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

Yu Fu
Department of Industrial and Systems Engineering, Texas A&M University
Amarnath Banerjee (banerjee@tamu.edu)
Department of Industrial and Systems Engineering, Texas A&M University

Research Article

Keywords: service scheduling, urgent scheduling, stochastic integer programming, no-show, can- cellation, punctuality

Posted Date: August 26th, 2020

DOI: https://doi.org/10.21203/rs.3.rs-64599/v1

Version of Record: A version of this preprint was published at Operations Research Forum on August 27th, 2021. See the published version at https://doi.org/10.1007/s43069-021-00089-6.

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

Yu Fu[1] and Amarnath Banerjee[2]

[1]Department of Industrial and Systems Engineering, Texas A&M University

College Station, TX, US

logistics.fuyu@gmail.com, 979-739-7185

[2]Department of Industrial and Systems Engineering, Texas A&M University

College Station, TX, US

banerjee@tamu.edu, 979-458-2341

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

Abstract

This paper addresses a scheduling problem which handles urgent tasks along with existing schedules. The uncertainty in this problem comes from random situations of existing schedules and arrival of upcoming urgent tasks. To deal with the uncertainty, 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 SIP model designed in the method can easily accommodate uncertainties of the problems, such as: no-shows, cancellations and punctuality of previously scheduled patients as well as random arrival and preference of new patients. To solve the SIP model, the deterministic equivalent problem formulations are solved using the proposed bound-based sampling method.

keyword: 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 aircrafts, automobiles and locomotives; or unscheduled visit of patients/customers. These urgent service requests may disturb the existing schedules and therefore cause delays in services. The scheduling decision for the service system needs to take these urgent requirements into consideration and reduce the cost caused by the delay. The delays of services come from both the unscheduled tasks and the existing 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. Under the pandemic of COVID-19, the increase of urgent requests for the primary care clinics ask for a more resilient scheduling decision supporting system. The methodology designed for urgent scheduling in clinics can provide some insight in preparing for and handling urgent uncertainties and can also be applied to other similar scheduling problems in industry.

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 allows patients to request their appointments when they need medical care. The comparison between open-access policy and traditional far-in-advance policy has been discussed by Kopach et al. [2007], Liu et al. [2010], Phan and Brown [2009] and is extensively studied by Robinson and Chen [2010], Yan et al. [2015]. Open-access policy shows its strength in improving patient access and satisfaction, as well as reducing patients no-show rate. Since patients can receive medical care when they need it, they can choose clinic care instead of urgent care. Thus, open-access policy contributes to enhance clinic accessibility. Another positive effect is the decrease in appointment delay which is also known as indirect waiting time [Liu et al., 2010]. 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 turn. 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 those papers addressing open-access policy, either the phone-call requests [Robinson and Chen, 2010] or the walk-in patients [Yan et al., 2014] is treated as the single type of same-day requests. Actually, we can detect the difference between the same-day phone-call requests and the walk-in patients from their request time and style, as well as their arrival patterns and scheduling rules. Different from some papers which allow the open-access policy to include appointments for the current day or the following day [Kopach et al., 2007, Robinson and Chen, 2010], in this paper, open-access policy is referred to as the same-day request scheduling policy. 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.

We assume that the same-day request scheduling is conducted over a block-based one-day horizon. The block-based scheduling method divides a clinic day into several time intervals with equal or unequal lengths where each interval is a block [Muthuraman and Lawley, 2008, Yan et al., 2014]. As it happens in a real clinic day, 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. The clinic does not know for sure 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. These incomplete information is not deterministic but tractable through prediction or distribution fitting, so we call it uncertain information. Since scheduling problem is an optimization problem and we are introducing "uncertain" data into scheduling, stochastic Integer Programming (SIP) is therefore exploited. This paper sets number of Type 1 and Type 2 patients arriving for visits and number of patients served per block as random variables. The decision variables of this paper are defined to assign the same-day requests received in 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. Additionally, although the distributions of the random variables are specified for calculation purpose, the generic framework of the study can be applied to any type of distributions.

2 Literature Review

Clinic appointment scheduling is a popular topic in OR/IE studies. Since the pioneering study on clinic appointment policy in the 1950's by Bailey [1952] and Lindley [1952] until now, new theories, methodologies and technology have been introduced for the clinic scheduling problem. For a comprehensive review on literature about clinic scheduling before 2003, the paper by Cayirli and Veral [2003] provides a broad background statement. The study by Gupta and Denton [2008] offers a more recent update on this topic up to 2008.

The classification of clinic scheduling research can be illustrated from different perspectives. From the view of objectives of scheduling, research papers aim at reducing cost, increasing revenue or combination of the two. The cost consists of patients' waiting time, doctors' idle time, overtime and so on [Cayirli and Veral, 2003]. The revenue is usually calculated by number of patients served or scheduled [Gupta and Wang, 2008, Liu et al., 2010]. The combination of revenue and cost can be either cost-savings as the difference between revenue and cost [Chakraborty et al., 2010, Muthuraman and Lawley, 2008, Tsai and Teng, 2014], or average cost which is cost per served patient [Peng et al., 2014]. In this paper, we adopt the cost-saving objective function which considers possible revenue of appointments, the waiting-time costs associated with overflows and the idle-time costs associated with patient shortage. In addition, the overtime cost can be added to the objective function using the number of overflowed patients from the last block.

For patient arrival mode, literature for optimal scheduling can be classified as non no-show arrival [Robinson and Chen, 2003], arrival with no-show [Erdogan and Denton, 2013, Tsai and Teng, 2014], and arrival with no-show as well as cancellations [Liu et al., 2010]. This work takes no-show rates as an essential factor influencing clinic scheduling decision, and also discusses cancellation and lateness of patients.

As for patient choice, articles can be divided into two categories. One is to allow patients make choices among time blocks in a day [Rohleder and Klassen, 2000, Gupta and Wang, 2008, Wang and Gupta, 2011], the other category offers choices over days for a patient [Feldman et al., 2014]. This paper deals with same-day requests, so patients' choices are circumscribed in the current day. Another way to classify the problem is to distinguish who defines the scope of patients' choices. Gupta and Wang [2008], Wang and Gupta [2011] assume that the patients decide their preference on the blocks and the clinic can accept one of the choices or reject. Feldman et al. [2014] suppose that the clinic defines a scope of choices and the patient chooses one day or decline the choices. Here we go with the proposal from Feldman et al. [2014] with some departure. In our work, the clinic does not know how many same-day patients will send their request nor their preference, but the clinic can estimate their choice scope which is reflected in our model as assignment restrictions. We propose two assignment restrictions: (1) the attendance delay for Type 2 patients who cannot attend the clinic immediately after request, and (2) the tolerance constraint for Type 3 patients who cannot wait too long in the clinic. Other particular patients' preferences can be included in these two types of assignment restrictions.

