BLOCK-BASED OUTPATIENT CLINIC APPOINTMENTS SCHEDULING UNDER OPEN-ACCESS POLICY

A Dissertation

by

YU FU

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Amarnath Banerjee
Committee Members,	Lewis Ntaimo
	V. Jorge Leon
	Subodha Kumar
Head of Department,	Mark A. Lawley

December 2016

Major Subject: Industrial Engineering

ABSTRACT

Outpatient clinic appointment scheduling is an important topic in OR/IE studies. Open-access policy shows its strength in improving patient access and satisfaction, as well as reducing no-show rate. The traditional far-in-advance scheduling plays an important role in handling chronic and follow-up care. This dissertation 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 physician in turn.

In this dissertation, the online scheduling policy is addressed for the Type 2 and Type 3 patients, and the offline scheduling policy is used for the Type 1 patients. For the online scheduling policy, two stochastic integer programming (SIP) models are built under two different sets of assumptions. The first set of assumptions ignores the endogenous uncertainty in the problem. An aggregate assigning method is proposed with the deterministic equivalent problem (DEP) model. This method is demonstrated to be better than the traditional one-at-a-time assignment through both overestimation and underestimation numerical examples. The DEP formulations are solved using the proposed bound-based sampling method, which provides approximated solutions and reasonable sample size with the least gap between lower and upper bound of the original objective value.

On the basis of the first set of assumptions and the SIP model, the second set of assumptions considers patient no-shows, preference, cancellations and lateness, which introduce endogenous uncertainty into the SIP model. A modified L-shaped method and aggregated multicut L-shaped method are designed to handle the model with decision dependent distribution parameter. Distinctive optimality cut generation schemes are proposed for three types of distribution for linked random variables. Computational experiments are conducted to compare performance and outputs of different methods. An alternative formulation of the problem with simple recourse function is provided, based on which, a mixed integer programming model is established as a convenient complementary method to evaluate results with expected value.

The offline scheduling aims at assigning a certain number of Type 1 patients with deterministic service time and individual preferences into a limited number of blocks, where the sum of patients' service time in a block does not exceed the block length. This problem is associated with bin packing problem with restrictions. Heuristic and meta-heuristic methods are designed to adapt the added restrictions to the bin packing problem. Zigzag sorting is proposed for the algorithm and is shown to improve the performance significantly. A clique based construction method is designed for the Greedy Randomized Adaptive Search Procedure and Simulated Annealing. The proposed methods show higher efficiency than traditional ones.

This dissertation offers a series of new and practical resolutions for the clinic scheduling problem. These methods can facilitate the clinic administrators who are practicing the open-access policy to handle different types of patients with deterministic or nondeterministic arrival pattern and system efficiency. The resolutions range from operations level to management level. From the operations aspect, the block-wise assignment and aggregated assignment with SIP model can be used for the same-day request scheduling. From the management level, better coordination of the assignment of the Type 1 patients and the same-day request patients will benefit the cost-saving control.

DEDICATION

To my parents and husband.

ACKNOWLEDGEMENTS

First, I would like to express my deepest gratitude to my advisor, Professor Amarnath Banerjee, who not only guides me in exploring the intellectual rationale and ideas behind this dissertation and all other research projects, but also provides generous support for my physical and spiritual life. The numerous and precious advice from Dr. Banerjee can always shed light on where I can move toward and where is the key to overcome the obstacles in my research and life. The honorable attitude and passionate behavior of Dr. Banerjee toward work and life give me thorough influence, under which, I have kept and will always stick to the rules of being kind toward people and keeping integrity in life and work.

I appreciate the tremendous help from my committee members, Prof. Lewis Ntaimo whose in-class lecture and after-class direction about stochastic programming is the essential key to accomplishing this dissertation, Prof. V. Jorge Leon and Prof. Subodha Kumar who provide valuable and pragmatic comments and suggestions on my research. Many thanks to them for their time and effort in assessing and checking this dissertation, their meaningful insights and support for my academic fulfillment and improvement. All my committee members show their pleasing personalities and professional performance in the past two years which always make me feel warm and hopeful. I also would like to deliver my sincere acknowledgment to my former adviser Prof. Tamás Terlaky, the chair of Industrial and Systems Engineering department at Lehigh University for his perceptive and conversant guidance which leads me to the world of mathematics and operations research and ensured my solid research foundation.

I thank my dear friends I made at Texas A&M University and Lehigh University for their companionship and help which make my life happy and wonderful. Finally, I dedicate

this dissertation to my family, my parents and my husband for their unconditional love, dedication and support.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

1. INTRODUCTION

1.1 Background Introduction

Under the Affordable Care Act (ACA), making clinics accessible for more patients when they need the care is a topic that is gaining increasing attention from medical practitioners. “Open-access policy" is among the solutions to enhancing clinic accessibility. Under this policy, clinics can accept patients who send their requests on the same day when they need the medical care. This dissertation addresses the outpatient clinic block-wise scheduling problem under a hybrid scheduling policy combining regular far-in-advance policy and the open-access policy. Open access policy allows patients request their medical appointments close to their visit dates. Under this policy, patients can receive medical care when they need it, they can choose clinic care instead of turning to urgent care. Another positive effect is the decrease in appointment delay which is also known as indirect waiting time [1]. The appointment delay is defined as the time length between patient request and their visit date. As a consequence of reduction in indirect waiting time, patient satisfaction is improved. What’s more, a longer appointment delay may increase the chances of patients missing an appointment, which leads to the increase of no-show rate. Therefore, the open-access policy is an effective way to reduce no-show probability. Despite the advantages of the open-access policy, the traditional far-in-advance scheduling is still practiced by clinics for chronic and follow-up care [2]. The comparison between open-access policy and traditional far-in-advance policy has been discussed in [3], [1, 2] and is extensively studied in [4, 5]. Among the large amount of research work regarding outpatient clinic scheduling, open access policy is fairly mentioned but rarely studied quantitatively under the three-type-patient framework which is derived from the differences in their request time and style, and consequently their arrival patterns and scheduling rules are also distinctive. The three types of patients include (1) the regular patients under the traditional far-in-advance assignment, (2) the same-day request patients under open-access policy who do not arrive at the clinic until the service time, and (3) the walk-in patients under open-access policy who arrive at the clinic without appointments. It is obvious that Type 2 and Type 3 patients are the same-day request patients.

The clinic 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 [6, 7]. Based on the three-type-patient framework, different research approaches are leveraged for scheduling different types of patients. The scheduling methods can be classified into online scheduling and offline scheduling. If the decision for all patients is made before the first block of a clinic day then the scheduling is considered static or offline [8, 9]. The offline scheduling is associated with decisions made with full information about number of people 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 [10, 11, 12, 6, 13, 14]. The online scheduling handles the scheduling without complete information about number of people or service time.

