A Stochastic Programming Model for Service Scheduling with Uncertain Demand: an Application in Open-Access Clinic Scheduling

Yu Fu¹ · Amarnath Banerjee¹

Received: 12 August 2020 / Accepted: 21 July 2021 / Published online: 27 August 2021
© The Author(s), under exclusive licence to Springer Nature Switzerland AG 2021

Abstract

This paper addressed a scheduling problem which handles urgent tasks along with existing schedules. The uncertainties in this problem come from random process of existing schedules and unknown upcoming urgent tasks. To deal with the uncertainties, this paper proposes a stochastic integer programming (SIP) based aggregated online scheduling method. The method is illustrated through a study case from the outpatient clinic block-wise scheduling system which is under a hybrid scheduling policy combining regular far-in-advance policy and the open-access policy. The COVID-19 pandemic brings more challenges for the healthcare system including the fluctuations of service time and increasing urgent requests which this paper is designed for. The schedule framework designed in the method is comprehensive to accommodate various uncertainties in the healthcare service system, such as: no-shows, cancellations and punctuality of patients as well as preference of patients over time slots and physicians.

Keywords Service scheduling · Urgent scheduling · Stochastic integer programming · No-show · Cancellation · Punctuality

1 Introduction

Unscheduled jobs that need prompt services often complicate a scheduling problem. This is a common problem in several industries. For example, the unscheduled repairs of airplanes, automobiles and locomotives; or unscheduled visit of patients/customers; or breaking news for broadcasting industry. These urgent service

✉ Amarnath Banerjee
banerjee@tamu.edu

Yu Fu
logistics.fuyu@gmail.com

¹ Department of Industrial and Systems Engineering, Texas A&M University, College Station, TX, USA

requests may disturb the existing schedules and therefore cause delays in services. A resilient scheduling system is supposed to take these future urgent requirements into consideration and reduce the cost caused by the delay. The delays of services come from not only the unscheduled tasks but also the uncertainties in the scheduled tasks. Examples of the uncertainties in the scheduled services include but not limited to underestimated or overestimated service time for each task, unexpected cancellations, unplanned tardiness and so on. Given the odd features of scheduling, stochastic programming, which handles optimization with randomness, becomes a leverage in providing a sufficient solution for the complex problem. In this paper, we address the application of stochastic programming in healthcare services. In the pandemic of COVID-19, the increase in urgent requests for the primary care clinics ask for a more resilient scheduling decision supporting system. Our work can provide some insight in preparing for and handling urgent uncertainties in healthcare services and can also be applied to similar scheduling problems in other fields.

In outpatient appointment scheduling, the open-access policy which accepts the same-day patient visit requests is widely practiced by clinics. Contrary to the traditional far-in-advance policy which usually has a long gap between the request date and the visit date, the open-access policy provides patients more convenience when they need expeditious healthcare service. The comparison between open-access policy and traditional far-in-advance policy has been discussed by [1–3] and is extensively studied by [4, 5]. Open-access policy shows its strength in improving patient access and satisfaction, as well as reducing patients' no-show rate and in-direct waiting time [2]. This paper discusses a hybrid policy under which a clinic deals with three types of patients. The first type of patients are those who request their appointments before the visit day. The second type of patients schedule their appointment on the visit day. The third type of patients are walk-in patients who go to the clinic without appointments and wait to see the doctor. In the three types of patients, Type 1 patients are scheduled under the traditional far-in-advance scheduling policy. Type 2 and 3 patients are the same-day request patients under open-access policy. Both far-in-advance and same-day requests can be scheduled in a block-based framework designed in this paper, and we focus on handling the assignment of the same-day requests. Among the large number of research studies regarding outpatient clinic scheduling, open-access policy is fairly mentioned but rarely studied intensively under the three-type-patient framework. In most papers addressing open-access policy, either Type 2 requests [4] or Type 3 requests [6] are treated as the only type of the same-day requests. However, the same-day phone-call requests and the walk-in patients have the following differences. First, Type 2 patients might have positive no-show probabilities, but Type 3 patients will have zero no-show probabilities. Second, the First-Come-First-Serve(FCFS) rule is a hard constraint for Type 3 patients but not for Type 2 patients. The differences of Type 2 and Type 3 patients bring more uncertainties to the scheduling problem and make it more difficult. Addressing three types of patients together makes this paper closer to the real scheduling in the open-access clinics and increases the practical value of this this paper.

We assume that the same-day requests are scheduled over a block-based one-day period. This scheduling method divides a clinic day into several time intervals with equal or unequal lengths where each interval is a block [6, 7]. At the beginning of the

day, the clinic already knows the assignment of Type 1 patients in all the blocks. At the beginning of each block, the clinic knows the assignments of same-day requests received in all previous blocks, and how many patients have overflowed from the immediate previous block to the current block. The clinic does not know how many same-day requests will arrive at the current block, or how many assigned patients will make the appointments, or how many patients can be handled in the current block. This incomplete information for the clinic is not deterministic but tractable through prediction or distribution data, and hence, we will handle it in the Stochastic Integer Programming (SIP) model. In our model, the number of Type 1 patients arrive at the clinic and the number of patients complete the service per block are set as random variables. The decision variables are defined to assign the same-day requests received in the current block to the remaining blocks. In order to solve the SIP model, we derive a transformation from SIP to integer programming. To overcome the difficulty of a large-scale sample space, a bound-based sampling method is developed. Distinguished from the traditional one-at-a-time assignment in other clinic scheduling papers, this paper establishes an aggregate assignment method with the SIP model. Besides the above features, this paper also considers patients preferences and first-come-first-serve (FCFS) rule. Practice managers often confront the dilemma of determining the number of traditional far-in-advance patients and the number of patients that they expect to see as same-day appointment requests. Good patient appointment management will help the practice managers find better balance between the two type of appointments. In this paper we not only provide a method for scheduling the same-day requests, but also inspect the impact from number of pre-scheduled type 1 patients on the assignment of the same-day requests through sensitivity analysis.

In the rest of this section, we summarize the contributions of this paper from modeling and methodology perspectives. First, it is novel to introduce stochastic integer programming into the decision of patient number and sequence in blocks. Appointment scheduling research works can be classified into two categories: a) to decide service time per patient, and b) to decide number of patients per block. Most papers adopting stochastic programming (SP) in clinic scheduling fall into the first category [8, 9]. The second category can be found in papers using Markov Chain methods or other dynamic methods based on stochastic model [2, 10, 11]. For category a) problem, the total number of patients assigned to see physicians are set as known parameters which in category b) problem is set as a decision variable. Therefore decision in category b) is a prerequisite of the decision in a). Patients' sequence can be decided in both a) and b); however, most papers in category a) decide time allowance with predefined patient/job sequence. In this paper, we decide number of same-day request patients assigned to each block. The block order and the FCFS rule together also decide the sequence of the patient. This paper targets at deciding the number of patients per block and the patients' sequence which makes it special in category b). It also distinguishes itself from the preceding papers in category b) by using SP model which is more common in category a). We define the "block throughput" as the number of patients that complete their appointments in a block. The block throughput is a concept directly related to patients' service time and is also the cornerstone for overbook in order to offset patients' no-shows. Since the

block throughput directly reflects physicians' initiative in changing their work speed according to their workload and schedule, so compared with service time per patient type, block throughput is more flexible to describe the work efficiency of physicians. Therefore, in comparison with papers using SP, we adopt a new perspective of scheduling. In comparison with papers addressing optimal assignment number, we adopt a new method.

Second, the way we tackle uncertainties in this paper is special. Under the open-access policy, a clinic has incomplete information about the following questions: 1) how many same-day requests will arrive at the current block. 2) How many assigned patients will make the appointments. 3) How many patients can be handled in the current block. For uncertainty in 1), most clinics and research work ignore it since they adopt one-at-a-time assignment, in which the assignment decision is made for each request sequentially and independently. The model we developed can be used for one-at-a-time assignment, however, we propose an aggregated assignment policy to replace the traditional one-at-a-time assignment policy. We suggest to make the assignment decision at the beginning of a block based on estimation of uncertain information in 1). This policy aggregates the requests spreading out over a block and achieve the best allocation plan to this group of requests. For uncertainties in 2) and 3), we treat them as random variables and exploit statistical inference to obtain distribution information before solving the decision model.

Last, we highlight the contributions of this paper by showing its comprehensiveness in including various environment factors in the model. According to review work of [12], so far no existing scheduling model, no matter online [7–9, 13–15] or offline [16, 17] methods, considers more than three of the following factors: patient punctuality, patients' no-show, patient cancellation, random service time (in our case random block throughput), as well as patient preferences over physicians and blocks. The model we provide can handle all of them together. These factors ensure the model a high approximation level to reality.

The rest of the paper is arranged as following: Section 2 is about the comprehensive literature summary about clinic scheduling problem. Section 3 establishes a model framework under which SIP with single physician, SIP with multiple physicians, and robust optimization (RO) models are formulated. Section 4 proposes the solution method for SIP model where a bound-based sampling method is elaborated through which the SIP model and the RO model are compared in special numerical experiments. Section 5 illustrates the aggregate assignment method which is robust in tolerating inaccurate estimation of the same-day requests for the SIP model. Section 6 analyzes the importance of the same-day requests estimation and other model input parameters.

2 Literature Review

The clinic appointment scheduling is a popular topic in OR/IE studies. Since the pioneering study on clinic appointment policy in the 1950's by [18] and [19], ample theories, methodologies and technology have been introduced for the clinic scheduling problem. The paper by [20] provides a comprehensive review on

literature about clinic scheduling before 2003. The study by [21] offers a more recent update on this topic up to 2008. The most recent review study is addressed by [12] who classify the operations research in outpatient appointment scheduling into strategic, tactical and operational levels. In the following three paragraphs, we compare and contrast our work with several similar-topic/similar-method papers in detail.