An important classification of scheduling is the online/offline scheduling. For research focusing on appointment assignment optimization, if the decision for all patients is made before the first block of a clinic day then the scheduling is considered static or offline [Liao et al., 1993, Wang, 1993]. 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 [Chakraborty et al., 2010, Denton and Gupta, 2003, Erdogan and Denton, 2013, Muthuraman and Lawley, 2008, Peng et al., 2014, Tsai and Teng, 2014]. The online scheduling handles the scheduling without complete information about number of people or service time. Online scheduling is addressed in this paper. In the following four paragraphs, we compare and contrast this paper with several similar papers in detail.

Muthuraman and Lawley [2008] as well as Chakraborty et al. [2010] 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. Tsai and Teng [2014] present a very similar work to Chakraborty et al. [2010] 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. 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 uncertain data.

Peng et al. [2014] assume three types of patients differentiated by their arrival modes 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, as well as best arrangement for Type 1 patients.

Denton and Gupta [2003] propose a two-stage stochastic programming model (SP) 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 Weiss [1990] and Wang [1993]. To facilitate solving the model fast and effectively, they developed aggregation bounds for the recourse function and bounds for dual multipliers in a block. Robinson and Chen [2003] also use a similar model as Wang [1993], 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. Erdogan and Denton [2013] extend this approach in Denton and Gupta [2003] 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. Although SP is the common tool to achieve best scheduling, these work differs from our paper in their focus. Their work is carried out to answer a question such as what are the best time allowance for each of the predefined sequence of patients given their random service time? Our work offers answer to how many of the randomly

arrived walk-ins and phone-call requests can be assigned in the remaining blocks.

In contrasting with the paper by Denton and Gupta [2003] which exploits the two-stage SP model and assumes that the variables in both stages are continuous, this paper uses SIP models for scheduling optimization. The advantage of SP for dealing with uncertainty in data, has also brought more complexity to the calculation. The large sample space of the random variable in a SP prevents people to exhaust every possibility in the distribution, instead, the sampling method is prevalently adopted to shrink the number of scenarios during the calculation [Shapiro and Nemirovski, 2005]. According to Ahmed [2010], the difficulty of evaluating the expected value can be solved by various sampling methods. Existing sampling methods include the interior sampling methods where samples are drawn during the course of solving the SP problem [King and Wets, 1991, Higle and Sen, 1991, 1996], and the exterior sampling methods which deal with the approximation model of the problem [Kleywegt et al., 2001]. For the interior sampling methods, King and Wets [1991] suggest that one can increase the number of scenarios by one at each iteration. For the nth 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. Higle and Sen [1991, 1996] develop a stochastic decomposition method which generates only one new sample at each iteration. As for exterior sampling methods, Kleywegt et al. [2001] discuss in detail the convergence and lower bound for sample size N. In this paper we develop a new sampling method based on bounds of objective values that distinguishes from the existing methods.

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 contrasts to most online scheduling method. 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. For example, the first and second block in the morning of the clinic day may expect more requests than the last block in the day. 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 by 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 2 patients for service in the current block. Step 6: The end of the current block. The successive block starts a new process from Step 1.

The significant difference of this method from one-at-a-time assignment is that, the assigning decision is based on estimation. Based on this event sequence, we build the two-stage SIP model. The first stage of SIP model deals with deterministic data, while the second stage estimates consequence and gives feedback with random scenarios. If the number of patients that can be served in a block exceeds the number of patients that may arrive in a block, an idle-time cost associated with patient shortage is produced. If the number of arrivals exceeds the number of patients that can be served in a block, then an overflow cost associated with patient waiting time is generated. Moreover, the real overflows will be added to the number of patients to be served in the immediate successive block. More assumptions of this model are: (1) There is one physician in the model.

(2) The clinic can reject requests of Type 2 and Type 3 patients. (3) There is a delay between arrival time of Type 2 patient requests and their scheduled block. (4) There is a waiting length tolerance for Type 3 patients. (5) Type 1 patients have no-show rates, all Type 2 patients will make the visits. (6) Assignment of Type 3 patients follows a FCFS rule. (7) We know the probability distribution of uncertain data (discrete, i.i.d.). (8) Waiting time cost is generated when patients overflow from original block into the successive block. (9) All patients who make the visits will come on time. (10) All blocks have equal length.

3.2 Decision Model for One Block

3.2.1 Notation

This model will use indices and parameters listed in Table 1 to Table 5.

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

Table 2: Parameters

m	number of blocks per clinic day
A_{kj}	preference factor of k^{th} Type 2 patient for j^{th} remaining block
B_{tj}	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
s_j	number of Type 2 requests received in j^{th} remaining block
w_j	number of Type 3 patients arrived at the clinic in j^{th} remaining block
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.
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

3.2.2 Formulations

Suppose the current block is the ith 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 remaining blocks

Table 3: First Stage Decision Variables

x_{kj}	binary variable showing whether the k^{th} Type 2 patient is assigned to block j
y_{tj}	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

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
$\nu_j(\omega)$	number of type 1 patients assigned to j^{th} remaining block that will make the appointment
$\alpha_j(\omega)$	number of Type 2 patients assigned to j^{th} remaining block that will make the appointment

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) are the second-stage constraints.

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

$$\text{s.t.} \quad \textstyle\sum_{j\in J} x_{jk} \le 1, \ \forall k \in K \tag{2}$$

$$\textstyle\sum_{j\in J} y_{jt} \le 1, \ \forall t \in T \tag{3}$$

$$x_{jk} \le A_{kj}, \quad j \in J, \ k \in K \tag{4}$$

$$y_{jt} \le B_{tj} z_t, \quad j \in J, t \in T \tag{5}$$

$$a_j = \bar{a}_{j+i} + \textstyle\sum_{k\in K} x_{jk}, \quad j \in J \tag{6}$$

$$b_j = \bar{b}_{j+i} + \textstyle\sum_{t\in T} y_{jt}, \quad j \in J \tag{7}$$

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

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

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)

where

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

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

$$q_1 = \acute{q} \tag{12}$$

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

Objective function (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. $\mathbb{E}[Q(X, Y, \mathbf{a}, \mathbf{b}, \omega)]$ is the expectation of the second-stage objective function value. From the second stage model, we can see that the SIP-i model has relatively complete recourse since for any $\eta(\omega) + \mathbf{a} + \mathbf{b} \in \mathbb{Z}^{|J|}$, we can always find $\mathbf{q}, \mathbf{g}$ such that (11) to (13) are satisfied.

Constraints (2), (3) are about 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 kth patient as shown in (14). The delay factor indicates the time length in terms of number of clinic blocks the patient needs to arrive at the clinic after request. 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 for preference and waiting tolerance of Type 3 patients. Each entry is defined by tolerance factor δ_t for tth patient as stated in (15).

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

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

Constraints (6), (7) are designed to store the accumulated number of patients assigned to each block 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 tth 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 tth 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 ν_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, $\acute{q} = 0$. 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:

$$q_{j+1} = \max\{\nu_j + q_j + a_j + b_j - \tau_j, 0\} \tag{16}$$

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

$$g_j = \max\{\tau_j - \nu_j - q_j - a_j - b_j, 0\} \tag{17}$$

Given the definition of η_j, i.e.

$$\eta_j := \nu_j - \tau_j \tag{18}$$

These equations lead to constraint (11). Distributions of the random variables of the second stage can be found in Appendix I. Formulations based on SIP-i considering patients no-shows, cancellations and punctuality can be found in Appendix II.