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. So when the clinic makes decision for the same-day requests, assignment of Type 1 patients are given as known information. Since the traditional far-in-advance policy deals with chronic and follow-up care, so the service time of each Type 1 patients is more predictable than the same-day requests. With this information, we can use offline scheduling approach to arrange the appointments of Type 1 patient. In practice, one policy for allocating Type 1 patients is to arrange the follow-up or chronic care patients into a limited portion of blocks leaving other blocks empty for the same-day requests. The occupied blocks for Type 1 patients are supposed to be fully utilized so that the same-day request decision maker can ignore them, consequently, the same-day request assignment will have decision variables with fewer dimensions.

However, for the same-day request patients, the clinic knows less information. 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“uncertain" data is considered in scheduling, stochastic Integer Programming (SIP) is therefore exploited for the same-day request patients online scheduling.

1.2 Research Topics and Contributions

According to the real scenarios in clinics, this dissertation studies the online scheduling for Type 2 and Type 3 patients in which assignment of Type 1 patients are taken as known information, arrivals and clinic throughput in future are uncertain. Assignment of Type 1 patients is addressed separately through offline scheduling. Three topics are discussed for the assignment problems for same-day requests and regular requests. The first and second topics focus on same-day request online scheduling with different assumptions on the uncertain data. The third topic describes the offline scheduling of Type 1 patients. The content and contributions of three topics are stated below.

1.2.1 Topic I: General block-wise online scheduling method for the same-day request patients

Under this topic, a two-stage SIP model for decision in each block is proposed to handle online scheduling with uncertain information about block throughput and no-shows of Type 1 patient. In those papers addressing open-access policy, either the phone-call requests [4] or the walk-in patients [7] is treated as the single type of same-day requests. This dissertation offers a pioneer work which provides exact optimization solution methods for online scheduling under the three-type-patient framework. The general model considers patient preferences and FCFS rule of Type 3 patients. An innovation in scheduling method namely, the aggregated assignment, is established. It distinguishes itself from the traditional one-at-a-time scheduling method by estimating the number of requests the clinic may receive in one block. In contrast to the papers addressing online scheduling which deals with deterministic number of patients in a clinic day and taking individual service time as uncertain data, this dissertation does not specify the number of patients to be served in a day. Individual service time is not directly used as uncertain data. Instead, throughput of each block is set as one of the essential uncertain data. For the online scheduling part, the dissertation has innovations in sampling method which generates a reasonably small sample size that minimizes the bounds, and the SIP solution method which addresses endogenous uncertainty where the decision variables of the first-stage influence the distribution of uncertain data of the second-stage.

At the time of writing this dissertation, a paper on this topic titled “Block Based Outpatient Clinic Online Scheduling Under Open-access Policy: A Stochastic Programming Model for Aggregate Assignment. By Yu Fu, Amarnath Banerjee” is under review in Manufacturing & Service Operations Management.

1.2.2 Topic II: Online scheduling for the same-day request patients with endogenous uncertainty

This topic is addressed on the basis of Topic I where no-shows of Type 2 patients, cancellations as well as punctuality of patients are introduced into the model. It is obvious that the assigned number of patients will affect the arrived number of patients for appointments. To be concrete, the number of Type 2 patients that are assigned in each block is a decision variable, if the no-show probability of Type 2 patients is considered, then the number of Type 2 patients who will make the visits with appointment becomes a random variable depending on the decision variable. This type of uncertainty is call endogenous uncertainty [15]. In this problem, the decision variables decide the upper bound of some of the random variables. Although SP model with endogenous uncertainty is well studied in literature, no existing method can be applied to handle the clinic scheduling problem here. Based on the analysis of problem properties, this dissertation considers different situations about the dependent random variables distribution, and develops a modified L-shaped algorithm as well as an aggregated multicut L-shaped algorithm to solve the SIP model.

At the time of writing this dissertation, a paper on this topic titled "Open-access Outpatient Clinic Online Scheduling under Endogenous Uncertainty. By Yu Fu, Amarnath Banerjee, Lewis Ntaimo" is being prepared for submission.

1.2.3 Topic III: Offline Scheduling for Type 1 patients

The clinic is assumed to know more information about Type 1 patients. In this topic, the number of Type 1 patients to be served in a clinic day and their expected service time are assumed to be deterministic. This dissertation suggests that the assignment of Type 1 patients can be handled as an offline scheduling. Generally speaking, this topic offers allocation plan of the patients into blocks so that accumulative service time in a block does not exceed the block length and minimum number of blocks are occupied. The decision time horizon is not necessarily restricted to one clinic day, so the blocks under consideration may span several days. Each patient has preference on the blocks, so the individual assignment must obey the corresponding restrictions. Given the service time of each patient and the time limit of each block, the clinic assigns all these patients into the blocks following the assignment restrictions. The target is that the smallest possible numbers of blocks are used to serve as many as possible Type 1 patients, so that more blocks are available for the same-day requests patients. Contributions of the dissertation on this topic include: developed integer programming formulations of the problem considering different types of assignment targets. Analyzed the complexity of the problem and the relation with other classical problems. Proposed a heuristic method which can perform the assignment efficiently and effectively compared with traditional heuristics like first-fit and best-fit. Designed a meta-heuristic algorithm with maximum independent set based construction, neighborhood representation and local search methods. The performance of the heuristic and meta-heuristic methods is compared with with the exact solution method as well as existing construction methods.

At the time of writing this dissertation, a paper on this topic titled “Zigzag Sorting and Maximum Independent Set Based Heuristic Meta-heuristic Methods for Restricted Bin Packing: An Application in Clinic Scheduling. By Yu Fu, Amarnath Banerjee” is under review in European Journal of Operational Research.

2. LITERATURE REVIEW

2.1 Clinic Scheduling Classifications

Since the pioneering study on clinic appointments scheduling in the 1950's by Bailey [16] and Lindley [17] 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, a broad background statement is available in [18]. The study in [19] offers a coherent update on this topic up to 2008. Beside the online/offline scheduling division, the following paragraphs provide different perspectives for classification of clinic scheduling.