As for the methods exploited for clinic scheduling, we can find analytic studies based on queuing theory [22, 23], and on top of queuing theory, the Markov-chain model [2, 24–26], simulation modeling [27, 28], dynamic programming [29, 30], approximation/heuristic algorithms [31–35], robust optimization [36–38], and stochastic programming (SP) as adopt in this paper [8, 9, 15, 39, 40]. Specially, it is popular to combine two or more methods mentioned above together as stochastic analytic method [2, 7, 10, 11, 13, 14, 41]. In this paper, we proposed a framework where the stochastic integer programming is the base which contains the robust optimization as a special case. Its variants can handle ample stochastic factors such as no-shows, early/late arrival, cancellation, preferences over physicians, and so on.

An important classification of scheduling is the online/offline scheduling. Online scheduling is addressed in this paper. For research focusing on appointment assignment optimization, if the decision for all patients is made before the clinic day then the scheduling is considered static or offline [16, 17]. The offline scheduling is associated with decisions made with full information about number of patients to be scheduled and their service time. In contrast, if the decision is made one by one or ahead of each block, then the scheduling is considered dynamic or online [7–9, 13–15]. This paper deals with online scheduling which has the advantage to handle the scheduling facing unknown future.

From the view of objectives of scheduling, research papers aim at reducing cost, increasing revenue or the combination of the two. The cost consists of patients' waiting time, doctors' idle time, overtime, and so on [20]. The revenue is usually calculated by number of patients served or scheduled [2, 42]. The combination of revenue and cost can be either cost-savings as the difference between revenue and cost [7, 13, 15], or average cost which is cost per served patient [14]. In this paper, we adopt the cost-saving objective function which considers possible revenue of appointments, the waiting-time costs and the idle-time costs associated with patient shortage as well as the overtime cost for the number of overflowed patients. This comprehensive objective function makes the model to draw a balance between "earning more revenue" and "respecting patients' and "physicians' feeling".

For the patient arrival mode, literature for the optimal scheduling can be classified as non no-show arrival [40], arrival with no-show [9, 15], and arrival with no-show and cancellations [2]. This work takes no-show rates as an essential factor influencing the clinic scheduling decision, and also discusses the cancellation and lateness of patients. As for the patient choice scope, in the literature, there are patients' preferences over blocks in a day [42–44], or over days [10]. This paper deals with the same-day requests, so patients' choices are circumscribed in the current day. For the patients' preferences over doctors, most papers we discussed here are dealing with one single physician except for [27, 33, 45] where simulation and multi-server queuing model are used. The scheduling model we proposed is very

flexible be extended from single physician to multiple physicians, multiple departments or even multiple clinics as illustrated in Sect. 3.3.

For the papers which discuss open-access policy and the same-day requests in clinics, [29] discuss the same-day requests scheduling through dynamic programming method. They take the preference of patients on physicians into consideration and address the balance problem between pre-scheduled appointments and same-day requests. Nevertheless, the uncertainties of patients attendance, physician work efficiency, no-shows as well as services time are not considered in their model. They also ignore the differences between walk-ins and same-day phone call requests. [14] assume three types of patients which we adopt in this paper. They work out comprehensive stochastic formulations to depict the assignment constraints like FCFS rules, no-shows, cancellations, overtime, idle time, starting time, and waiting time. However, the subtle considerations and some nonlinear and stochastic constraints make it far from a solvable stochastic programming model. They use discrete-event simulation to determine some parameters which we suppose in our model as random variable. Genetic algorithm is used in their paper to pursue local optimal allocations for Type 2 and Type 3 patients with block capacity up to 2, and best arrangement for Type 1 patients. Their block capacity is a similar notion as block throughput in our paper, but they set it as a constant parameter. The framework we proposed in this paper can handle as many uncertain factors as in [14]. Our advantage is that, under this framework, it is possible to have exact solutions using Bender's decomposition which we will address in another paper.

For the papers which help to decide optimal patient numbers, stochastic analytic method are widely used. [7] as well as [13] work out a sequential assignment method for multiple type of patients to clinic time blocks. For each arrived request, their algorithm assigns it to one of the blocks by trying each block one by one from the current block to the final block to find the one with the lowest average cost. The cost is calculated based on the distributions of number of arriving patients at the beginning of a block and the number of overflows among blocks which have been formulated. [15] present a very similar work to [13] with improvement in applying the method to multiple resources and calculation overflows using convolution estimation method and joint cumulative estimation method. The differences between our paper and those above are quite perceptible. First of all, they assign only one patient at a time, as the one-at-a-time mode. We propose an aggregate assignment which dominates the one-at-a-time method in robustness and tolerance on input inaccuracy. Second, their assignment method is exhaustive and based on the first order statistic, i.e., the expected value of random variables. We introduce a two-stage SIP model to handle the multiple assignment with uncertainty, which reveals more impacts of random variables on the assignment decision and is more closer to the reality.

For the SP models, [8] propose a two-stage SP model based sequential bounding approach to obtain the optimal appointment scheduling for a single server system. Their model is built on the basis of earlier research by [26] and [17]. To facilitate solving the model fast and effectively, they developed aggregation bounds for the recourse function and bounds for dual multipliers in a block. [40] also use a similar model as [17], but rather than solving it as a two-stage model, they take the

approximation model as a linear model and then solve it with conjugate gradient search. [9] extend this approach in [8] to clinic appointment. They develop a multi-stage stochastic model on the basis of a two-stage model and utilize nested decomposition algorithm and customized cuts to achieve optimal solution. The SP model proposed by [39] decides both time allowance and sequence of the patients. They use L-shaped methods and progressive hedging to solve the SP model. All these SP models above have continuous decision variables, where the model we have is stochastic integer programming (SIP) which increases the computational complexity significantly. As mentioned before, we use SIP to optimize number of assigned patients which traditionally is decided by queuing theory-based methods. This is a novel application of SIP on scheduling.

Robust optimization and SP are the common tools to achieve the best decision of time allowance and sequence for predefined set of patients given their random service time. A typical difference between robust optimization and stochastic optimization is that the latter one allows various distributions of the random variables which in former one very limited [36–38]. However, our paper leverages SP to provide the answer to how many of the randomly arrived walk-ins and phone-call requests can be assigned in the remaining blocks. We also propose a robust optimization model for the same problem and compare the performance of the two models through a few numeric examples.

The uncertainty in data and integer decisions increase the complexity of calculation for SP model. The solution method for SP model includes the exact L-shaped method [39], sample average approximation (SAA) [46, 47] and heuristics [40]. This paper uses the sample average approximation method. The large sample space of the random variable in a SP prevents people to exhaust every possible scenario in the distribution, instead, the sampling method is widely used to shrink the number of scenarios during the calculation [48]. According to [47], there are two types of sampling methods: the exterior sampling method and the interior sampling method. The exterior sampling method deals with the deterministic approximation model of the problem generated after the sampling. For the question how to decide the sample size N for exterior sampling method, [49] discuss in detail the convergence and lower bound for the sample size N . [50] also present the formula of lower bound of it. Based on [47], the SAA which approximates the expectation function by a sample average function is an example of using exterior sampling method. The interior sampling method distinguishes from the exterior sampling method from when to do the sampling. The exterior sampling method generates samples before solving the problem, while in interior sampling method, the samples are drawn during the course of solving the SP problem [51–53]. For the interior sampling methods, [53] suggest that one can increase the number of scenarios by one at each iteration. For the n th step, there will be n samples drawn. When n gets large enough, one can obtain the average value of solutions as an approximation of optimal solution. [51, 52] develop a stochastic decomposition method which generates only one new sample at each iteration. Interior sampling method is flexible to be applied together with decomposition method to solve SP with more complicated features such as integer recourse [50] or nonlinear constraints [46] or multistage decisions [54]. In this paper, we develop a new interior sampling method for SIP based on bounds

of objective values. Features that distinguishes our method from the existing interior methods include: 1) the simple structure of the model decides that our method is more about reducing the sample size than serving a partial rule in solving the model. For example, the interior sampling method from [46] is combined with policies dealing with nonlinear constraints. The interior sampling method from [50] is embedded in decomposition. Therefore, our method can be used as a prototype for sampling and applied with other algorithms in these complicated models. 2) Compared with other pure interior sampling methods in solving SIP, our method shows its strength in reducing the sample size which will be addressed later.

Conclusively, this paper demonstrates its strength and creativity in both the modeling and solution method parts. For the modeling part, it deals with dynamic scheduling for the same-day-request patients with no-shows and various restrictions, the comprehensiveness of the model is not found in previous literature. Especially, the aggregate assignment is novel among the online scheduling methods. For the solution method part, the bound-based sampling method designed here makes the two-stage SIP model produce reasonable solutions easily and fast.

3 Problem Statement and Formulation

3.1 Assumptions

For the block-wise scheduling, the analysis and decisions are carried out within each block. The benefit of the block-wise scheduling is that the decision maker can aggregate information and resources of one block and make decisions accordingly. The block containing the current clock time is referred as the “current block”. The blocks lying earlier than the current block on the time line are the “previous blocks”. The current block and the following blocks are the “remaining blocks”. For the convenience of calculation and demonstration, we assume that the length of blocks are equal. However, our analysis and method can also be applied to the case of unequal length blocks. This paper approximates reality from both event sequence and availability of information. We assume that the event sequence in one block is: Step 0: At the beginning of the day, the clinic observes the number of Type 1 patients assigned for each block. Step 1: At the beginning of the current block, the clinic observes the number of Type 2 and Type 3 patients assigned today before the current block. Step 2: The clinic estimates Type 2 and Type 3 requests received at this block. Step 3: The clinic makes an assignment plan for the estimated Type 2 and Type 3 requests. Step 4: The clinic starts to receive Type 2 and Type 3 requests and assign them one by one following the decision in Step 3. Step 5: The clinic starts to observe the arrivals of Type 1 and Type 2 patients in the current block. Step 6: The service for patients in the current block starts until the end of the current block. Then the successive block starts a new process from Step 1.

