services and Manufacturing Journal, 30*(1–2), 1–28.

Morales-España, G., Correa-Posada, C. M., & Ramos, A. (2016). Tight and compact MIP formulation of configuration-based combined-cycle units. *IEEE Transactions on Power Systems, 31*(2), 1350–1359.

Ostrowski, J., Linderoth, J., Rossi, F., & Smriglio, S. (2011). Orbital branching. *Mathematical Programming, 126*(1), 147–178.

Pinedo, M. L. (2016). *Scheduling: theory, algorithms, and systems*. Springer.

Pinto, J. M., & Grossmann, I. E. (1998). Assignment and sequencing models for the scheduling of process systems. *Annals of Operations Research, 81*, 433–466.

Pochet, Y., & Wolsey, L. A. (2006). *Production planning by mixed integer programming*. Springer Science & Business Media.

Robinson, L. W., & Chen, R. R. (2003). Scheduling doctors' appointments: optimal and empirically-based heuristic policies. *IIE Transactions, 35*(3), 295–307.

Rockafellar, R. T., & Wets, R. J.-B. (1991). Scenarios and policy aggregation in optimization under uncertainty. *Mathematics of Operations Research, 16*(1), 119–147.

Rohleder, T. R., & Klassen, K. J. (2000). Using client-variance information to improve dynamic appointment scheduling performance. *Omega, 28*(3), 293–302.

Sabria, F., & Daganzo, C. F. (1989). Approximate expressions for queueing systems with scheduled arrivals and established service order. *Transportation Science, 23*(3), 159–165.

Salzarulo, P. A., Mahar, S., & Modi, S. (2016). Beyond patient classification: Using individual patient characteristics in appointment scheduling. *Production and Operations Management, 25*(6), 1056–1072.

Shapiro, A., & Homem-de Mello, T. (2000). On the rate of convergence of optimal solutions of Monte Carlo approximations of stochastic programs. *SIAM Journal on Optimization, 11*(1), 70–86.

Sheppard, M. (2012). `allfitdist` function. GitHub repository. https://github.com/dcherian/tools/blob/master/misc/allfitdist.m.

Soriano, A. (1966). Comparison of two scheduling systems. *Operations Research, 14*(3), 388–397.

T'kindt, V., & Billaut, J.-C. (2006). *Multicriteria scheduling: Theory, models and algorithms*. Springer Science & Business Media.

Vanden Bosch, P. M., & Dietz, D. C. (2001). Scheduling and sequencing arrivals to an appointment system. *Journal of Service Research, 4*(1), 15–25.

Vissers, J., & Wijngaard, J. (1979). The outpatient appointment system: Design of a simulation study. *European Journal of Operational Research, 3*(6), 459–463.

Wagner, H. M. (1959). An integer linear-programming model for machine scheduling. *Naval Research Logistics, 6*(2), 131–140.

Weiss, E. N. (1990). Models for determining estimated start times and case orderings in hospital operating rooms. *IIE Transactions, 22*(2), 143–150.

Welch, J., & Bailey, N. J. (1952). Appointment systems in hospital outpatient departments. *The Lancet, 259*(6718), 1105–1108.