Table 2.1: Classification of Clinic Scheduling Problems from the Perspective of Decisions

Types	Service Allowance	Appointment Order	Blocks	Literature
Type A	known	known	(+)	[20, 16, 21, 22, 23, 24]
Type B	decision	known	(+)	[20, 9, 11, 25, 12]
Type C	decision	decision	(+)	[10, 6, 14]
Type D	(-)	decision	known	[26, 13]

From the perspective of the known and unknown information and decision variables in the problem, we extend the three-type division in [20] into four types as shown in Table 2.1. The symbol (+) means blocks may or may not appear as known information in these problem types. The symbol (-) indicates that the service allowance may or may not be set as a decision variable. For Type A problem, there are no more decisions to be made when the scheduling result is completely available, the target is then to analyze the cost or factor effects. In this dissertation, multiple patients are assigned to multiple blocks, and the scheduling is completed dynamically for each block. The order of block defines the order of services. So this research falls into the category of Type D.

From the view of objectives of clinic 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 [18]. The revenue is usually calculated by number of patients served or scheduled [27, 1]. The combination of revenue and cost can be either cost-savings as the difference between revenue and cost [10, 6, 14], or average cost which is cost per served patient [13]. In this dissertation, the cost-saving objective function 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 [25], arrival with no-show [12, 14], arrival with no-show and cancellations [1]. 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 [28, 27, 29], the other category offers choices over days for a patient [30]. 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. [27, 29] assume that the patients decide their preference on the blocks and the clinic can accept one of the choices or reject. Feldman et al.[30] suppose that the clinic defines a scope of choices and the patient chooses one of the choices or declines the scheduling. This dissertation goes with the proposal from [30] with some departure. In this 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 this dissertation as assignment restrictions. Here, two assignment restrictions are proposed: (1) the attendance delay for Type 2 patients who cannot attend 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.

2.2 Clinic Appointments Online Scheduling

From the methods exploited for clinic scheduling, there are analytic study based on queuing theory [31, 32, 26, 33, 34], Markov-chain [35, 36, 1, 7], simulation modeling [37, 38], dynamic programming [39, 8, 40], stochastic programming (SP) as adopted in Topic I and II of this dissertation [10, 11, 12, 6, 25, 14], and other approximation algorithms [20, 41, 40, 42, 43]. Among these approaches, queuing theory, Markov-chain and dynamic programming are often applied together as stochastic analytic method. The following paragraphs present online clinic scheduling literature using stochastic analytic method, simulation and heuristiscs method, and stochastic programming method.

Muthuraman, Lawley [6] and Chakraborty [10] 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 [14] present a very similar work to [10] with improvement in applying this method to multiple resources and calculation of overflows using convolution estimation method and joint cumulative estimation method. The differences between this work and those above are quite perceptible. First of all, they assign only one patient at a time, using the one-at-a-time mode. An aggregated assignment is proposed here. Second, their assignment method is exhaustive and based on the first order statistic, i.e. the expected value of random variables. In this dissertation, a two-stage SIP model is introduced to handle multiple assignments with uncertain data. Third, the ways of formulating the overflows are not the same. They define the number of overflows as a random variable related to arrival number, assignment and service time. This model does not specify the distribution of overflows since they are decision variables, but the distribution of number of served patients per block as well as input and output of blocks are used here to address overflows.

Peng, Qu and Shi [13] assume three types of patients differentiated by their arrival modes, which is adopted in this paper. They work out comprehensive stochastic formulations to depict the assignment constraints such as 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. These parameters are assumed to be random variables here. 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. In comparison with this work, the same assumption about the types of patients are shared, but significantly distinctive methodologies are used. The advantage of this work is the promise of convergence, i.e., instead of a local optimal solution obtained from a meta-heuristic, the two-stage SIP model will return an exact global optimal result of the samples or report infeasibility or unboundedness.

Denton and Gupta [11] 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 [44] and Wang [9]. 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[25] also use a similar model as [9], 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 [12] extend this approach in [11] 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. They prove theoretically that the FCFS policy is optimal for scheduling with 2 patients. Although SP is the common tool to achieve best scheduling, these works differ from this dissertation 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? They focus more about the time the scheduled patients should come to see the doctor. This work offers answer to how many of the randomly arrived walk-ins and phone-call requests can be assigned in the remaining blocks. They ignore the arrival mode and patients' choices, which are considered here.

2.3 Stochastic Programming and Stochastic Integer Programming

The online scheduling in this dissertation uses SIP models for scheduling optimization. The advantage of SP for dealing with uncertainty in data, has also brought more complexity to calculation. The large sample space of random variable in SP prevents people to exhaust every possibility in the distribution; instead, the sampling method is prevalently adopted to shrink the number of scenarios during calculation [45]. The integer version of SP makes the solving process even harder [46] due to the loss of convexity. Ahmed [47] summarizes three difficulties of solving SIP models: (1) the tractability of the second-stage model, (2) the difficulty of evaluating the expected value in objective function, and (3) the optimization of the objective function. Nevertheless, there is a large body of research work for solving SIP models. Ahmed also introduces corresponding state-of-art in conquering the three difficulties [47]. According to [47], 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 [48, 49, 50], and the exterior sampling methods which deal with the approximation model of the problem [46]. For the interior sampling methods, King and Wets [48] 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 [49, 50] develop a stochastic decomposition method which generates only one new sample at each iteration. As for exterior sampling methods, Kleywegt, Shapiro and Homen de Melo [46] discuss in detail the convergence and lower bound for sample size N. In this dissertation, a new sampling method based on bounds of objective values that distinguishes from the existing methods is developed.

Research of SIP algorithms started from early 1980's. Stougie [51] in his thesis proposed algorithms for SIP. Schultz [52, 53] discussed properties of SIP and then with his coauthors conducted a study on the state-of-art on the topic [54]. After that, various methods are worked out to solve SIP which include but not limited to dual decomposition method designed by Caroe and Schultz [55], cutting plane methods from Caroe [56] as well as Sherali and Fraticelli [57], branch-and-bound method [58], and disjunctive decomposition methods from Sen, Higle and Ntaimo [59, 60, 61]. For SIP model with binary-first stage and pure integer complete recourse function, Laporte and Louveaux [62] propose a cut (named as L^2 cut by [59, 60, 61]) to handle the binary-first stage properly. The SIP models in Topic I and II of this dissertation fall in this category. For the model in Topic I, it is very convenient to convert the SIP model into a DEP integer model, which can be solved by CPLEX efficiently. As for the SIP model in Topic II, either L^2 cut or DEP transformation cannot be applied directly due to the endogenous uncertainty.