Based on this event sequence, we build the two-stage SIP model for Step 3 above. The first stage of SIP model tackles assignment decision with deterministic data obtained in Step 1 and Step 2 above. The second stage estimates the consequence of the assignment decision under random scenarios and gives feedback in the follow

Table 1 Sets and Indices

J	set of remaining blocks
K	set of Type 2 patients whose requests are received in the current block
T	set of Type 3 patients that arrive at the clinic in the current block
i	index of the current block
j	index of remaining blocks
n	index of scenarios
k	index of Type 2 patients whose requests are received in the current block
t	index of Type 3 patients that arrive at the clinic in the current block

way: If the number of patients that can be served in a block exceeds the number of patients that assigned to a block, then an idle-time cost associated with patient shortage is produced; if the number of assigned patients exceeds the number of patients that can be served in a block, then an overflow cost associated with patient waiting time is generated. The actual overflows will be added to the number of patients to be served in the immediate successive block in Step 1 above. More assumptions of this model are: (1) The clinic can reject requests of Type 2 and Type 3 patients. (2) There is a delay between arrival time of Type 2 patient requests and their scheduled block. (3) There is a waiting length tolerance for Type 3 patients. (4) Type 1 patients have no-show rates, all Type 2 patients will make the visits. (5) Assignment of Type 3 patients follows a FCFS rule. (6) We know the probability distribution of uncertain data (discrete, i.i.d.). (7) Waiting time cost is generated when patients overflow from original block into the successive block. (8) All patients who make the visits will come on time. (9) All blocks have equal length. Assumption (4) and (8) make the uncertainties in the second-stage model exogenous which means they are independent of the decision variables in the first-stage model. If we assume that Type 2 patients have some no-show rates or earliness/lateness possibilities, then we have endogenous uncertainties which depend on the decision variables in the

Table 2 Parameters

m	number of blocks per clinic day
A_{jk}	preference factor of k^{th} Type 2 patient for j^{th} remaining block
B_{jt}	preference factor of t^{th} Type 3 patient for j^{th} remaining block
r_j	number of Type 1 patients assigned to j^{th} remaining block
ρ^j	the largest index of block which the Type 3 patients requested in block j are assigned to
c_2	cost of rejecting one Type 2 patient
c_3	cost of rejecting one Type 3 patient
$\bar{a}_j$	number of Type 2 patients that have been assigned to block j before the current block
$\bar{b}_j$	number of Type 3 patients that have been assigned to block j before the current block
$\hat{q}$	number of patients overflow from previous assigned block to the current block
c_f	cost of one patient overflow from one block to next block
c_s	cost of one patient shortage in one block
c_o	cost of seeing one patient after the close time of clinic

Table 3 First-Stage Decision Variables

x_{jk}	binary variable showing whether the k^{th} Type 2 patient is assigned to block j
y_{jt}	binary variable showing whether the t^{th} Type 3 patient is assigned to block j
z_t	binary variable showing whether the t^{th} Type 3 patient is assigned or rejected
a_j	integer variable showing number of Type 2 patients that have been assigned to block j after the current block
b_j	integer variable showing number of Type 3 patients that have been assigned to block j after the current block

first-stage model. Solution methods for the SIP models with exogenous uncertainties are different from those for the models with endogenous uncertainties. For SIP with exogenous uncertainties in this paper, we adopt sample average approximation (SAA) method associated with the deterministic equivalent problem (DEP) which are not applicable for SIP with endogenous uncertainties. For SIP with endogenous uncertainties, we applied decomposition methods in another paper. Assumption (2), (3) and (5) show the major differences of Type 2 and Type 3 patients in this paper. To be more specific, for Type 2 patients, the assignment model shall consider the time gap between request time and service start time to give them enough time to travel to the clinic. For Type 3 patients, the assignment model should not let them wait too long before service starts. We will address these two aspects in the assignment restriction constraints. The FCFS rule of assignment is only for Type 3 patients which guarantees that the walk-in patients who arrive earlier will get service earlier. For Type 2 patients, it is possible that patients call earlier get service later than patients call later.

3.2 Two-stage Integer Stochastic Programming Model for One Block with One Physician

3.2.1 Notation

This model will use indices and parameters listed in Tables 1, 2, 3, 4, and 5.

Table 4 Second-Stage Random Variables

$\hat{\omega}$	random variable of second stage
ω	outcome of the random variable
$\tau_j(\omega)$	number of patients can be served in j^{th} remaining block
$v_j(\omega)$	number of type 1 patients assigned to j^{th} remaining block that will make the appointment
$\eta_j(\omega)$	difference between the number of type 1 patients who arrived for the service in block j and the number of patients can be served in this block

Table 5 Second-Stage Decision Variables

q_j	integer variable for number of patients overflowing from block $j - 1$ to block j
g_j	integer variable for number of patients that can be handled but are not assigned to block j (patient shortage or surplus capacity in block j)

3.2.2 Formulations

Suppose the current block is the i th block of the clinic day. The SIP-i model below is the two-stage SIP decision model for block i . All the decision variables associated with the remaining blocks are indexed from 1 to h , and their original indices are from i to $i + h$. Function and constraints (1) to (9) belong to the first stage. Constraints (11) to (13) go with the second-stage model.

$$\text{(SIP-i)} \quad \min \quad -c_2 \sum_{j \in J} \sum_{k \in K} x_{jk} - c_3 \sum_{j \in J} \sum_{t \in T} y_{jt} + \mathbb{E}[Q(X, Y, \mathbf{a}, \mathbf{b}, \omega)] \quad (1)$$

$$\text{s.t.} \quad \sum_{j \in J} x_{jk} \leq 1, \quad \forall k \in K \quad (2)$$

$$\sum_{j \in J} y_{jt} \leq 1, \quad \forall t \in T \quad (3)$$

$$x_{jk} \leq A_{jk}, \quad j \in J, k \in K \quad (4)$$

$$y_{jt} \leq B_{jt} z_t, \quad j \in J, t \in T \quad (5)$$

$$a_j = \bar{a}_j + \sum_{k \in K} x_{jk}, \quad j \in J \quad (6)$$

$$b_j = \bar{b}_j + \sum_{t \in T} y_{jt}, \quad j \in J \quad (7)$$

$$\sum_{t \in T} (i + j) y_{jt} \geq \beta^{i-1} \sum_{t \in T} z_t, \quad j \in J, \quad t \in T \quad (8)$$

$$x_{jk}, y_{jt} \in \{0, 1\}, \quad a_j, b_j \in \mathbb{Z}^+, \quad j \in J, \quad k \in K, \quad t \in T \quad (9)$$

where

$$Q(X, Y, \mathbf{a}, \mathbf{b}, \omega) = \min \quad c_f \sum_{j \in J} q_j + c_s \sum_{j \in J} g_j \quad (10)$$

$$\text{s.t.} \quad q_{j+1} - q_j - g_j = \eta(\omega)_j + a_j + b_j, \quad j \in J \quad (11)$$

$$q_1 = \dot{q} \quad (12)$$

$$q_j, g_j \in \mathbb{Z}^+, \quad j \in J \quad (13)$$

The objective function in (1) is designed to minimize unassigned same-day requests received in the current block and minimize the overflow and patient shortage costs of the remaining blocks. The full formulation of the objective function is $c_2(|K| - \sum_{j \in J} \sum_{k \in K} x_{jk}) + c_3(|T| - \sum_{j \in J} \sum_{t \in T} y_{jt}) + \mathbb{E}[Q(X, Y, \mathbf{a}, \mathbf{b}, \omega)]$, where the first term is the cost of unassigned Type 2 patients, the second term is the cost of unassigned Type 3 costs, and the third term is the expectation of the second-stage objective function value. Since $c_2|K|$ and $c_3|T|$ are constant, we derived the simplified objective function in (1).

Constraints (2), (3) are about the decisions of accepting or rejecting patient requests. Constraints (4), (5) are about patient assignment restrictions. Preference matrix A subjects to both the preference and restriction for Type 2 patients. The restriction part can be defined using a series of arrival delay factor $\rho = \{\rho_k\}$ associated with k th patient as shown in (14). The delay factor indicates the time length in terms of number of blocks the patient between the request time and their arrival time. So any block beyond the arrival delay $i + \rho_k$ can be chosen to serve the patient. In a similar way, matrix B consists of binary entries based on preference and waiting tolerance of Type 3 patients. Each entry of B is associated with the tolerance factor δ_t which shows the max number of blocks the t th Type 3 patient can wait in the clinic.

$$A_{jk} = \begin{cases} 1 & \text{if } j + i \in [\min\{i + \rho_k, m\}, m] \quad \& \\ & \text{patient } k \text{ is willing to make the appointment in block } j + i \\ 0 & \text{otherwise} \end{cases} \quad (14)$$

$$B_{jt} = \begin{cases} 1 & \text{if } j + i \in [j + i, \min\{i + \delta_t, m\}] \quad \& \\ & \text{patient } t \text{ is willing to make the appointment in block } j + i \\ 0 & \text{otherwise} \end{cases} \quad (15)$$

Constraints (6), (7) are designed to store the accumulated number of patients assigned to each of the remaining blocks by the end of current block. Variables $\mathbf{a}, \mathbf{b}$ appear in both first-stage model and second-stage model. Constraint (8) is about the FCFS rule for Type 3 patients. We do not apply this rule to Type 2 patients because of their delay restrictions. The element $(i + j)y_{jt}$ on the left-hand side gives the index of block where the t th Type 3 patient is assigned to, β^{i-1} is the largest index of assigned block for Type 3 patients who arrived in the previous block. This constraint guarantees that if the t th Type 3 patient's request is accepted, then the block assigned to this patient cannot be earlier than the last assigned block for Type 3 patients who came earlier than the current block. This constraint goes with the assumption that Type 3 patients arrived in the same block do not obey the FCFS rule.