4 Solve The SIP-i Model

4.1 Deterministic Equivalent Problem and Bound-Based Sampling Method

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. In this paper, we demonstrate solution method for the SIP-i model which assumes only no-show of patients. In another paper, we will discuss the complex solution method for the complete version of this problem with constraints mentioned in Appendix II.

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 with only no-show is presented below.

$$\text{(DEP-i)} \quad \min \quad -c_2 \sum_{j\in J}\sum_{k\in K} x_{jk} - c_3 \sum_{j=1}^{h}\sum_{t\in T} y_{jt} - \sum_{n=1}^{N} p_n \left(c_f \sum_{j=1}^{m} q_j^n + c_s \sum_{j=1}^{m} g_j^n \right) \tag{19}$$

$$\text{s.t.} \quad \text{contraints (2) to (9)}$$

$$q_{j+1}^n - q_j^n - g_j^n = \eta(\omega)_j^n + b_j, \quad j \in J,\ n = 1, \cdots, N \tag{20}$$

$$q_1^n = \acute{q},\ n = 1, \cdots, N \tag{21}$$

$$q_{j+1}^n, g_j^n \in \mathbb{Z}^+, \quad j \in J,\ n = 1, \cdots, N \tag{22}$$

DEP-i is an integer programming problem, which can be solved using CPLEX for small N. The value of N can be determined using distributions of random variables. Similar to the SIP-i model,

for DEP-i, we assume that: 1) the arrival of Type 1 patients follows a Binomial Distribution described in Appendix I. 2) Type 2 patients have full attendance. 3) The number of patients served in one block follows a Poisson Distribution with mean $\frac{l}{\xi}$. There is no cancellation. 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:

$$N \approx \prod_{j \in J} (r_j \tau') \tag{23}$$

For example, let $\frac{l}{\xi} = 5$, we have $\tau' = 8$, let $r_j = 2, |J| = 10$, then $N \approx 16^{10}$. DEP-i with 16^{10} scenarios is an approximation of the original problem, since it only includes 95% percent possible values of τ. This large number makes CPLEX unable to solve DEP-i, so we adopt sample average approximation (SAA) method to pick a small sample size $\hat{N}$ and draw M batches of samples. By solving the M approximated DEP-i models with $\hat{N}$ scenarios for each, we can obtain the lower bound of objective function value of the original DEP-i [Ahmed and Shapiro, 2002]. Let $\hat{f}^n_{\hat{N}}$ be the objective function value of the nth batch, then the lower bound is given by

$$L_{M\hat{N}} = \frac{1}{M} \sum_{n=1}^{m} \hat{f}^n_{\hat{N}} \tag{24}$$

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 $\bar{f}^n_{\bar{N}}(\hat{X}, \hat{Y})$ be the objective function value for the nth batch, then the upper bound is

$$U_{\bar{M}\bar{N}} = \frac{1}{\bar{M}} \sum_{n=1}^{\bar{M}} \bar{f}^n_{\bar{N}}(\hat{X}, \hat{Y}) \tag{25}$$

The confidence intervals of the lower and upper bounds can be determined using sample standard deviation of $\hat{f}^n_{\hat{N}}$ and $\bar{f}^n_{\bar{N}}(\hat{X}, \hat{Y})$ based on the central limit theorem [Ahmed and Shapiro, 2002].

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 (24), 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 (25) 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 output as suggested sample size. The optimal solution will be derived from the approximation with the output sample size $\hat{N}$. All the computational experiments are also 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 $t = 1, \cdots, n_1$ **do**

Let $\hat{N} = n_2 + t\bar{n}$;

for $n = 1, \cdots, M$ **do**

solve DEP-i with $\hat{N}$ samples, get solution $\hat{X}, \hat{Y}$ and objective value $\hat{f}^n_{\hat{N}}$;

plug in $\hat{X}, \hat{Y}$ with $\bar{N}$ samples and $\bar{M}$ batches and obtain $U^n_{\bar{M}\bar{N}}$ using (25);

get the difference $d^n_{\hat{N}} = U^n_{\bar{M}\bar{N}} - \hat{f}^n_{\hat{N}}$;

end for;

Get $\bar{d}^t_{\hat{N}} = \frac{1}{M}\sum_{n=1}^{M} d^n_{\hat{N}}$;

Let $D_t = \frac{1}{t}\sum_{k=1}^{t} \bar{d}^t_{\hat{N}}$;

if $D_t < D_{t-1}$ **then**

$\hat{N}(t)^* = n_2 + t\bar{n}$;

$(X^*, Y^*) = \text{argmin}_{(\hat{X},\hat{Y})}\{f^n_{\hat{N}} \mid n = 1, \cdots, M, \hat{N} = \hat{N}(t)^*\}$;

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 some existing exterior and interior sampling methods. For exterior sampling method, in Kleywegt et al. 2001, the lower bound of sample size for ϵ-optimal solution to original problem with $1-\alpha$ probability is:

$$N \geq \frac{3\sigma^2_{\max}}{(\epsilon-\delta)^2} \log\left(\frac{|\mathcal{S}|}{\alpha}\right) \tag{26}$$

Here $\mathcal{S}$ is the set of feasible solutions, $\sigma^2_{\max}$ is the maximal variance of differences between objective values and $\delta \in [0, \epsilon]$. In DEP-i, the set of feasible solution consists of values for $X, Y, \mathbf{a}, \mathbf{b}$. Since $\mathbf{a}, \mathbf{b}$ depend on X, Y, so we are just concerned with the values of X, Y. The binary property of X, Y makes it easy to find out the upper bound of $|\mathcal{S}|$. Since m is the number of blocks, $|K|_i + |T|_i$ be the number of patients need to be assigned in block i. For each patient, only one of the m blocks can be set to 1, so we have:

$$|\mathcal{S}| \leq 2^{m(|K|_i+|T|_i)} \tag{27}$$

Using this method, let $(\epsilon-\delta)^2 \approx 3$, $\alpha = 0.01$, we get $|\mathcal{S}| \approx 2^{250}$, so we have the lower bound that can be written as $38\sigma^2_{\max}$. Since $\sigma^2_{\max}$ is the maximum variance of the objective function value, it will be no less than the squared difference between when we decide to accept all patients requested in block 1 and reject all patients in block 1 which is $(|K| + |T|)^2 = 625$. Under this situation, we have the lower bound of sample size larger than $38 \times 625 = 23750$ which is still too large for CPLEX to handle. Therefore, the method in Kleywegt et al. [2001] does not fit DEP-i.