2.4 Stochastic Programming with Endogenous Uncertainty

Jonsbråten [63] states that two-stage SP problems can be divided into two categories based on the uncertainty types : (1) SP with exogenous uncertainty where random variables are independent of first-stage decision variables; (2) SP with endogenous uncertainty where random variables depend on first-stage decision variables. For SP model associated with endogenous uncertainty, there are subtle classifications addressed in literature [64, 65, 66]. Nevertheless, all the divisions are derived on basis of the two types: (I) the decision variables of the previous stage determine the uncertain information structure. (II) The decision variables of the previous stage change the probability distribution of random variables. There also exist models combining both parts. For type (I), the problem subcategories include: (a) decision dependent uncertain information revelation time, and (b) decision dependent number of random variables. For type (II), the subcategories can be (c) decision dependent distribution selection, or (d) decision dependent distribution parameters.

Research on type (I) topic can be found in [67, 68]. Existing solution methods for type (II) are customized according to the structure of the problem. For type (c) problems, Ahmed [69] proposes a 0-1 hyperbolic programming formulation based method to address the endogenous uncertainty where a decision alters distributions of random variables. Viswanath, Peeta and Salman [70] construct an equivalent deterministic program of a two-stage stochastic programming model for transportation network with decision dependent probability distribution of the random variables. Held and Woodruff [71] develop heuristics for multi-stage interdiction of stochastic networks. The method is specialized for the network structure.

For type (d) problems Vayanos, Kuhn and Rustem [72] propose a robust optimization based method to resolve the uncertainty where the uncertainty component can be observed only if the related binary variable is set to 1. To some extent, their work also falls into the category of (b). Laumanns, Prestwich and Kawas [73] study a SP problem with decision dependent probability of scenarios via Bayes' Rule. Their method is general but only applicable to binary random variables. The problem in this dissertation belongs to category (d) of type (II) where the first-stage decision variable defines the upper bound of the random variable. None of the existing methods can be applied directly to this problem, so a modified L-shaped method and aggregated multicut L-shaped method are proposed to solve this problem. The designed algorithms can be generalized to solve two-stage SP with decision dependent distribution parameter which is relevant to upper bound of random variables.

Solution methods for SP/SIP with endogenous uncertainty rest on methods for SP/SIP with exogenous uncertainty. This dissertation inherits most theories from SP/SIP methods. The L-shaped method built by Van Slyke and Wets [74] is modified and the multicut L-shaped method is provided by Birge and Louveaux [75] to accommodate the endogenous uncertainty. For multicut L-shaped method, Trukhanov, Ntaimo and Schaefer [76] develop a adaptive multicut aggregation method which contributes to reduce the size of the master problem. Two aggregation schemes are proposed in their dissertation: redundancy threshold and round on the number of aggregates. In this dissertation, a new aggregation scheme is exploited based on the subsets of random variables.

Application areas of the SP/SIP with endogenous uncertainty focus on gas [77], vaccination [66], networks [69, 71, 70] and location problem [69]. In scheduling area, SP models are adopted [11, 25, 44, 9] but none of them address endogenous uncertainty. This dissertation introduces SP/SIP with endogenous uncertainty into the scheduling problem as a complementary method for the widely used first-order analytic models.

2.5 Bin Packing

In Topic III, the offline clinic scheduling problem is associated with the Bin Packing Problem (BPP), which was studied for the first time by Johnson [78]. BPP is to assign n items with size in $(0, 1]$ into n bins with capacity of 1, and minimize the number of bins used. BPP is proven to be NP-complete [79, 80]. There are several approximation methods for BPP. The simplest one is the Next-Fit (NF) algorithm [81]. This method assigns the items in their index order to the bin which has capacity for it, the bins are checked also in their index order. If the current Bin i cannot hold the current item, a new bin , Bin $i + 1$ is introduced as the current bin, and the new item will start from the current bin. NF is a 2-approximation method [81]. There is a similar method named First-Fit (FF), its difference from NF is that the new item will always start the check from Bin 1. The upper bound of FF objective is $1.7k^* + 2$ where k^* is the optimal number of bins used. Another approximation method called Best-Fit (BF) is designed based on FF, the improvement is that BF searches the bin with the smallest residual capacity. BF has an approximation ratio of 1.7. If we sort the item in a non-increasing order according to their size, and apply the order on NF, FF, BF, we get Next-fit Decreasing (NFD), First-fit Decreasing (FFD) and Best-fit Decreasing (BFD) algorithms. The approximation ratio of these methods can as small as 1.5[81]. However, given the assignment restrictions of the clinic scheduling problem, those approximations cannot guarantee feasible solutions. FF and FFD are implemented for this problem, but it always ran into infeasible solutions where some patients cannot be assigned to their blocks because they are occupied by other patients. Besides the approximation approaches, there are research studies about the exact method on BPP [82, 83, 84, 85]. In this paper, a zig-zag sorting method is used to predefine the order of items (patients), maximum-degree fit and feasibility restore mechanism are established for the approximation method.

As for the application of meta-heuristics on BPP, Layeb and Chenche used Greed Randomized Adaptive Search Procedure (GRASP) to minimize the number of bins[86]. In construction phase, the greedy randomized algorithm is based on both FF and BF heuristics. Tabu Search is used as the local search algorithm. The neighborhood is defined as randomly selecting a bin from the result in the construction phase and packing all the items in the bin to other bins by using the First Fit strategy. By using benchmark data sets from three different classes, the GRASP result is shown to be very close to the best known result. Genetic Algorithm was applied in one-dimensional Bin Packing problem in [87]. The author introduced a new-defined chromosome which starts with the number of used bins and is followed by the weights of items in all the bins. The mutation phase moves the chromosome to its neighborhood which is defined as moving one object to another bin. The iterations are stopped as the difference between the best and worst solution reaches zero. Brugger et al. [88] proposed an ant colony optimization approach for the one-dimensional bin packing. Each ant constructs a solution by sequentially filling bins until all the items have been packed. If no more items can be added to the current bin, a new bin will be opened. By proposing a new pheromone decoding scheme and a new pheromone update strategy, they reached a result at least as good as hybrid grouping genetic algorithm by Falkenaue [89]. In this dissertation, Maximum Independent Set (MIS) is used to represent the allocation problem [90], GRASP and simulated annealing (SA) are used as search procedures.

3. BLOCK-BASED OUTPATIENT CLINIC ONLINE SCHEDULING UNDER OPEN-ACCESS POLICY: A STOCHASTIC PROGRAMMING MODEL FOR AGGREGATE ASSIGNMENT

In this section, the same-day request scheduling is conducted over a block-based one-day horizon. 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 is therefore exploited. The numbers of Type 1 and Type 2 patients arriving for visits and number of patients served 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 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 work establishes an aggregate assignment method with the SIP model. Besides the above features, this work also considers patients preferences, first-come-first-serve (FCFS) rule, patient lateness and cancellations. 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.

The work in this section 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 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.