Second-stage objective function (10) is established to evaluate the cost of overflows and patient shortage for the remaining blocks as the consequence of assigning decision in the first-stage. Constraint (11) is derived from input/output balance of

each remaining blocks as a consequence of decisions made by the end of the current block. For block j , the input number of patients includes v_j, q_j, a_j, b_j , the number of patients that can be served (output) in it is τ_j . Especially, for the first block, $\hat{q} = 0$. If the input number is larger than the served number, then a positive number of patients q_{j+1} will overflow to the next block, which means:

$$q_{j+1} = \max\{v_j + q_j + a_j + b_j - \tau_j, 0\} \quad (16)$$

If the input number is smaller than the served number, then it generates a positive patient shortage g_j which means:

$$g_j = \max\{\tau_j - v_j - q_j - a_j - b_j, 0\} \quad (17)$$

Given the definition of η_j , i.e.,

$$\eta_j := v_j - \tau_j \quad (18)$$

These equations lead to constraint (11). From the second-stage model, we can see that the SIP-i model has relatively complete recourse since for any $\boldsymbol{\eta}(\omega) + \mathbf{a} + \mathbf{b} \in \mathbb{Z}^{|J|}$, we can always find $\mathbf{q}, \mathbf{g}$ such that (11) to (13) are satisfied. The relatively complete recourse feature is import for sample average approximation (SAA). In reality, distributions of the random variables of v_j and τ_j can be detected by statistics inference methods from the historic data of the clinic. In this paper, we make assumptions for the distributions based on literature and historic data from a small clinic that is affiliated to our institution's health science center as elaborated in Sect. 4. SIP-i as a framework can be extended to consider patients' no-shows, cancellations and punctuality as described in [Appendix](#).

3.3 Two-Stage Integer Stochastic Programming Model for Multiple Physicians

Here we extend the SIP-i model for a clinic with multiple physicians. In addition to Table 1, we define set P as the set of physicians and p as the index of physicians in set P . Some of the parameters in Table 2 and decision variables in Table 3 get one additional dimension as shown in Table 6. Under the multiple physician scenario, the patients may have preference over different physicians. If patient k prefers not to see physician p , we have $A_{jkp} = 0 \forall j$. By introducing more physicians into the system, we expect less overflows among blocks since physicians work parallel and therefore block throughput is increased. If we take patient category or different clinics into consideration, we can have one dimensions on top of the model we proposed here.

The SIP-i-multi is established based on SIP-i in order to accommodate the physician settings. In this model, we assume that different physicians have different service productivity, but the cost of overflows or idle time are the same for all the physicians. However, SIP-i-multi still has the same structure as SIP-i, and therefore, in the rest of this paper, we focus on SIP-i for solution method, experiments, and sensitivity analysis.

Table 6 Additional Parameters And Decision Variables

A_{jkp}	parameter	preference factor of k^{th} Type 2 patient for j^{th} remaining block and p^{th} physician
B_{jlp}	parameter	preference factor of l^{th} Type 3 patient for j^{th} remaining block and p^{th} physician
$\tilde{a}_{jlp}$	parameter	number of Type 2 patients that have been assigned to block j and physician p before the current block
$\tilde{b}_{jlp}$	parameter	number of Type 3 patients that have been assigned to block j and physician p before the current block
x_{jkp}	variable	whether the k^{th} Type 2 patient is assigned to block j and physician p
y_{jlp}	variable	whether the l^{th} Type 3 patient is assigned to block j and physician p
a_{jp}	variable	number of Type 2 patients that assigned to block j and physician p in the current block
b_{jp}	variable	number of Type 3 patients that assigned to block j and physician p in the current block
$\tau_{jp}(\omega)$	random	number of patients can be served in block j by physician p
$v_{jp}(\omega)$	random	number of type 1 patients assigned to block j and physician p that will make the appointment
$\eta_{jp}(\omega)$	random	difference between the number of type 1 patients who arrived for the service in block j with physician p and the number of patients can be served in this block by this physician

$$\text{(SIP-i-multi)} \quad \min \quad -c_2 \sum_{j \in J} \sum_{k \in K} \sum_{p \in P} x_{jkp} - c_3 \sum_{j \in J} \sum_{t \in T} \sum_{p \in P} y_{jtp} + \mathbb{E}[Q(X, Y, \mathbf{a}, \mathbf{b}, \omega)] \quad (19)$$

$$\text{s.t.} \quad \sum_{j \in J} \sum_{p \in P} x_{jkp} \leq 1, \quad \forall k \in K \quad (20)$$

$$\sum_{j \in J} \sum_{p \in P} y_{jtp} \leq 1, \quad \forall t \in T \quad (21)$$

$$x_{jkp} \leq A_{jkp}, \quad j \in J, k \in K, p \in P \quad (22)$$

$$y_{jtp} \leq B_{jtp} z_t, \quad j \in J, t \in T, p \in P \quad (23)$$

$$a_{jp} = \bar{a}_{jp} + \sum_{k \in K} x_{jkp}, \quad j \in J, p \in P \quad (24)$$

$$b_{jp} = \bar{b}_{jp} + \sum_{t \in T} y_{jtp}, \quad j \in J, p \in P \quad (25)$$

$$\sum_{t \in T} \sum_{p \in P} (i+j)y_{jtp} \geq \beta^{i-1} \sum_{t \in T} z_t, \quad j \in J, \quad t \in T \quad (26)$$

$$x_{2jkp}, y_{jtp} \in \{0, 1\}, \quad a_{jp}, b_{jp} \in \mathbb{Z}^+, \quad j \in J, \quad k \in K, \quad t \in T, \quad p \in P \quad (27)$$

where

$$Q(X, Y, \mathbf{a}, \mathbf{b}, \omega) = \min \quad c_f \sum_{j \in J} q_j + c_s \sum_{j \in J} g_j \quad (28)$$

$$\text{s.t.} \quad q_{j+1} - q_j - g_j = \sum_{p \in P} (\eta(\omega)_{jp} + a_{jp} + b_{jp}), \quad j \in J \quad (29)$$

$$q_1 = \hat{q} \quad (30)$$

$$q_j, g_j \in \mathbb{Z}^+, \quad j \in J \quad (31)$$

3.4 Robust Optimization Model for One Block with One Physician

In this section, we proposal a robust optimization model (RO-i) under the framework of SIP-i. The differences between SIP-i and RO-i come from 1) in the objective function part, SIP-i uses the expectation of overflow and idle time costs based on the scenarios of the random variables, but RO-i directly use the cost value. 2) for the random variables v_j and τ_j , RO-i has constraints in (41) and (42) to define the scope of the random variables, but SIP-i uses the distributions of them. 3) overall speaking, RO-i provides the most cost-efficient assignment of Type 2 and Type 3

patients who send their requests in the current block for each of the remaining blocks given the fluctuation of v_j in $[\underline{v}_j, \bar{v}_j]$ and the fluctuation of τ_j in $[\underline{\tau}_j, \bar{\tau}_j]$. SIP-i gives the best assignment of Type 2 and Type 3 patients who send their requests in the current block for each of the remaining blocks considering the expected impact of overflow and idle time on the costs. RO-i is comparable to the special case of SIP-i when v_j and τ_j both have uniform distribution. We did some numeric experiment on comparing the two in next section which shows that when the random variables follow uniform distribution, RO-i and SIP-i tend to make the same decision.

$$\text{(RO-i)} \quad \min \quad -c_2 \sum_{j \in J} \sum_{k \in K} x_{jk} - c_3 \sum_{j \in J} \sum_{t \in T} y_{jt} + c_f \sum_{j \in J} q_j + c_s \sum_{j \in J} g_j \quad (32)$$

$$\text{s.t.} \quad \sum_{j \in J} x_{jk} \leq 1, \quad \forall k \in K \quad (33)$$

$$\sum_{j \in J} y_{jt} \leq 1, \quad \forall t \in T \quad (34)$$

$$x_{jk} \leq A_{jk}, \quad j \in J, k \in K \quad (35)$$

$$y_{jt} \leq B_{jt} z_t, \quad j \in J, t \in T \quad (36)$$

$$a_j = \bar{a}_j + \sum_{k \in K} x_{jk}, \quad j \in J \quad (37)$$

$$b_j = \bar{b}_j + \sum_{t \in T} y_{jt}, \quad j \in J \quad (38)$$

$$\sum_{t \in T} (i+j)y_{jt} \geq \beta^{i-1} \sum_{t \in T} z_t, \quad j \in J, \quad t \in T \quad (39)$$

$$q_{j+1} - q_j - g_j = v_j - \tau_j + a_j + b_j, \quad j \in J \quad (40)$$

$$\underline{v}_j \leq v_j \leq \bar{v}_j, \quad j \in J \quad (41)$$

$$\underline{\tau}_j \leq \tau_j \leq \bar{\tau}_j, \quad j \in J \quad (42)$$

$$x_{jk}, y_{jt} \in \{0, 1\}, \quad a_j, b_j \in \mathbb{Z}^+, \quad q_j, g_j \in \mathbb{Z}^+, \quad j \in J, \quad k \in K, \quad t \in T \quad (43)$$

4 Solve The SIP-i Model

4.1 Deterministic Equivalent Problem

As we mentioned before, solution method for a SP model includes decomposition, sample-based approximation as well as heuristics. In this paper, we focus on solving the sample-based approximation of SIP-i. For a SP model with any discrete distributed random variable, we can derive the deterministic equivalent problem (DEP). The DEP is obtained by associating each second-stage variable with all scenarios of the random variable.