For interior sampling method, the method proposed by Higle and Sen [1991, 1996] does not apply to the DEP-i model since it works on the basis of the stage-wise decomposition method of SP. So here we only compare our sampling method with the method from King and Wets [1991]. Figure 1 shows the comparison of our Bound-Based sampling method and Kleyweget's lower bound as well as King and Wets' sampling method on the DEP-1 model. For all the numerical calculations in this article, we have consulted with a clinic in town that provides outpatient services. The data used in this comparison has been adapted from their historical patient arrival and service time data. Here we set $n_1 = 20, n_2 = 10, \bar{n} = 10, M = 20, \bar{M} = 2000, \bar{N} = 50, |K| = 12, |T| = 13, m = 10, \xi = 5, c_1 = c_2 = c_3 = c_4 = 1$. Unless specially mentioned, the computational experiments in the

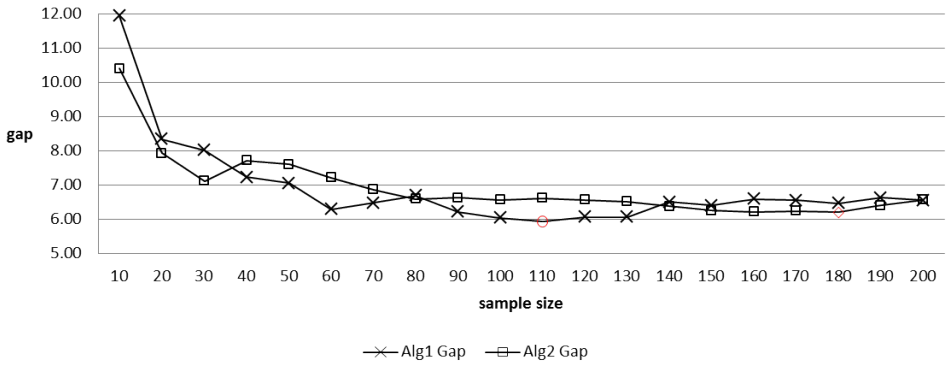


Figure 1: Comparison of Sampling Methods

following context will use this setting. For the method in [King and Wets, 1991], we change the step length from 1 to 10 to keep it consistent with our sampling method. Their procedure after modification is illustrated in Algorithm 2.

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

Initialization: let iter $= 0; n = 0; F_n = 0;$

while Stop criteria not satisfied **do**

iter $=$ iter $+ 1, n = n + \bar{n};$

solve DEP-i with n samples to obtain objective function value $\hat{f}_n^{\text{iter}}$;

Let $F_n = \frac{1}{n}\sum_{t=1}^{n} \hat{f}_n^{iter}$;

$\hat{N}^* = \text{argmin}_n\{F_n\};$

end while

end Algorithm 2

In Figure 1, we plot the gap D_t of averaged bounds as calculated in Algorithm 1 and the averaged objective function value F_n as calculated in Algorithm 2 over different number of samples. We can see that Bound-Based sampling method in the algorithm 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 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.

4.2 The Aggregate Assignment Method

The aggregate assignment method distinguishes itself from the one-at-a-time assignment in Muthuraman and Lawley [2008] by the feature of scheduling multiple patients at the decision step in each block. The fundamental step of the aggregate assignment is to estimate how many same-day requests the clinic receives in each block. After the estimation, the DEP-i model is implemented with the estimated number of requests, the assignment of each received request will follow the optimal solution of the DEP- i model according to the type of patient and the order of arrival. If the real number of requests of Type 2 or Type 3 patients is more than the estimated value, run the DEP- i model for each additional single request. The following procedure shows how the aggregate assignment works using DEP-i.

Algorithm 3 The Aggregate Assignment Method

Initialization: $\acute{q}^i = 0, \beta^i = 0, \forall i = 1, \cdots, m$;

for $i = 1, \cdots, m$ **do**

Step 1: the current block is block i, $\acute{q}^i = q_2^{i-1}$;

Step 2: estimate $|K|$ and $|T|$;

Step 3: choose proper sample size and run DEP-i model, obtain the optimal assignment solution;

Step 4: assign received requests one by one following the optimal assignment solution, update $\bar{\mathbf{a}}, \bar{\mathbf{b}}$ accordingly;

Step 5: **if** number of requests goes beyond the estimated value **then**

for each additional request **do**

set the DEP-i model for one request;

implement Step 3 and assign the request using the obtained solution ;

update $\bar{\mathbf{a}}, \bar{\mathbf{b}}$ accordingly;

end for

end if

Step 6: after assigning all the requests of the current block, update β^i, $i = i + 1$ go to Step 1;

end for

end Algorithm3

The one-at-a-time assignment procedure is shown in Algorithm 4.

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

Initialization: $\acute{q}^i = 0, \beta^i = 0, \forall i = 1, \cdots, m$;

for $i = 1, \cdots, m$ **do**

Step 1: the current block is block i, $\acute{q}^i = q_2^{i-1}$;

Step 2: **for** each request **do**

build DEP-i for this request properly;

choose proper sample size and run DEP-i model following Algorithm 1, obtain the optimal assignment solution;

assign this patient following the result of the solution, update $\bar{\mathbf{a}}, \bar{\mathbf{b}}$ accordingly;

end for

Step 3: when all the requests of the current block are handled, update β^i, $i = i+1$ go to Step 1;

end for

end Algorithm4

Table 6: 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

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 accuracy of request estimation: underestimation and overestimation. Table 6

Figure 2: Aggregate Underestimation Costs vs. Average One-at-a-time Costs

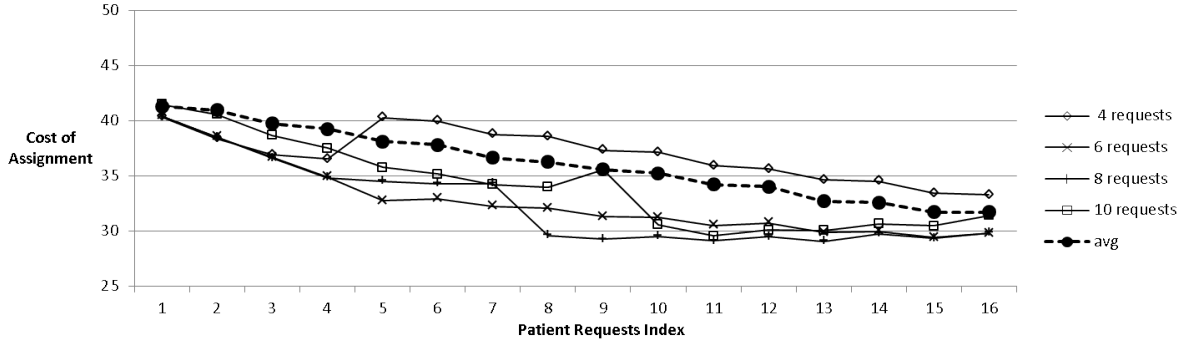
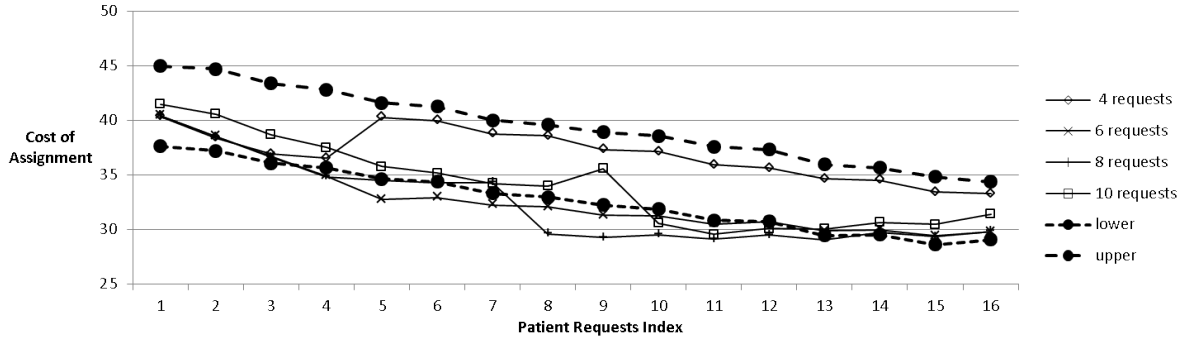


Figure 3: Aggregate Underestimation Costs vs. 95% C.I. of One-at-a-time Costs



shows the cost comparison between aggregate assignment and the one-at-a-time assignment for Block 1 for underestimation and overestimation.