3.1 Problem Statement and Formulation

3.1.1 Assumptions

For the block-wise scheduling, 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, the analysis and method can also be applied to the case of unequal length blocks. This section 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 of the current day.
- 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 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.

Step 2 is the estimation of number of requests which defines the dimension of decision variables in Step 3. The significant difference of this method from one-at-a-time assignment is that, the assigning decision is based on estimation. Step 4 is the beginning of assignment. The two-stage SIP model is developed on the basis on basis of this event sequence. The first stage involves the steps before decision Step 3, while the second stage evaluates the consequence of the decision. Information observed in Step 0 and 1 is deterministic, information estimated in Step 2 is also taken as deterministic. Information observed after Step 4 is uncertain. So the first stage deals with the deterministic data, while the second stage estimates consequence and gives feedback with random scenarios. The random variables in the second stage includes information about: the number of arrived Type 1 and Type 2 patients with appointments in each of the remaining blocks, as well as the number of patients the clinic can serve in each of the remaining blocks.

After the assign decision, consequences of the decision can be calculated using second-stage decision variables and random variables. 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 arrival 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 system. (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) The probability distribution of uncertain data is known and is discrete and independent. (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.

3.1.2 Decision Model for One Block

3.1.2.1 Notations

Indices used in the model are:

- $i, i \in 1, \cdots, m$: index of the current block.
- $j, j \in 1, \cdots, h$: index of the remaining blocks.
- $k, k \in 1, \cdots, \hat{s}$: index of Type 2 patients who send requests in block i.
- $t, t \in 1, \cdots, \hat{w}$: index of Type 3 patients who arrive in block i.
- $n, n \in 1, \cdots, N$: index of a scenario of the random variable.

Estimated parameters:

- $\hat{s}$: estimated number of Type 2 patient requests received in one block.

- $\hat{w}$: estimated number of Type 3 patients arrive at one block.

First-stage parameters are fixed as:

- m: number of blocks per clinic day.
- l: length of one block.
- $\boldsymbol{r} := \{r_j\}$: number of Type 1 patients scheduled for block j.
- A: the assign restriction matrix for Type 2 patient requests at block i.
- B: the assign restriction matrix for Type 3 patient requests at block i.
- c_2: unit revenue of assigning one Type 2 patients.
- c_3: unit revenue of assigning one Type 3 patients.
- c_f: unit cost for one patient overflows from one block to its successive block.
- c_s: unit cost for idle-time of physician (evaluated by patient shortage).
- N: number of scenarios of the random variable.
- ξ: mean service time for one patient.
- $h := m - i + 1$: number of remaining blocks.
- $\bar{\boldsymbol{a}} := \{\bar{a}_j\}$: number of Type 2 patients assigned to block j before the current block.
- $\bar{\boldsymbol{b}} := \{\bar{b}_j\}$: number of Type 3 patients assigned to block j before the current block.

First-stage decision variables are set as follows:

- $X^i := \{x^i_{jk}\}$: 0-1 decision variable showing whether kth Type 2 patient requested in current block is assigned to block j.

- $Y^i := \{y^i_{jt}\}$: 0-1 decision variable showing whether tth Type 3 patient arrived at current block is assigned to block j.

- $\boldsymbol{z}^i := \{z^i_t\}$: 0-1 decision variable showing whether tth Type 3 patient is assigned.

- $\boldsymbol{a}^i := \{a^i_j\}$: nonnegative integer variable for number of Type 2 patients assigned to block j till the end of current block.

- $\boldsymbol{b}^i := \{b^i_j\}$: nonnegative integer variable for number of Type 3 patients assigned to block j till the end of current block.

- β^i: the largest index of block which the Type 3 patients requested in block i are assigned to, so $\beta^i \in [i, m]$.

- $\acute{q}^i$: number of patients overflow from previous assigned block to the current block.

The superscript i means that they are the decision of the current block i. It also applies to the following notations. It is ignored in the modeling part. Second-stage parameters (random variables) are listed below: Let $\tilde{\omega}$ be the random variable and ω be its outcome, it contains the information of following uncertain data:

- $\boldsymbol{\nu}^i(\omega) := \{\nu^i_j(\omega)\}$: number of Type 1 patients assigned to block j and will arrive at block j.

- $\boldsymbol{\tau}^i(\omega) := \{\tau^i_j(\omega)\}$: number of patients can be served in block j.

- $\boldsymbol{\eta}^i(\omega) := \{\eta^i_j(\omega)\}$: right-hand-side of second-stage model with outcome ω which is a linear combination of the above random variables for blocks, i.e. $\eta^i_j(\omega) = \nu^i_j(\omega) - \tau^i_j(\omega)$.

Second-stage variables are defined as:

- $\boldsymbol{q}^i := \{q^i_j\}$: nonnegative integer variable for number of patients overflow from block $j-1$ to block j.
- $\boldsymbol{g}^i := \{g^i_j\}$: nonnegative integer variable for number of patients that can be served but are not assigned to block j (patient shortage or surplus capacity in block j).

3.1.2.2 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 are indexed from 1 to h, and their original indices are from i to $i+h$. Function and constraints (3.1a) to (3.1i) belong to the first-stage. Constraints (3.2b) to (3.2d) are the second-stage constraints.

$$\text{(SIP-i)} \quad \min \ -c_2 \textstyle\sum_{j=1}^{h} \sum_{k=1}^{\hat{s}} x_{jk} - c_3 \sum_{j=1}^{h} \sum_{t=1}^{\hat{w}} y_{jt} + \mathbb{E}[Q(X, Y, \boldsymbol{a}, \boldsymbol{b}, \omega)] \tag{3.1a}$$

$$\text{s.t.} \quad \textstyle\sum_{j=1}^{h} x_{jk} \le 1, \ \forall k = 1, \cdots, \hat{s} \tag{3.1b}$$

$$\textstyle\sum_{j=1}^{h} y_{jt} \le 1, \ \forall t = 1, \cdots, \hat{w} \tag{3.1c}$$

$$x_{jk} \le A_{jk}, \quad j = 1, \cdots, h, \ k = 1, \cdots, \hat{s} \tag{3.1d}$$

$$y_{jt} \le B_{jt} z_t, \quad j = 1, \cdots, h, \ t = 1, \cdots, \hat{w} \tag{3.1e}$$

$$a_j = \bar{a}_{j+i} + \textstyle\sum_{k=1}^{\hat{s}} x_{jk}, \quad j = 1, \cdots, h \tag{3.1f}$$

$$b_j = \bar{b}_{j+i} + \textstyle\sum_{p=1}^{\hat{w}} y_{jt}, \quad j = 1, \cdots, h \tag{3.1g}$$