For the SIP-i model, assume that there are N scenarios for random variable ω , the n^{th} scenario n has probability p_n . Then change the second-stage decision variables $\mathbf{q}, \mathbf{g}$ into two-dimensional matrices, i.e. q_j^n denotes overflow to block j under scenario n . The DEP model of SIP-i is presented in (44) to (46).

$$(DEP-i) \quad \min \quad -c_2 \sum_{j \in J} \sum_{k \in K} x_{jk} - c_3 \sum_{j=1}^h \sum_{t \in T} y_{jt} - \sum_{n=1}^N p_n \left(c_f \sum_{j=1}^m q_j^n + c_s \sum_{j=1}^m g_j^n \right) \quad (44)$$

constraints (2) to (9)

$$\text{s.t.} \quad q_{j+1}^n - q_j^n - g_j^n = \eta(\omega)_j^n + b_j, \quad j \in J, n = 1, \dots, N \quad (45)$$

$$q_1^n = \dot{q}, \quad n = 1, \dots, N \quad (46)$$

$$q_{j+1}^n, g_j^n \in \mathbb{Z}^+, \quad j \in J, n = 1, \dots, N \quad (47)$$

DEP-i is an integer programming problem, which can be solved using CPLEX for small N . For large size of N , it is impossible to get exact solution. The value of N can be determined using distributions of random variables v_j and τ_j .

4.2 Distribution of Random Variables in DEP-i

First, let us discuss the distribution of v_j , i.e., the number of Type 1 patients arrive at block j . According to [7] as well as [20], the attendance of patients can be assumed to follow Binomial distribution. Let p_1 be the no-show probability of one Type 1 patient, then let us assume that the number of Type 1 patients that arrive at the j th block follows binomial distribution $v_j \sim B(r_j, 1 - p_1)$ satisfying (48).

$$\Pr\{v_j \text{ Type 1 patients will make the appointment at block } j\} = \binom{r_j}{v_j} (1 - p_1)^{v_j} p_1^{r_j - v_j}, \quad v_j = 1, \dots, r_j \quad (48)$$

Next, let us address the distribution of the block throughput τ_j which is the number of patients can finish the visit in block j . So far there is no direct description in the literature about the block throughput, but τ_j depends on the service time which in the literature has the following types of distributions: Exponential, uniform, Gamma,

Weibull and a few others [20]. In this paper, we assume two types of distribution for τ_j : a) Poisson distribution; b) uniform distribution. The assumption for Poisson distribution was based on the physician service time data collected through 30 days at a local clinic in 2013. We checked the Goodness of fit based on Chi-square Statistics of a few distributions such as Weibull, normal, beta, exponential, gamma, uniform, lognormal, and triangular against the data. Exponential is the Top-1 distribution in terms of Goodness of fit which can also be detected from the histogram of the data in Fig. 1. The assumption for uniform distribution is for the purpose of comparison with Robust Optimization in RO-i. Let l_j be the length of block j and ξ be mean service length per patient, the mean block throughput for τ_j is $\frac{l_j}{\xi}$ under the distribution Poisson and uniform. In the rest of this paper, we adopt the historic mean service length from the data, i.e., $\xi = 24$ min or $\frac{l_j}{\xi} = 5$, i.e., $\tau_j \sim \text{Poisson}(5)$.

4.3 Bound-Based Sampling Method

Without loss of generality, unless particular notice, in the following part of this section, we will discuss the solution method for the assignment at the first block, i.e., $i = 1$. DEP-1 for SIP-1 has $m(|K| + |T|) + |T|$ binary decision variables, $2m + 2mN$ integer decision variables and $(m + 1)(|K| + |T|) + 2m + m|T| + mN + N$ constraints. The size of the problem increases linearly with the number of scenarios. DEP-1 for SIP-1-multi has $(m(|K| + |T|) + |T|)p$ binary decision variables, $2mp + 2mN$ integer decision variables and $(mp + 1)(|K| + |T|) + 2mp + m|T| + (m + 1)N$ constraints. We can see that number of decision variables especially the binary variables is increased

Fig. 1 Histogram of Physician Service Time from Local Clinic

by p times, the number of constraints is increased by $(|K| + |T|)m(p - 1) + (2m - 1)p$ which in comparison with the number of scenarios may not be that significant. Given the increase of number of random variables in DEP-1 for SIP-1-multi, the number of scenarios will increase exponentially. Since SIP-i-multi has the same structure as SIP-i, here we focus on SIP-i for the solution method.

Let τ' be the smallest integer such that $\Pr\{\tau > \tau'\} \leq 0.05$, where τ denotes the number of patients that can be served in one block, then we can get that the number of scenarios for block j is $r_j\tau'$, so we have:

$$N \approx \prod_{j \in J} (r_j \tau') \quad (49)$$

For example, if $\tau' = 8$, $r_j = 2$, $|J| = 10$, then $N \approx 16^{10}$. DEP-i with 16^{10} scenarios is an approximation of the original problem, and it only includes 95% percent possible values of τ . This large number makes any exact optimization solver unable to solve DEP-i. Therefore, we adopt sample average approximation (SAA) method to pick a small sample size $\hat{N}$ and draw M batches of samples. As we mentioned earlier, the existing methods for choosing proper sample size $\hat{N}$ can be classified into exterior sampling and interior sampling. The exterior sampling method is to select/calculate a proper sample size in a designed algorithm and then solve the SAA using this sample size. The interior sampling method is to dynamically sample and solve the SAA in the designed algorithm where the most proper sample size and the optimal solution of corresponding SAA with this sample size will be achieved at the same step. In this paper, we adopt the interior sampling method which is implemented through lower and upper bound of the SIP-i model.

By solving the M approximated DEP-i models with $\hat{N}$ scenarios in each, we can obtain the lower bound of objective function value of the original DEP-i [50]. Let $\hat{f}_N^n$ be the objective function value of the n th batch, then the lower bound is given by

$$L_{M\hat{N}} = \frac{1}{M} \sum_{n=1}^M \hat{f}_N^n \quad (50)$$

The upper bound of the original DEP-i problem can be obtained by calculating objective function value with some feasible solution $\hat{X}, \hat{Y}$ with $\bar{M}$ batches of $\bar{N}$ scenarios where $\bar{M} \gg M$. Let $\bar{f}_N^n(\hat{X}, \hat{Y})$ be the objective function value for the n th batch, then the upper bound is

$$U_{\bar{M}\bar{N}} = \frac{1}{\bar{M}} \sum_{n=1}^{\bar{M}} \bar{f}_N^n(\hat{X}, \hat{Y}) \quad (51)$$

The confidence intervals of the lower and upper bounds can be determined using sample standard deviation of $\hat{f}_N^n$ and $\bar{f}_N^n(\hat{X}, \hat{Y})$ based on the central limit theorem [50].

A proper approximation of the original problem goes with a reasonable sample size which makes the model calculable and the objective function close to the “true value”. It is hard to obtain the “true value” of the original problem, but we

can utilize the lower bound and upper bound of the original problem to find a good estimation of the “true value”. Algorithm 1 below is designed to obtain the proper sample size for the approximation. This algorithm aims at finding a sample size that reasonably shrinks the average gap between lower bound and upper bound of the original problem. It increases the sample size used in lower bound calculation as in (50) and compares the average gap between the adaptive lower bound and averaged upper bound. Note that the upper bound compared in Algorithm 1 is not exactly the one as shown in (51) but an average level of upper bounds. $\bar{n}$ is the step length which indicate how much $\hat{N}$ increases per iteration, n_2 is the baseline for $\hat{N}$, n_1 is the number of iterations we have in searching for a good sample size. At the end of this algorithm, the sample size of lower bound with the smallest gap over the n_1 iterations will be the suggested sample size. All the computational experiments in this paper are performed with this sample size.

Algorithm 1 Bound-Based Sampling Method

Initialization : choose values for $n_1, n_2, \bar{n}, M, \bar{N}, \bar{M}$ where $\bar{M} \gg M$, let $D_0 = \infty$;
for $t = 1, \dots, n_1$ **do**
 Let $\hat{N} = n_2 + t\bar{n}$;
 for $n = 1, \dots, M$ **do**
 solve DEP-i with $\hat{N}$ samples, get optimal solution $\hat{X}, \hat{Y}$ and objective value $\hat{f}_{\hat{N}}^n$;
 plug in $\hat{X}, \hat{Y}$ with $\bar{N}$ samples and $\bar{M}$ batches and obtain $U_{M\bar{N}}^n$ using (51);
 get the difference $d_{\hat{N}}^n = U_{M\bar{N}}^n - \hat{f}_{\hat{N}}^n$;
 end for;
 Get $\bar{d}_{\hat{N}}^t = \frac{1}{M} \sum_{n=1}^M d_{\hat{N}}^n$;
 Let $D_t = \frac{1}{t} \sum_{k=1}^t \bar{d}_{\hat{N}}^k$;
 if $D_t < D_{t-1}$ **then**
 $\hat{N}(t)^* = n_2 + t\bar{n}$;
 $(X^*, Y^*) = \operatorname{argmin}_{(\hat{X}, \hat{Y})} \{f_{\hat{N}}^n \mid n = 1, \dots, M, \hat{N} = \hat{N}(t)^*\}$;
 end if
end for
Output $\hat{N}(t)^*$ as a proper sample size.
end Algorithm1

We compare the results from bound-based sampling method in Algorithm 1 and the interior sampling method proposed by from [53] which is illustrated in Algorithm 2. For the method in Algorithm 2, we change the step length $\bar{n}$ from 1 to 10 to keep it consistent with our sampling method. From the details of the two methods, we can see that Algorithm 1 distinguishes itself from Algorithm 2 in the following aspects: 1) in each iteration, Algorithm 1 uses the adaptive lower bound $\frac{1}{M} \hat{f}_{\hat{N}}^n$ and the average upper bound $\frac{1}{M} U_{M\bar{N}}^n$. The original Algorithm 2 was independent of any bound. 2) Algorithm 1 gets the sample size with the minimum accumulative average gap; Algorithm 2 suggests to use the sample size with the minimum accumulative average objective function.