For underestimation, from Algorithm 3 we can see that when the estimated request is smaller than the real number of requests, we need to 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 6 shows the percentage of the assignment costs obtained from Algorithm 3 in different levels compared with the average costs or bounds obtained

Figure 4: Aggregate Overestimation Costs vs. Average One-at-a-time Costs

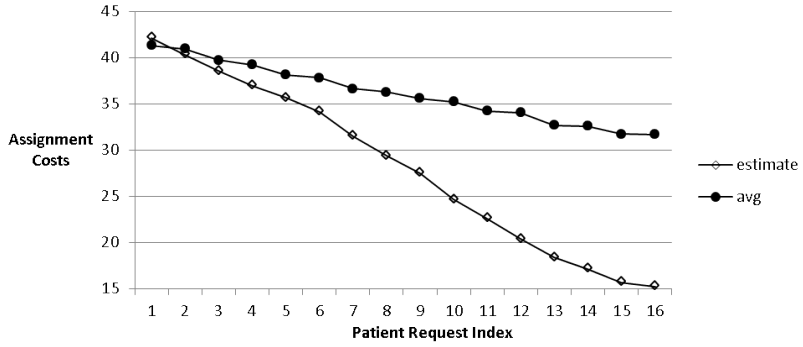
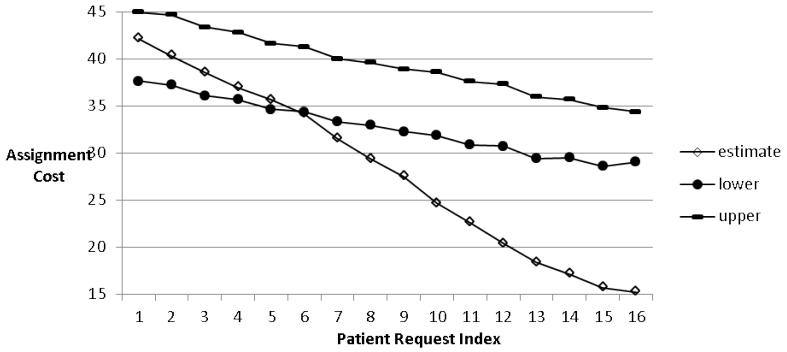


Figure 5: Aggregate Overestimation Costs vs. 95% C.I. of One-at-a-time Costs



from Algorithm 4. Figures 2 and 3 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, 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 6 as well as Figures 4 and 5 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 Figure 5, 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.

5 Sensitivity Analysis and Value of SIP model

5.1 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 Figures 2 to 5. 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, we set the estimated requests $\hat{s}, \hat{w}$ of the first block with the same value increasing 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 Figures 6 to 8, 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 Figure 8 we can see that the trend of objective function value follows the trend of shortage cost in the first segment. In the second segment, 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. Therefore, under the initial setting: $c_1 = c_2 = c_3 = 1, \tau \sim \text{Poisson}(5),\ m = 10,\ r_i \sim \text{uniform}(0, 5)$, if our estimation falls in the second interval [34,68], then a low level overall cost can be guaranteed.

Figure 6: Objective Value and Assigned Requests of DEP-1 with Different Request Estimations

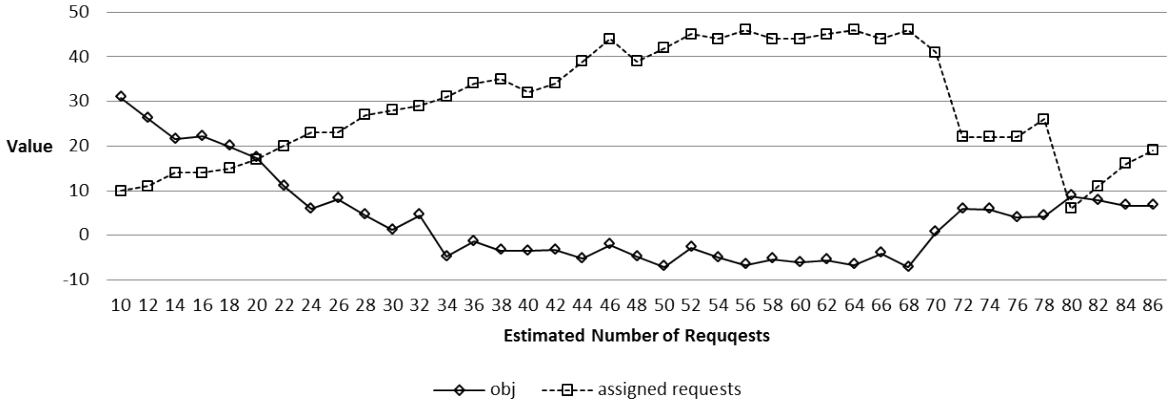


Figure 7: Objective Value and Overflow Cost ($\mathbf{q}$) of DEP-1 with Different Request Estimations