$$\textstyle\sum_{t=1}^{\hat{w}} (i+j) y_{jt} \ge \beta^{i-1} \sum_{p=1}^{\hat{w}} z_t, \quad j = 1, \cdots, h, \quad t = 1, \cdots, \hat{w} \tag{3.1h}$$

$$x_{jk}, y_{jt} \in \{0, 1\}, \ j = 1, \cdots, h, \ k = 1, \cdots, \hat{s}, \ t = 1, \cdots, \hat{w} \tag{3.1i}$$

$$a_j, b_j \in \mathbb{Z}^+, \ j = 1, \cdots, h \tag{3.1j}$$

where

$$
\begin{aligned}
Q(X, Y, \boldsymbol{a}, \boldsymbol{b}, \omega) \quad = \quad & \min \quad c_f \textstyle\sum_{j=1}^{h} q_j + c_s \sum_{j=1}^{h} g_j && \text{(3.2a)} \\
\text{s.t.} \quad & q_{j+1} - q_j - g_j = \eta(\omega)_j + a_j + b_j, \quad j = 1, \cdots, h && \text{(3.2b)} \\
& q_1 = \acute{q} && \text{(3.2c)} \\
& q_j, g_j \in \mathbb{Z}^+, \quad j = 1, \cdots, h && \text{(3.2d)}
\end{aligned}
$$

Objective function (3.1a) 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. $\mathbb{E}[Q(X, Y, \boldsymbol{a}, \boldsymbol{b}, \omega)]$ 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 $\boldsymbol{\eta}(\omega) + \boldsymbol{a} + \boldsymbol{b} \in \mathbb{Z}^h$, we can always find $\boldsymbol{q}, \boldsymbol{g}$ such that (3.2b) to (3.2d) are satisfied.

Constraints (3.1b) and (3.1c) are about decisions of accepting or rejecting patient requests. Constraints (3.1d) and (3.1e) are about patient assignment restrictions. Matrix A with binary entries is the restriction matrix for Type 2 patients. It can be defined using a series of arrival delay factor $\boldsymbol{\rho} = \{\rho_k\}$ associated with the kth patient as shown in (3.3). 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 waiting tolerance of Type 3 patients. Each entry is defined by tolerance factor δ_t for tth patient as stated in (3.4).

$$
A_{jk} = \begin{cases} 1 & \text{if } j + i \in [\min\{i + \rho_k, m\}, m] \\ 0 & \text{otherwise} \end{cases} \tag{3.3}
$$

$$B_{jt} = \begin{cases} 1 & \text{if } j+i \in [j+i, \min\{i+\delta_t, m\}] \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

Constraints (3.1f) and (3.1g) are designed to store the accumulated number of patients assigned to each block by the end of current block. Variables $\boldsymbol{a}, \boldsymbol{b}$ appear in both first-stage model and second-stage model. Constraint (3.1h) 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.

The second-stage objective function (3.2a) 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 (3.2b) 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, p_j, a_j, b_j, the number of patients 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:

$$q_{j+1} = \max\{\nu_j + q_j + a_j + b_j - \tau_j, 0\} \tag{3.5}$$

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

$$g_j = \max\{\tau_j - \nu_j - q_j - a_j - b_j, 0\} \tag{3.6}$$

Given the definition of η_j, i.e.

$$\eta_j := \nu_j - \tau_j \tag{3.7}$$

These equations lead to constraint (3.2b).

3.1.3 Estimation of Parameters and Uncertain Data

3.1.3.1 Number of Same-Day Requests

For estimating the number of same-day requests, we assume that arrival of requests follows a Non-homogeneous Poisson Process (NHPP) which suggests different mean values for Poisson process during different periods of time. The Poisson process is widely used for patient arrival in clinic scheduling studies [18]. NHPP is usually suggested in phone-call arrival process [91]. Since estimation of request is done before the scheduling decision, in practice, the clinic can use other methods for estimation. Here it is assumed that the Type 2 requests in block i is from a NHPP with intensity function given by:

$$\lambda(i) = \begin{cases} \lambda_1 & \text{if } i \in [1, \lfloor m/4 \rfloor] \cup [\lfloor m/2 \rfloor + 1, \lfloor m/2 \rfloor + \lfloor m/4 \rfloor] \\ \lambda_2 & \text{if } i \in [\lfloor m/4 \rfloor + 1, \lfloor m/2 \rfloor] \\ \lambda_3 & \text{if } i \in [\lfloor m/2 \rfloor + \lfloor m/4 \rfloor + 1, m] \end{cases} \tag{3.8}$$

Let S_i be the number of requests received by the end of block i, S_{i-1} be the number of requests received by the end of block $i-1$, then the number of requests received in block

i denoted by $s_i = S_i - S_{i-1}$ has a Poisson distribution with mean given by:

$$E[s_i] = \int_{i-1}^{i} \lambda(i)\, \mathrm{d}i \tag{3.9}$$

The expected number of requests received in block i equals the mean of s_i. We assume the arrival of Type 3 patients also follows NHPP with intensity function:

$$\zeta(i) = \begin{cases} \zeta_1 & \text{if } i \in [1, \lfloor m/4 \rfloor] \cup [\lfloor m/2 \rfloor + 1, \lfloor m/2 \rfloor + \lfloor m/4 \rfloor] \\ \zeta_2 & \text{if } i \in [\lfloor m/4 \rfloor + 1, \lfloor m/2 \rfloor] \cup [\lfloor m/2 \rfloor + \lfloor m/4 \rfloor + 1, m] \end{cases} \tag{3.10}$$

In the same way, the expected number of walk-in patients that arrive in block i denoted by w_i is:

$$E[w_i] = \int_{i-1}^{i} \zeta(i)\, \mathrm{d}i \tag{3.11}$$

3.1.3.2 Attendance

If we do not consider punctuality and cancellation of patients, the attendance of patients can be assumed to follow Binomial distribution as suggested in literature [6, 18]. Let p_1 be the no-show probability of one Type 1 patient, assume that the number of regular patients that arrive at the ith block follows Binomial Distribution $\nu_i \sim \mathrm{B}(r_i, 1 - p_1)$, and $\boldsymbol{v}$ be a $1 \times r_i$ vector such that:

$$\Pr\{\nu_i \text{ patients arrive at block } i\} = \binom{r_i}{\nu_i} (1 - p_1)^{\nu_i} p_1^{r_i - \nu_i}, \quad \nu_i = 1, \cdots, r_i \tag{3.12}$$

In the same way, if we assume each of the Type 2 patient has a no-show rate of p_2, then we have:

$$\Pr\{\alpha_i \text{ patients arrive at block } i\} = \binom{a_i}{\alpha_i}(1-p_2)^{\alpha_i} p_2^{a_i-\alpha_i}, \quad \alpha_i = 1, \cdots, a_i \tag{3.13}$$