Algorithm 2 Interior Sampling Algorithm from King and Wets [1991]

Initialization: let $t = 0$; $\hat{N} = 0$; $F_{\hat{N}} = 0$;
while $t < n_1$ **do**
 $t = t + 1$, $\hat{N}(t) = t\bar{n}$;
 solve DEP-i with $\hat{N}$ samples to obtain objective function value $\hat{f}_{\hat{N}(t)}^t$;
 Let $F_{\hat{N}} = \frac{1}{t} \sum_{i=1}^t \hat{f}_{\hat{N}(i)}^i$;
 $\hat{N}^* = \operatorname{argmin}_{\hat{N}} \{F_{\hat{N}}\}$;
end while
end Algorithm 2

Figure 2 shows the comparison of our bound-based sampling method versus King and Wets' sampling method on the DEP-1 model. For Algorithm 1, we set $n_1 = 20$, $n_2 = 10$, $\bar{n} = 10$, $M = 20$, $\bar{M} = 2000$, $\bar{N} = 50$, $|K| = 12$, $|T| = 13$, $m = 10$, $c_2 = c_3 = c_f = c_s = c_o = 1$. Unless specially mentioned, the computational experiments in the following context will use this setting. For Algorithm 2, besides the same model input data, we also set the same maximum iteration number as in Algorithm 1. In Fig. 2 we plot the gaps between lower and upper bounds for each iteration in Algorithm 1 and Algorithm 2. For Algorithm 1, the gap plotted is $\bar{d}_{\hat{N}}^t$ as calculated in Algorithm 1 for each batch size n . For Algorithm 2, the gap plotted was not used to decide the proper batch size but was calculated only for the comparison purpose. It is the gap between the upper bound in (51) and the averaged objective function value $F_{\hat{N}}$ calculated in Algorithm 2. In our experiments, the bound-based sampling method suggests using the sample size of 110, while King and Wets' algorithm implies sample size 180. The average computational time for DEP-i with 110 scenarios is around 0.21 second. Using the same settings and equipment, the average time for DEP-i with 180 scenarios is around 0.36 second. Based on the experiments, besides the slight advantage in saving computational time, the bound-based sampling method also shows a smaller gap in bounds as 5.93 from 110 scenarios versus 6.47 over 180 scenarios. Another benefit of the sampling method is that it produces

Fig. 2 Comparison of Sampling Methods

bounds for the objective function value while proposing a proper sample size. In addition, while using this algorithm, the confidence interval (C.I.) of the objective function value and the best solution can also be calculated with little cost of time.

4.4 Comparison of DEP-i and RO-i

Here we compare the results of DEP-1 and RO-1. In this numeric experiments, we set $|K| = |T| = \hat{t}, m = 10, c_2 = c_3 = c_f = c_s = c_o = 1, \tau \sim \text{uniform}(\underline{\tau}, \bar{\tau}), v \sim \text{uniform}(\underline{v}, \bar{v})$. We set 10 different pairs for $(\underline{\tau}, \bar{\tau}), (\underline{v}, \bar{v})$ and let $\hat{t} \in [5, 10, 15, 20, 25, 30, 35, 40]$. Then it means that the number of same-day requests ranges from 10 to 80. Table 7 shows the summary of differences between DEP-1 and RO-1. Column 1 shows the total number of the same-day requests received. Column 3 is about the difference in $|X| + |Y|$, i.e., the total assigned same-day request patients. Column 4 is the ratio between column 3 and column 1. We can see that the differences in assignments between DEP-1 and RO-1 are less than 4% of the requests. It implies that SIP-i and RO-i have close performance when all the random variables follow uniform distribution. The advantage of SIP-i over RO-i is the ability to handle various distributions of random variables ranging from closed-form distributions to empirical distributions.

5 The Aggregate Assignment Method

In this section, we will discuss how to apply SIP-i model in reality. In real clinics, patient requests arrive one by one, so the schedule of the requests is also made one by one. We call the schedule made for each single patient request the one-at-a-time assignment [7]. Here we propose the a different method named as aggregate assignment method which schedules multiple patients at the same time. The SIP-i model developed in this paper can be applied to both one-at-a-time assignment and aggregate assignment. For the former one, we just let $|K| + |T| = 1$ which means only one same-request is received at the block. For the latter one, the fundamental step is to estimate how many same-day requests the clinic receives in each block as elaborated

Table 7 DEP-1 vs. RO-1 Assignment Results

# Same-day requests	# Scenarios	Avg. difference in # assigned	Avg. Difference ratio
10	10	0.20	2.00%
20	10	0.32	1.60%
30	10	0.36	1.20%
40	10	0.64	1.60%
50	10	1.04	2.08%
60	10	2.32	3.87%
70	10	2.04	2.91%
80	10	2.00	2.50%

in the [Appendix](#). After the estimation, the SIP-i model is solved with the estimated number of requests. The assignment of each received request will follow the optimal solution of the SIP-i model according to the type of patient and the order of arrival. Let $\hat{s}$ be the estimator of $|K|$, i.e., number of Type 2 patients send request in the current block, and $\hat{w}$ be the estimator of $|T|$, i.e., number of walk-in patients arrive at the clinic in the current block. The following procedure shows how the aggregate assignment works using DEP-i. The one-at-a-time assignment procedure is shown in Algorithm 4.

Algorithm 3 The Aggregate Assignment Method

Initialization: $\hat{q} = 0, \beta^0 = 0, k = 0, t = 0$;
for $i = 1, \dots, m$ **do**
 Step 1: the current block is block i , update $\hat{q}, \beta^{i-1}$;
 Step 2: estimate $|K|$ and $|T|$, get $\hat{s}$ and $\hat{w}$;
 Step 3: choose proper sample size and run DEP-i model with $|K| = \hat{s}, |T| = \hat{w}$, obtain the optimal assignment solution x, y ;
 Step 4: **for** each request **do**
 if it is a Type 2 patient request
 $k = k + 1$;
 if $k \leq \hat{s}$: assign the k^{th} Type 2 patient according to $x_{jk}, \forall j \in (i, m)$;
 else if $k > \hat{s}$: solve DEP-i model with $|K| = 1, |T| = 0$, obtain the optimal solution $\hat{x}$,
 assign the Type 2 patient according to $\hat{x}$;
 if it is a Type 3 patient request
 $t = t + 1$;
 if $t \leq \hat{w}$: assign the t^{th} Type 3 patient according to $y_{jt}, \forall j \in (i, m)$;
 else if $t > \hat{w}$: solve DEP-i model with $|K| = 0, |T| = 1$, obtain the optimal solution $\hat{y}$,
 assign the Type 3 patient according to $\hat{y}$;
 update $\beta^i, \bar{a}, \bar{b}$;
 Step 5: after assigning all the requests of the current block, $i = i + 1$ go to Step 1;

Algorithm 4 The One-at-a-time Assignment Method

Initialization: $\hat{q} = 0, \beta^0 = 0$;
for $i = 1, \dots, m$ **do**
 Step 1: the current block is block i , update $\hat{q}, \beta^{i-1}$;
 Step 2: **for** each request **do**
 if it is a Type 2 patient request
 solve DEP-i with $|K| = 1, |T| = 0$, obtain the optimal assignment solution x ;
 assign this patient following according to x
 if it is a Type 3 patient request
 solve DEP-i with $|K| = 0, |T| = 1$, obtain the optimal assignment solution y ;
 assign this patient following according to y
 update $\beta^i, \bar{a}, \bar{b}$;
 Step 3: when all the requests of the current block are handled, $i = i + 1$ go to Step 1;

Theoretically, from an overall perspective, the aggregate assignment makes a better use of the available space of the remaining blocks, since it considers optimal assignment for a group of patients instead of an individual request. The advantage of aggregated assignment method is demonstrated through the following experiment. In the experiment, we compare the two assignment methods under two cases of

Table 8 Person-wise Assignment Cost Comparison

Probabilities	Underestimation				Overestimation
	4 requests	6 requests	8 requests	10 requests	16 requests
< Avg	0.25	1	1	0.9375	0.9375
< Lower Bound	0	0.5	0.5625	0.1875	0.6875
> Upper Bound	0	0	0	0	0

accuracy of request estimation: underestimation and overestimation. Table 8 shows the cost comparison between aggregate assignment and the one-at-a-time assignment for Block 1 for underestimation and overestimation.

For underestimation, where real number of patient requests are higher than the estimation, following Algorithm 3, we will run DEP-1 with aggregated mode for the estimated number, and then run DEP-1 with one-at-a-time mode for each of the remaining requests. In the experiment, four underestimation of request numbers are evaluated taking 16 as the real number of requests received. They are: (1) 4 same-day requests with 2 Type 2 requests and 2 Type 3 requests; (2) 6 same-day requests including 3 Type 2 and 3 Type 3 requests; (3) 8 same-day requests including 4 of each type; (4) 10 same-day requests with 5 of each type. The average assignment cost of the 16 same-day requests received in Block 1 is drawn using Algorithm 3 over 20 batches of 110 samples for the aggregated assignment. The average cost and 95% confidence interval (C.I.) of the one-at-a-time assignment are calculated using Algorithm 4. Table 8 shows the percentage of the assignment costs obtained from Algorithm 3 in different levels compared with the average costs or bounds obtained from Algorithm 4. Figures 3 and 4 illustrate the plots of the costs comparison. It is obvious that the aggregate assignment is no worse than the one-at-a-time assignment for the underestimation situation. For estimation larger than 6 requests, the aggregate assignment is significantly better than the one-at-a-time assignment on average.