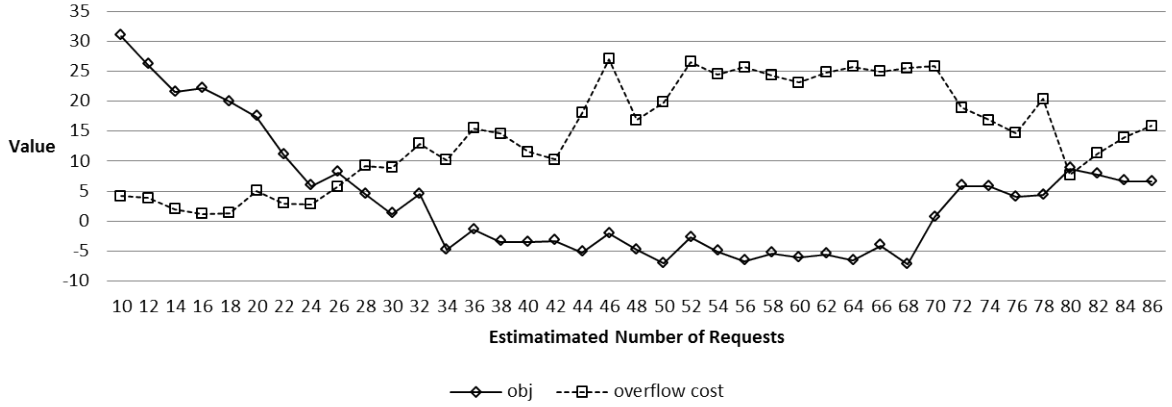
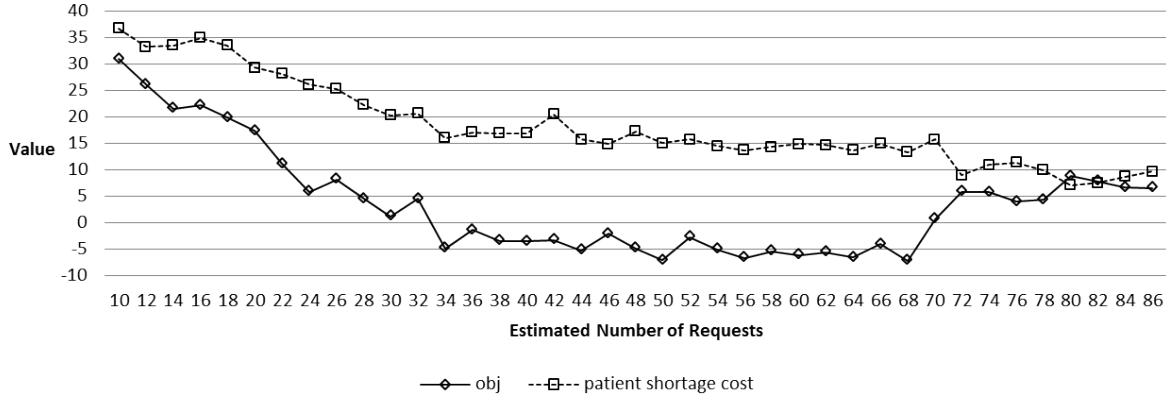


Figure 8: Objective Value and Patient Shortage Cost ($\mathbf{g}$) of DEP-1 with Different Request Estimations



5.2 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 7. Since the objective function coefficient is involved as a factor, the magnitude of the objective 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 7 present the ranks of the impact of factors under the metrics through the entropy-based analysis addressed in Fu and Banerjee [2014]. 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 $\tau \sim$ 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 7. 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.

Table 7: 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

6 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. Leaning 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 circumenstances, 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 teleheath 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.

References

S. Ahmed. Introduction to stochastic integer programming. `https://www2.isye.gatech.edu/~sahmed/eorms_sip.pdf`, 2010. Accessed: 2020–08–10.

S. Ahmed and A. Shapiro. The sample average approximation method for stochastic programs with integer recourse. *SIAM Journal of Optimization*, 12:479–502, 2002.

N. Bailey. A study of queues and appointment systems in hospital outpatient departments with special reference to waiting times. *Journal of the Royal Statistical Society*, 14:185–199, 1952.

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

S. Chakraborty, K. Muthuraman, and M. Lawley. Sequential clinical scheudling with patient no-shows and general service time distributions. *IIE Transactions*, 42:354 – 366, 2010.

B. Denton and D. Gupta. A sequential bounding approach for optimal appointment scheduling. *IIE Transactions*, 35:1003–1016, 2003.

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

J. Feldman, N. Liu, H. Topaloglu, and S. Ziya. Appointment scheduling under patient preference and no-show behavior. *Operations Research*, 62(4):794–811, 2014.

Y. Fu and A. Banerjee. An entropy-based approach to improve clinic performance and patient satisfaction. *Proceedings of the 2014 Industrial and Systems Engineering Research Conference*, 2014.

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

D. Gupta and L. Wang. Revenue management for a primary-care clinic in the presence of patient choice. *Operations Research*, 56(3):576–592, 2008.

J. L. Higle and S. Sen. Stochastic decomposition: An algorithm for two stage linear programs with recourse. *Mathematics of Operations Research*, 16:650–669, 1991.

J. L. Higle and S. Sen. Stochastic decomposition: A statistical method fo rlarge scale stochastic linear programming. *Kluwer Academic Publishers*, page 220, 1996.

A. J. King and R. J. Wets. Epi–consistency of convex stochastic programs. *Stochastics and Stochastic Reports*, 34(1):83–92, 1991.

A. J. Kleywegt, A. Shapiro, and T. Homem de Mello. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization*, 12(2):479–502, 2001.

R. Kopach, P. C. DeLaurentis, M. Lawley, K. Muthuraman, L. Ozsen, R. Rardin, H. Wan, P. Intrevado, X. Qu X, and D. Willis. Effects of clinical characteristics on successful open access scheduling. *Health Care Management Science*, 10:111–124, 2007.

C. Liao, C. D. Pegden, and M. Rosenshine. Planning timely arrivals to a stochastic production or service system. *IIE Transactions*, 25(5):63–73, 1993.

D. V. Lindley. The theory of queues with a single server. *Proceedings Cambridge Philosophy Society*, 48:277–289, 1952.

N. Liu, S. Ziya, and V. G. Kulkarni. Dynamic scheduling of outpatient appointments under patient no-shows and cancellations. *Manufacturing and Services Operations Management*, 12:347–365, 2010.

K. Muthuraman and M. Lawley. A stochastic overbooking model for outpatient clinical scheduling with no-shows. *IIE Transactions*, 40:820–837, 2008.

Y. Peng, X. Qu, and J. Shi. A hybrid simulation and genetic algorithm approach to determine the optimal scheduling templates for open access clinics admitting walk-in patients. *Computers & Industrial Engineering*, 72:282–296, 2014.

K. Phan and S. R. Brown. Decreased continuity in a residency clinic: A consequence of open access scheduling. *Family Medicine*, 41(1):46 –50, 2009.

L. W. Robinson and R. R. Chen. Scheudling doctor's appointments: Optimal and empirically-based heuristic policies. *IIE Transactions*, 35(3):295–307, 2003.

L. W. Robinson and R. R. Chen. A comparison of traditional and open-access policies for appointment scheduling. *Manufacturing & Service Operations Management*, 12(2):330–346, 2010.

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

A. Shapiro and A. Nemirovski. On complexity of stochastic programming problems. *Continuous Optimization Applied Optimization*, 99:111–146, 2005.

P. J. Tsai and G. Teng. A stochastic appointment scheduling system on multiple resources with dynamic call-in sequence and patient no-shows for an outpatient clinic. *European Journal of Operational Research*, 239:427–436, 2014.

P. P. Wang. Static and dynamic scheduling of customer arrivals to single-server system. *Computers and Operations Research*, 24:703–716, 1993.

W. Wang and D. Gupta. Adaptive appointment systems with patient preferences. *Manufacturing & Service Operations Management*, 12(3):373–389, 2011.

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

C. Yan, J. Tang, and B. Jiang. Sequential appointment scheduling considering walk-in patients. *Mathematical Problems in Engineering*, 2014(564832), 2014.

C. Yan, J. Tang, B. Jiang, and R. Y. K. Fung. Comparison of traditional and open-access appointment scheduling for exponentially distributed service time. *Journal of Healthcare Engineering*, 6 (3):345 – 376, 2015.