3.1.3.3 Number of Patients that Can be Served in a Block

In literature, the distributions suggested for service time include: Exponential, Uniform, Gamma, Weibull and a few others [18]. In this dissertation, the number of patients served in a block is set as a random variable. The distribution of this random variable can be derived from the service time. Three distributions are considered here: Poisson, Discrete Uniform and distribution derived from exponential service time. For the first two distributions, the mean served number is $\frac{l_i}{\xi}$, where l_i is the length of block i. For the last one, the following analysis is performed. Let T_k^i be the sum of service time of the first k patients in block i, then we have:

$$\Pr\{\tau \text{ patients are served in block } i\} = \Pr\{T_{\tau+1}^i > l_i \text{ AND } T_\tau^i \le l_i\} \tag{3.14}$$

Assume that service time of each patient follows a homogeneous Exponential Distribution with mean ξ, so T_k^i follows a Gamma Distribution with shape parameter k and scale distribution ξ. Then the distribution can be updated as follows:

$$\begin{aligned}\Pr\{\tau \text{ patients are served in block } i\} &= \Pr\{T_{\tau+1}^i > l\}\Pr\{T_\tau^i \le l\} \\ &= (1 - F(l; \tau+1, \xi))F(l; \tau, \xi)\end{aligned}, \quad \tau \ge 1 \tag{3.15}$$

where $F(*)$ is the *cdf* of Gamma Distribution. If there are unequal lengths of blocks, different values for l_i can be plugged in accordingly.

3.1.4 Special Cases

3.1.4.1 Unequal Lengths of Blocks

Assume that the blocks have unequal lengths, then estimation of number of patient requests received in a block should be adjusted with the length of the block. The distribution of number of patients that can be served in a block should also be updated. This situation has been discussed in detail in Section 3.1.3.3.

3.1.4.2 No-shows of Type 2 Patients

The SIP-i model is built based on the assumption of zero no-show rates of Type 2 patients. In reality, the same-day request patients can also have a certain rate of no-show [13]. In this situation, we can define a new random variable $\boldsymbol{\alpha} := \{\alpha_j\}$ representing the number of Type 2 patients arriving with appointments. The distribution of α_j depends on the value of a_j which is a first-stage decision variable. Introducing this random variable brings endogenous uncertainty to this problem. It means that the decision variable will affect the distribution of the random variable. The definition of η_j should be changed into:

$$\eta_j := \nu_j + \alpha_j - \tau_j \tag{3.16}$$

With endogenous uncertainty, SIP-i can only be solved by stage-wise decomposition with special cuts which is discussed in Section 4.

3.1.4.3 Cancellations of Same-day Request Patients

If the cancellation of patients is considered, additional assumptions and settings need to be introduced. This section only considers cancellations of Type 2 and Type 3 patients. Cancellations of Type 1 patients will be discussed in Section 3.1.4.4. For the current day, given the index of current block as i, let (p, j, t) be a triple of index associated with every assigned patients so far. Suppose the decision of cancellation is made at the beginning of each block. For $t = 2, 3$, the triple denotes patients with type t that made the request at $(i - p)$th block of today and were assigned to the the $(i + j)$th block. Especially, cancellation for Type 3 patients means they leave the clinic before the assigned block. Let L be the length between the request received block and the block when the patient decides to cancel the appointment. Let α_{pjt} be the probability that patient with (p, j, t) who have not canceled the appointment by the current block will arrive for the appointment. Then the probability is defined as:

$$\alpha_{pj2} = \Pr\{L \geq i + j + 1, \text{arrive for appointment} \mid L \geq i\} \tag{3.17}$$

$$\alpha_{pj3} = \Pr\{L \geq i + j + 1 \mid L \geq i\} \tag{3.18}$$

Let a_{pj2} be the number of Type 2 patients who request in block $i - p$ and are assigned to block $(i + j)$. Let $\hat{a}_{pj2}$ be the number of Type 2 patient who have not canceled the appointment by the current block and arrive at the assigned block. b_{pj3} and $\hat{b}_{pj3}$ can be defined for Type 3 patients in a similar manner. Assume that $\hat{a}_{pjt}$ and $\hat{b}_{pjt}$ are random variables from binomial distribution as shown below:

$$\hat{a}_{pj2} \sim \mathrm{B}\big\{a_{pj2}, \alpha_{pj2}\big\} \tag{3.19}$$

$$\hat{a}_{0j2} \sim \mathrm{B}\left\{\sum_{k=1}^{\hat{s}} x_{jk}, \alpha_{0j2}\right\} \tag{3.20}$$

$$\hat{b}_{pj3} \sim \mathrm{B}\big\{b_{pj3}, \alpha_{pj3}\big\} \tag{3.21}$$

$$\hat{b}_{0j3} \sim \mathrm{B}\left\{\sum_{t=1}^{\hat{w}} y_{jt}, \alpha_{0j3}\right\} \tag{3.22}$$

Before solving SIP-i, the values of α_{pj2} and α_{pj3} can be determined through statistic inference or forecasting. Then a_j, b_j need to be removed from (3.2b), and α_j in (3.16) need to be replaced with $\sum_p \hat{a}_{pj2} + \sum_p \hat{b}_{pj3}$.

3.1.4.4 Cancellations of Type 1 Patients

Liu, Ziya and Kulkarni [1] study the cancellation under far-in-advance scheduling policy where the cancellation is assumed to be handled on a daily basis at the beginning of each day. In this work, a new method is designed to handle the combination of daily cancellation and block-wise cancellation. For cancellations of Type 1 patients, let p in the triple (p, j, t) denote the number of days since the day when the patient sent the appointment request until today. Let D be the length between the request received day and the day when the patient decides to cancel the appointment.

Let θ_{pj1} be the probability that the patient with $(p, j, 1)$ who has not canceled the appointment until the current block will make the appointment of today. Assuming that the cancellation behavior of patients is independent of the appointment date, we have:

$$\theta_{pj1} = \text{Pr}\{L \geq i + j + 1, \text{arrive for appointment} \mid L \geq i, D \geq p\} \tag{3.23}$$

Let $\hat{\nu}_{pj1}$ be the number of Type 1 patients who make the visit in the assigned block, we have:

$$\hat{\nu}_{pj1} \sim \text{B}\big\{r_{i+j}, \theta_{pj1}\big\} \tag{3.24}$$

Then in SIP-i model, ν_j in (3.16) is replaced with $\sum_p \hat{\nu}_{pj1}$, a_j, b_j are removed from (3.2b), α_j in (3.16) is replaced with $\sum_p \hat{a}_{pj2} + \sum_p \hat{b}_{pj3}$. The information about ratio θ_{pj1} can be obtained through a study of historical data.