For the case of overestimation, we still assume the real number of requests is 16, and the estimation of requests ranges from 16 to 20. We run the DEP-1 model with one-at-a-time mode for 20 batches of sample size 110 for 16 requests, then take the average assignment cost for the 16 requests and 95% C.I. of the costs. In contrast,

Fig. 3 Aggregate Underestimation Costs vs. Average One-at-a-time Costs

Fig. 4 Aggregate Underestimation Costs vs. 95% C.I. of One-at-a-time Costs

we run the DEP-1 model with aggregated mode for 16, 17, 18, 19, 20 estimated requests for the first block, and then take the average assignment cost for the first 16 requests. The last column of Table 8 as well as Figs. 5 and 6 show the comparison results. The prompt observation is that in the situation of overestimation, aggregate assignment is better than one-at-a-time assignment on average. From Fig. 6, we can see that if the real request number is less than 6, then the former is no worse than the latter; if the real request number is larger than or equal to 6, then the former is dominantly better than the latter.

6 Sensitivity Analysis and Value of SIP model

6.1 Importance of Request Estimation

Although under both underestimation and overestimation, the aggregate assignment shows strength in gaining better cost level on average, the importance of accuracy of request estimation can also be detected in Figs. 3 to 6. It is not hard to discover that the performance of aggregate assignment is overwhelming when the estimation is close to the real value. So it is worth conducting more experiments to explore the effect of request estimation. In the following experiments, let $\hat{s}$ be the estimated

Fig. 5 Aggregate Overestimation Costs vs. Average One-at-a-time Costs

Fig. 6 Aggregate Overestimation Costs vs. 95% C.I. of One-at-a-time Costs

Type2 requests of received in the first block and $\hat{w}$ be the estimated Type 3 patients arrive in the first block. Let $\hat{s} = \hat{w}$ and each of them increases from 5 to 43, so the total number of requests goes from 10 to 86. Then we solve DEP-1 for each of the estimations with 20 batches each and take average results. In Figs. 7, 8, and 9, we plot the objective function value and its three components: revenue associated with total number of assigned requests, cost of overflows (as patient waiting time cost), and cost of patient shortage (as physician idle time cost). A prompt observation on the trend of objective function over increment of requests is the bowl shape. It goes down from 10 requests to 34 requests, then keeps a flat pattern between 34 requests and 68 requests, and then goes up again presenting apparently three pieces of segments with two break points: 34 requests and 68 requests. There are obvious trembles in all of the three segments which are caused by the randomness of scenarios. We can explain this bowl shape in the following way: When the estimated request number is close to the real capacity of the system, the model gradually approaches saturation status with small overflows or shortages, which produces the flatness of the second segment. In the first segment, the overflow is close to zero, and the shortage cost dominates. So from Fig. 9 we can see that the trend of objective function value follows the trend of shortage cost in the first segment. In the second segment,

Fig. 7 Objective Value and Assigned Requests of DEP-1 with Different Request Estimations

Fig. 8 Objective Value and Patient Shortage Cost (g) of DEP-1 with Different Request Estimations

the overflow cost preserves an increasing trend at first half interval, then keeps a high level until the end of the second segment. The number of assigned requests also demonstrates a similar trend as an offset of the overflow cost. Together with the stable trend of shortage, they lead to a low and flat objective value level in this segment. In the third segment, the overflow level drops as a result of decreased number of assigned request, the conciliation among the three components leads to a mild increase in objective function value as compared with the second segment. These experiments are conducted with the setting of block throughput $\tau \sim \text{Poisson}(5)$, and number of blocks $m = 10$. Therefore, the clinic daily throughput is about 50 which is close to the center of the second interval [34, 68] where a lower level of overall cost and We can see that, if the estimation of total requests falls in the second interval [34, 68], then a low level overall cost can be guaranteed

6.2 Further Sensitivity Analysis

Besides the number of requests, a clinic manager may also be interested in the influence of scheduling parameters. With the DEP-i model and the bound-based sampling method, it is very convenient to conduct sensitivity analysis on parameters and settings. Here we choose six factors to be studied: r_i , c_2 , c_3 , c_f , c_s , l . Each factor has five levels as shown in the second column of Table 9. Since the objective function coefficient is involved as a factor, the magnitude of the objective

Fig. 9 Objective Value and Patient Shortage Cost (g) of DEP-1 with Different Request Estimations

Table 9 Rank of Importance of Parameters under Different Distributions of τ

Factors	Levels	Poisson			Uniform		
		$\sum \sum X + \sum \sum Y$	$\sum \mathbf{q}$	$\sum \mathbf{g}$	$\sum \sum X + \sum \sum Y$	$\sum \mathbf{q}$	$\sum \mathbf{g}$
r_i	1, 2, 3, 4, 5	2↓	4↑	4↓	2↓	3↑	4↓
c_2	1, 2, 4, 6, 8	6↑	6↑	6↓	4↑	6↑	6↓
c_3	1, 2, 4, 6, 8	4↑	5↑	5↓	3↑	5↑	5↓
c_f	1, 2, 4, 6, 8	3↓	1↓	1↑	1↓	1↓	1↑
c_s	1, 2, 4, 6, 8	5↑	2↑	2↓	6↑	2↑	2↓
l	58, 68, 78, 88, 98	1↑	3	3↑	5↑	4	3

function is no longer a proper metric. So we utilize the components of the objective function: the average number of assigned request, the average sum of $\mathbf{q}$ and the average sum of $\mathbf{g}$, as the metrics for the evaluation. The third to eighth columns of Table 9 present the ranks of the impact of factors under the metrics through the entropy-based analysis addressed in [55]. The higher the information gain is, the more the factor contributes to the change of the corresponding metric. Rank 1 implies the highest information gain, and 5 is the lowest. Column 3 to 5 are the results obtained under the distribution of $\tau \sim \text{Poisson}(\frac{l}{\xi})$, and Column 6 to 8 are with distribution of $\tau \sim \text{Discrete Uniform}(0, \frac{2l}{\xi})$. We can see that under Poisson distribution of τ , unit overflow cost and patient shortage cost are the most significant factors contributing toward total overflow cost and total patient shortage cost. Since ξ is fixed in this experiment, l is proportional to the mean throughput of each block, so the mean throughput is the most important factor contributing toward assignment ratio of the same-day request. Since value of r_i decides the capacity available for the same-day request, we can say that the available capacity of each block is also important to the assignment ratio of the same-day request. The importance rank changes when we set the distribution of τ as uniform. However, the overflow cost still dominates. This implies that the clinic needs to give more priority to controlling the waiting time cost. Given the fact that the two distributions of τ share the same mean value, the deviation between the ranks under the two distributions reveals that the second and higher-order statistics of block throughput bring significant impact to the assignment decision. The monotonic trends of these components versus increases of the factors are depicted using $\uparrow$ for increase and $\downarrow$ for decrease in Table 9. The changes of the values of the three objective components are monotonic with the increase of the six factors except for the block length. For the block length, the trends of total overflow cost are convex curves for both distributions, and the trend of total patient shortage cost is also a convex curve for uniform distribution. This indicates that the clinic can find a proper block throughput to minimize the overflow and patient shortage cost in a certain scope. Except for the block length, the trends of the components keep consistent under the two different distributions.

7 Conclusions

This paper suggests the clinic administrators who are practicing the open-access policy and block-wise assignment to adopt the aggregated assignment with SIP model. This method obeys the real event sequence of the clinic and is able to handle various real situations such as no-shows, patient preferences, FCFS rules, cancellation, earliness and lateness. It delivers a reasonable solution with the best revenue-cost balance incurring limited computational cost. Learning on the estimation of the same-day requests in each block, the SIP model executes the aggregate assignment which is shown through numerical examples to perform better than the traditional one-at-a-time assignment for both overestimation and underestimation. Rather than exhausting every sample in the sample space of the random variables, we develop the bound-based sampling method to gain a reasonable sample size for the approximation. This sampling method provides a lower gap between upper bound and lower bound of original objective value. Using the sample size gained from the sampling method, we perform sensitivity analysis on a few parameters and settings which in practice can offer meaningful insights for clinic cost control as well as key factor identification and monitoring. The advantage of the SIP model over the first-order statistics model is demonstrated through entropy analysis for different distributions of τ .

The proposed method does not specify the orders of appointments within the blocks, but the output of SIP-i model offers sufficient information. In practice, the clinic manager can arrange the appointments based on the optimal values of $\mathbf{X}$, $\mathbf{Y}$ following their requests sequence. Time allowance of each appointment can be obtained from mean service time or the ratio of block length over upper bound of block throughput. In our model, the clinic gives options for patients for the appointments, but the choice of patients is not involved. In practice, a patient can choose one of the appointment requests which are received in the same block. Apparently, the patient who sends a request earlier in the block can have more choices than those who send requests later. In implementation of the proposed method, we suggest the clinic to process and analyze historical data to gain information about the parameters such as no-show ratios, cancellation ratios, distributions of uncertain data and so on. Accuracy of these information is a prerequisite condition for the appropriate decision. It is recommended that decision makers of the clinic should pay attention to the consistency and stability of work efficiency of the block throughput, find a proper length for blocks, and make policies to reduce the waiting time cost and physician idle-time cost. What is more, better coordination of the assignment of the Type 1 patients and the same-day request patients will result in the cost-saving control. Last but not least, we show that the overall cost stays at a low level when estimation of the same-day requests is close to the “real” request number the clinic needs. Implementing the proposed method will not ask for a high level of accuracy in estimation of the same-day requests, a “scope” of the requests is sufficient.

Under COVID-19 circumstances, a lot of regular clinic schedules has been changed from office visiting into telehealth service. However, we can still find

the Type 1, Type 2 and Type 3 patients in the telehealth system. Especially, Type 3 patients will wait online in the telehealth system. Although all the models and algorithms in this paper are designed for the clinic scheduling problem, it can be conveniently applied to other service reservation systems or online job shop scheduling with time window or other assignment restrictions.

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1007/s43069-021-00089-6>.

Funding The authors declare that there were no funding sources that sponsored this research study.