Figures

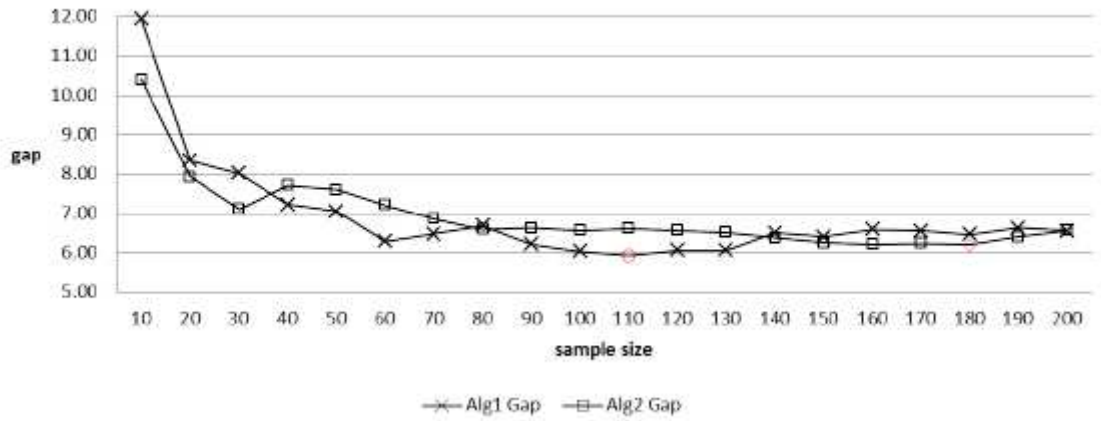


Figure 1

Comparison of Sampling Methods

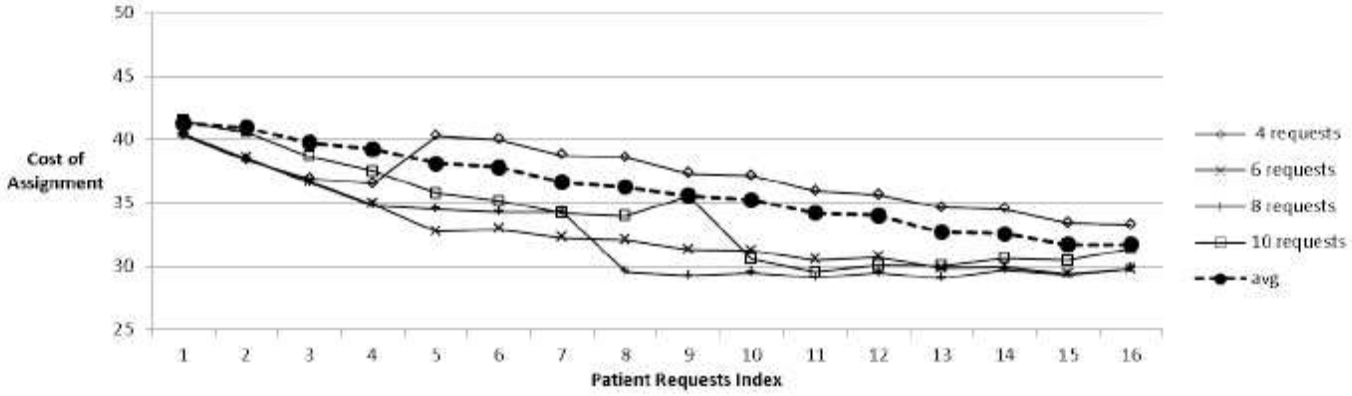


Figure 2

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

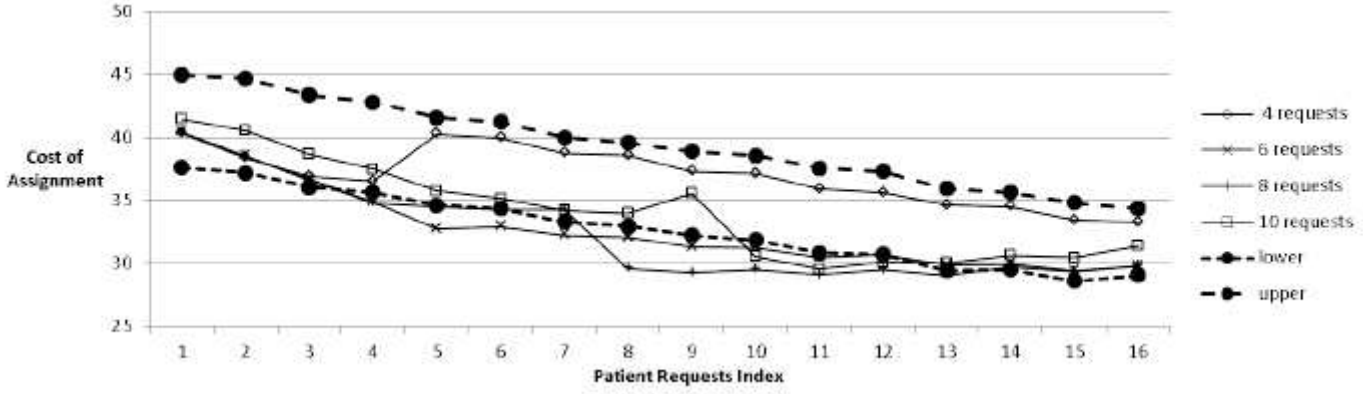


Figure 3

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

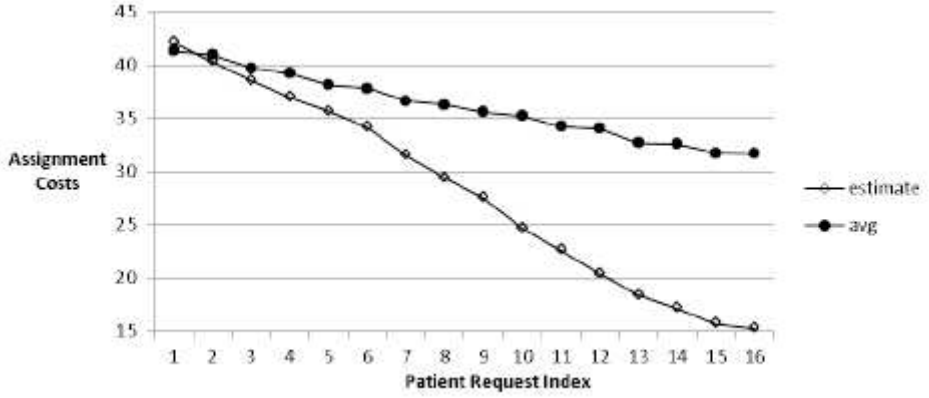


Figure 4

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

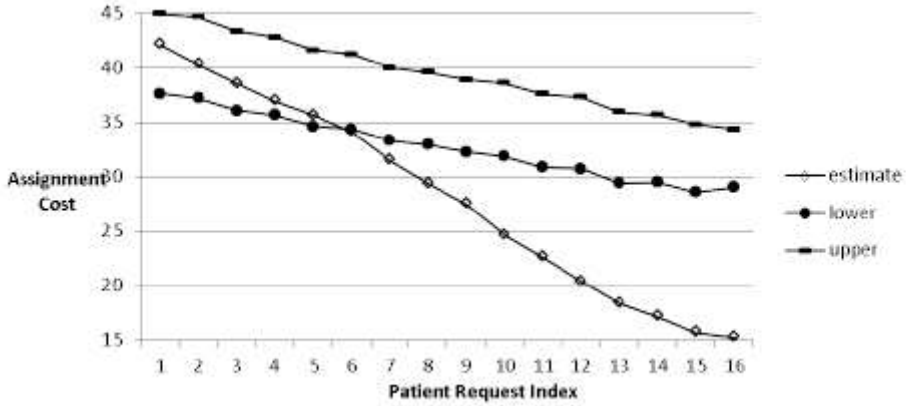


Figure 5

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

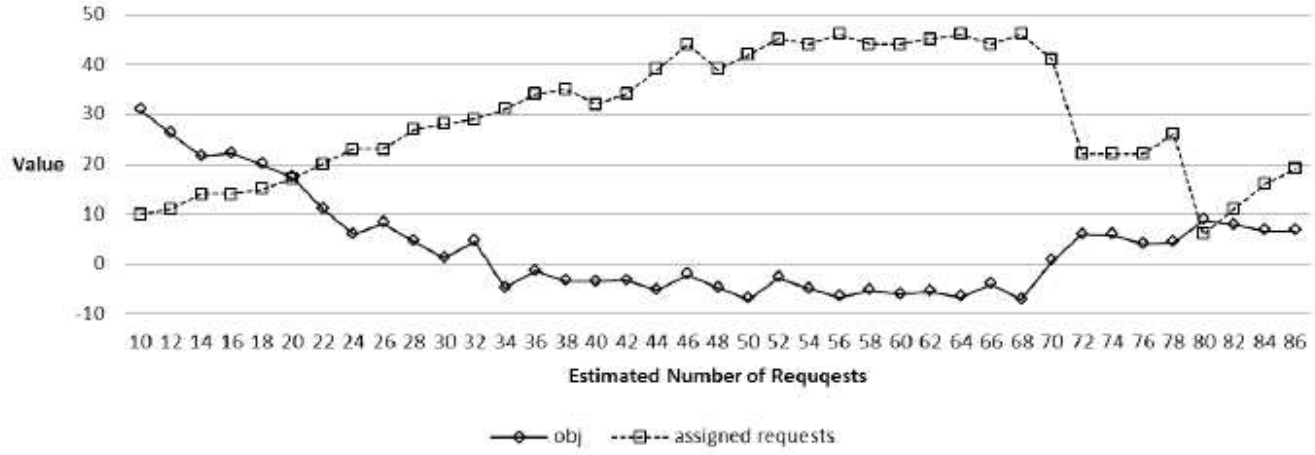


Figure 6

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

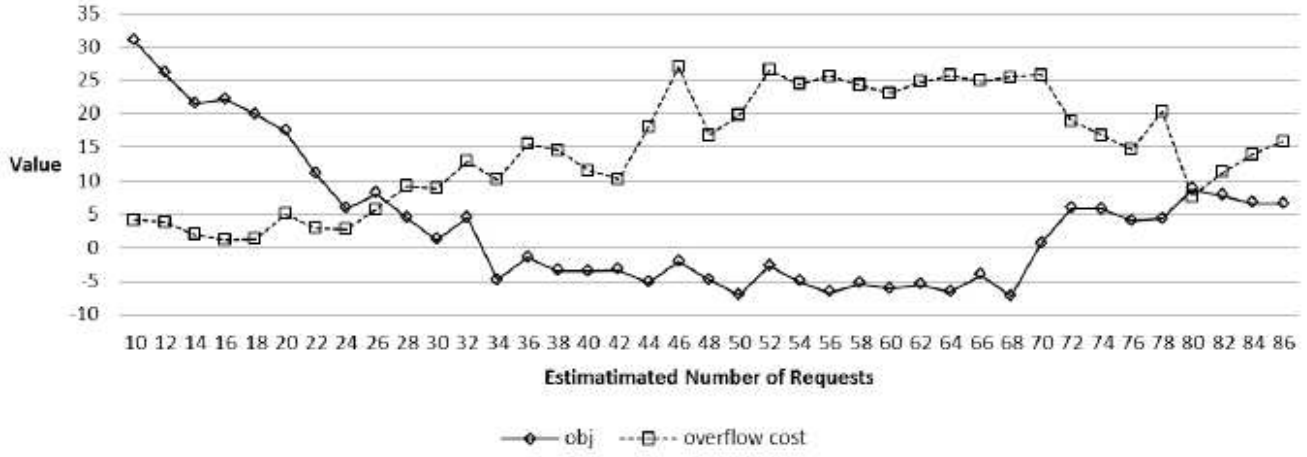


Figure 7

Objective Value and Overflow Cost (q) of DEP-1 with Different Request Estimations

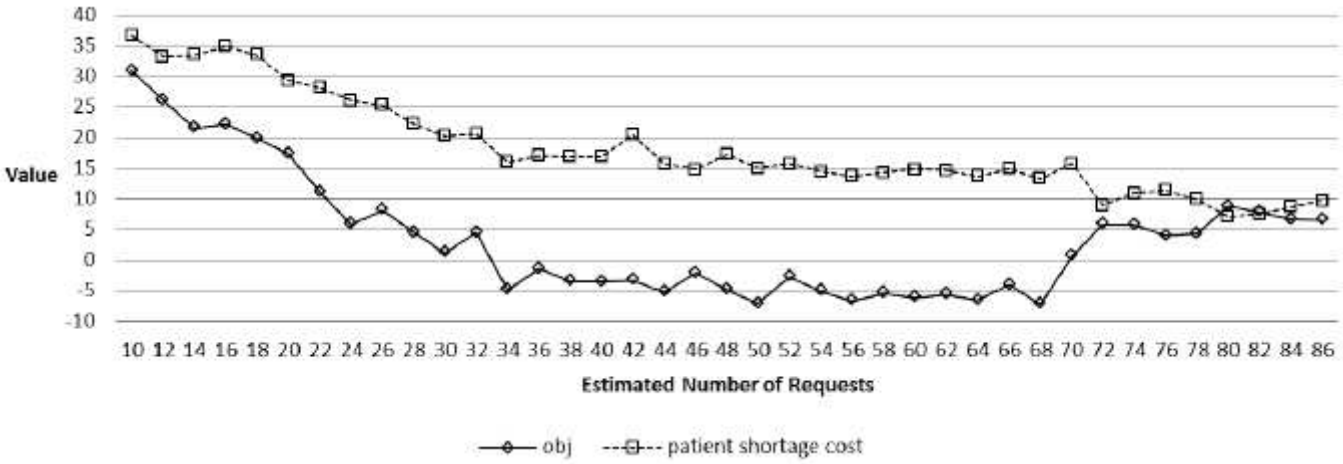


Figure 8

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

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- Appendices.pdf