Data Availability The initial data on patient arrival and patient service times was obtained from a local clinic under a confidentiality agreement that prevents us from disclosing the clinic information and the dataset. The data was aggregated to obtain the underlying distribution that was used and shown in Fig. 1 of the manuscript and Fig. 1 of the Appendix. The random data for the experiments were generated through Java code based on the parameters mentioned in Sects. 4.3 and 4.4. The algorithms in the paper were coded using Java and the different experimental scenarios were run using Java and CPLEX to produce the results that are discussed in the manuscript. Interested readers can contact the corresponding author to obtain a copy of the Java codes.

Declarations

Conflict of Interest The authors declare that they have no conflict of interest.

References

1. Kopach R, DeLaurentis PC, Lawley M, Muthuraman K, Ozsen L, Rardin R, Wan H, Intrevado P, Qu X, Willis D (2007) Effects of clinical characteristics on successful open access scheduling. *Health Care Manag Sci* 10:111–124
2. Liu N, Ziya S, Kulkarni VG (2010) Dynamic scheduling of outpatient appointments under patient no-shows and cancellations. *Manuf Serv Oper Manag* 12:347–365
3. Phan K, Brown SR (2009) Decreased continuity in a residency clinic: A consequence of open access scheduling. *Fam Med* 41(1):46–50
4. Robinson LW, Chen RR (2010) A comparison of traditional and open-access policies for appointment scheduling. *Manuf Serv Oper Manag* 12(2):330–346
5. Yan C, Tang J, Jiang B, Fung RYK (2015) Comparison of traditional and open-access appointment scheduling for exponentially distributed service time. *J Healthc Eng* 6(3):345–376
6. Yan C, Tang J, Jiang B (2014) Sequential appointment scheduling considering walk-in patients. *Math Probl Eng* 2014:564832
7. Muthuraman K, Lawley M (2008) A stochastic overbooking model for outpatient clinical scheduling with no-shows. *IIE Trans* 40:820–837
8. Denton B, Gupta D (2003) A sequential bounding approach for optimal appointment scheduling. *IIE Trans* 35:1003–1016
9. Erdogan SA, Denton B (2013) Dynamic appointment scheduling of a stochastic server with uncertain demand. *INFORMS J Comput* 25(1):116–132
10. Feldman J, Liu N, Topaloglu H, Ziya S (2014) Appointment scheduling under patient preference and no-show behavior. *Oper Res* 62(4):794–811
11. Patrick J, Putmeran M, Queyranne M (2008) Dynamic multi-priority patient scheduling for a diagnostic resource. *Oper Res* 56:1507–1525
12. Ahmadi-Javida A, Jalalib Z, Klassenc KJ (2017) Outpatient appointment systems in healthcare: A review of optimization studies. *Eur J Oper Res* 258(1):3–34

13. Chakraborty S, Muthuraman K, Lawley M (2010) Sequential clinical scheduling with patient no-shows and general service time distributions. *IIIE Trans* 42:354–366
14. Peng Y, Qu X, Shi J (2014) A hybrid simulation and genetic algorithm approach to determine the optimal scheduling templates for open access clinics admitting walk-in patients. *Comput Ind Eng* 72:282–296
15. Tsai PJ, Teng G (2014) A stochastic appointment scheduling system on multiple resources with dynamic call-in sequence and patient no-shows for an outpatient clinic. *Eur J Oper Res* 239:427–436
16. Liao C, Pegden CD, Rosenshine M (1993) Planning timely arrivals to a stochastic production or service system. *IIIE Trans* 25(5):63–73
17. Wang PP (1993) Static and dynamic scheduling of customer arrivals to single-server system. *Comput Oper Res* 24:703–716
18. Bailey N (1952) A study of queues and appointment systems in hospital outpatient departments with special reference to waiting times. *J R Stat Soc* 14:185–199
19. Lindley DV (1952) The theory of queues with a single server. *Proceedings Cambridge Philosophy Society* 48:277–289
20. Cayirli T, Veral E (2003) Outpatient scheduling in health care: A review of literature. *Prod Oper Manag* 12(4):519–549
21. Gupta D, Denton B (2008) Appointment scheduling in health care: Challenges and opportunities. *IIIE Trans* 40:800–819
22. Brahim M, Worthington DJ (1991) Queuing models for out-patient appointment systems: A case study. *J Oper Res Soc* 42(9):733–746
23. Kaandorp GG, Koole G (2007) Optimal outpatient appointment scheduling. *Health Care Manag Sci* 10(3):217–229
24. Brahim M, Worthington DJ (1991) The finite capacity multi-server queue with inhomogeneous arrival rate and discrete service time distribution and its application to continuous service time problems. *Eur J Oper Res* 50(3):310–324
25. Daya RW, Deanb MD, Garfinkela R, Thompson S (2010) Improving patient flow in a hospital through dynamic allocation of cardiac diagnostic testing time slots. *Decis Support Syst* 49(4):463–473
26. Weiss EN (1990) Models for determining estimated start times and case orderings in hospital operating rooms. *IIIE Trans* 22:143–150
27. Liu L, Liu X (1998) Block appointment systems for outpatient clinics with multiple doctors. *J Oper Res Soc* 49:1254–1259
28. Zhu ZC, Heng BH, Teow KL (2009) Simulation study of the optimal appointment number for outpatient clinics. *Int J Simul Model* 8(3):156–166
29. Balasubramanian H, Biehl S, Dai L, Muriel A (2014) Dynamic allocation of same-day requests in multi-physician primary care practices in the presence of prescheduled appointments. *Health Care Manag Sci* 17(1):31–48
30. Fries B, Marathe V (1981) Determination of optimal variable-sized multiple-block appointment systems. *Oper Res* 29(2):324–345
31. Lau H, Lau AH (2000) A fast procedure for computing the total system cost of an appointment schedule for medical and kindred facilities. *IIIE Trans* 32(9):833–839
32. Lin JK, Muthuraman, Lawley M (2011) Optimal and approximate algorithms for sequential clinical scheduling with no-shows. *IIIE Trans Health Syst Eng* 1(1):20–36
33. Liu L, Liu X (1998) Dynamic and static job allocation for multi-server systems. *IIIE Trans* 30:845–854
34. Vanden Bosch PM, Dietz CD, Simeoni JR (1999) Scheduling customer arrivals to a stochastic service system. *Nav Res Logist* 46:549–559
35. Vanden Bosch PM, Dietz CD, Simeoni JR (2000) Minimizing expected waiting in a medical appointment systems. *IIIE Trans* 32(9):841–848
36. Lin X, Janak SL, Floudas CA (2004) A new robust optimization approach for scheduling under uncertainty: i. bounded uncertainty. *Comput Chem Eng* 28:1069–1085
37. Mak H-Y, Rong Y, Zhang J (2015) Appointment scheduling with limited distributional information. *Manag Sci* 61(2):316–334
38. Mittal S, Schulz AS, Stiller S (2014) Robust appointment scheduling. In: Jansen K, Rolim JDP, Devanur NR, Moore C (eds) *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2014)*, vol 28. *Leibniz International Proceedings in Informatics (LIPIcs)*, pp 356–370

39. Bjorn P, Brian T, Berg D, Erdogan SA, Rohleder T, Hushka T (2014) Optimal booking and scheduling in outpatient procedure centers. *Comput Oper Res* 50:24–37
40. Robinson LW, Chen RR (2003) Scheduling doctor's appointments: Optimal and empirically-based heuristic policies. *IEE Trans* 35(3):295–307
41. Jiang R, Shen S, Zhang Y (2017) Integer programming approaches for appointment scheduling with random no-shows and service durations. *Oper Res* 65(6):1638–1656
42. Gupta D, Wang L (2008) Revenue management for a primary-care clinic in the presence of patient choice. *Oper Res* 56(3):576–592
43. Rohleder TR, Klassen KJ (2000) Using client-variance information to improve dynamic appointment scheduling performance. *Omega* 28(3):293–302
44. Wang W, Gupta D (2011) Adaptive appointment systems with patient preferences. *Manuf Serv Oper Manag* 12(3):373–389
45. Yeon N, Lee T, Jang H (2010) Outpatients appointment scheduling with multi-doctor sharing resources. *Proc Winter Simul Conf* 3318–3329
46. Li C, Bernal DE, Furman KC, Duran MA, Grossmann IE (2020) Sample average approximation for stochastic nonconvex mixed integer nonlinear programming via outer-approximation. *Optim Eng* 1–29
47. Verweij B, Ahmed S, Kleywegt AJ, Nemhauser G, Shapiro A (2003) The sample average approximation method applied to stochastic routing problems: A computational study. *Comput Optim Appl* 24:289–333
48. Shapiro A, Nemirovski A (2005) On complexity of stochastic programming problems. *Continuous Optimization Applied Optimization* 99:111–146
49. Kleywegt AJ, Shapiro A, de Mello TH (2001) The sample average approximation method for stochastic discrete optimization. *SIAM J Optim* 12(2):479–502
50. Ahmed S, Shapiro A (2002) The sample average approximation method for stochastic programs with integer recourse. *SIAM J Optim* 12:479–502
51. Higle JL, Sen S (1991) Stochastic decomposition: An algorithm for two stage linear programs with recourse. *Math Oper Res* 16:650–669
52. Higle JL, Sen S (1996) Stochastic decomposition: A statistical method for large scale stochastic linear programming. Kluwer Academic Publishers, 220
53. King AJ, Wets RJ (1991) Epi-consistency of convex stochastic programs. *Stochastics and Stochastic Reports* 34(1):83–92
54. Bagaram M, Tóth S (2020) Multistage sample average approximation for harvest scheduling under climate uncertainty. *Forests* 11
55. Fu Y, Banerjee A (2014) An entropy-based approach to improve clinic performance and patient satisfaction. *Proceedings of the 2014 Industrial and Systems Engineering Research Conference*

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.