3.1.4.5 Punctuality of Type 1 and Type 2 Patients

If we relax the assumption about patient punctuality, and assume that patients may arrive earlier or later than the assigned block, the SIP-i model can be modified accordingly. According to [18], in relevant literature, the punctual arrival of patients is processed in the following ways: (1) empirical distribution of arrival time, (2) exponential distribution of arrival time, (3) allow only one block earliness or lateness. Contrasting to these methods, here it is assumed that the clinic only tolerates one-block lateness but can handle all early arrivals. It implies three situations:

- If the patients arrive earlier, they will still be served in their original assigned blocks. Their waiting time before the assigned block will not incur any cost.

- If the patients arrive more than one block behind, then they will be treated as walk-in patients. Their original appointments will be taken as no-shows.

- If the patients arrive at the immediate successive block of their original assigned block, they will be served in the block when they arrive.

To handle lateness, let $\boldsymbol{\gamma} := \{\gamma_j\}$ denote the number of patients who arrive at block j but was assigned to $j-1$. This uncertain data is associated with $r_{i+j-1} + a_{j-1}$. Let ι_1 be the probability that a patient will be late, ι_2 be the probability that a patient will be late by only one block. Then the probability that a patient can make the appointment at the assigned block is $1 - \iota_1$. If the cancellations analyzed above are considered, the following updates are needed:

$$\hat{a}_{pj2} \sim \mathrm{B}\big\{a_{pj2}, \alpha_{pj2}(1-\iota_1)\big\} \tag{3.25}$$

$$\hat{a}_{0j2} \sim \mathrm{B}\Big\{\sum_{k=1}^{\hat{s}} x_{jk}, \alpha_{0j2}(1-\iota_1)\Big\} \tag{3.26}$$

$$\hat{b}_{0j3} \sim \mathrm{B}\Big\{\sum_{t=1}^{\tilde{w}} y_{jt}, \alpha_{0j3}\Big\} \tag{3.27}$$

$$\hat{\nu}_{pj1} \sim \mathrm{B}\{r_{i+j}, \theta_{pj1}(1-\iota_1)\} \tag{3.28}$$

$$\gamma_j \sim \mathrm{B}\{r_{i+j-1} + a_{j-1}, \iota_2\} \tag{3.29}$$

In Step 2, the number of Type 1 and Type 2 patients that arrived at the current block but assigned to block $i-2$ or before are determined and added to the estimated walk-in patient number. In formula (3.31b) $\tilde{w}$ is used instead of $\hat{w}$ to distinguish this difference. Then (3.16) should be replaced with:

$$\eta_j := \sum_p \hat{\nu}_{pj1} + \sum_p \hat{a}_{pj2} + \sum_p \hat{b}_{pj3} + \gamma_j - \tau_j \tag{3.30}$$

Rest of the SIP-i model remains the same except for removing a_j, b_j from (3.2b). Value of ι_1 and ι_2 can be obtained through analysis of historical data.

3.2 Solve The SIP-i Model

3.2.1 Deterministic Equivalent Problem and Bound-Based Sampling Method

For a SP model with any discrete distributed random variable, the deterministic equivalent problem (DEP) can be derived. 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 $\tilde{\omega}$, the nth scenario has probability p_n. Then change the second-stage decision variables $\boldsymbol{q}, \boldsymbol{g}$ 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 below.

$$
\begin{aligned}
\text{(DEP-i)}\ \min \quad & -c_2 \textstyle\sum_{j=1}^{h} \sum_{k=1}^{\hat{s}} x_{jk} - c_3 \sum_{j=1}^{h} \sum_{t=1}^{\hat{w}} y_{jt} \\
& - \textstyle\sum_{n=1}^{N} p_n \Big(c_f \sum_{j=1}^{m} q_j^n + c_s \sum_{j=1}^{m} g_j^n \Big) && (3.31a) \\
\text{s.t.} \quad & \text{contraints } (3.1b) \text{ to } (3.1i) \\
& q_{j+1}^n - q_j^n - g_j^n = \eta(\omega)_j^n + b_j,\ j = 1, \cdots, h,\ n = 1, \cdots, N && (3.31b) \\
& q_1^n = \acute{q},\ n = 1, \cdots, N && (3.31c) \\
& q_{j+1}^n, g_j^n \in \mathbb{Z}^+, \quad j = 1, \cdots, h,\ n = 1, \cdots, N && (3.31d)
\end{aligned}
$$

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, assume that the arrival of Type 1 patients follows a Binomial Distribution described in Section 3.4.2, Type 2 patients have full attendance, and number of patients served in one block follows 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 the number of scenarios for block j is $r_j\tau'$ and

$$N \approx \prod_{j=1}^{h} (r_j \tau') \tag{3.32}$$

For example, let $\frac{l}{\xi} = 5$, then $\tau' = 8$, let $r_j = 2, j = 1, \cdots, 10$, $N \approx 16^{10}$. 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 sample average approximation (SAA) method is adopted 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, the lower bound of objective function value of the original DEP-i can be obtainted [92]. Let $\hat{f}^n_{\hat{N}}$ be the objective function value of the nth batch, then the lower bound is given by

$$L_{M\hat{N}} = \frac{1}{M}\sum_{n=1}^{m} \hat{f}^n_{\hat{N}} \tag{3.33}$$

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

$$U_{\bar{M}\bar{N}} = \frac{1}{\bar{M}}\sum_{n=1}^{\bar{M}} \bar{f}^n_{\bar{N}}(\hat{X}, \hat{Y}) \tag{3.34}$$

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 [92].

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 the lower bound and upper bound of the original problem can be utilized 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 (3.33), 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 (3.34) but an average level of upper bounds. $\bar{n}$ is the step length which indicates 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 $\hat{N} = n_2 + t \times \bar{n}$ ;
    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 $\hat{X}, \hat{Y}$ into objective function with $\bar{N}$ samples and $\bar{M}$ batches and
        obtain $U^n_{\bar{M}\bar{N}}$ using (3.34) ;
        get difference $d^n_{\hat{N}} = U^n_{\bar{M}\bar{N}} - \hat{f}^n_{\hat{N}}$;
    end
    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
        $\hat{N}(t)^* = n_2 + t \times \bar{n}$ ;
        $(X^*, Y^*) = \text{argmin}_{(\hat{X},\hat{Y})}\{f^n_{\hat{N}} \mid n = 1, \cdots, M, \hat{N} = \hat{N}(t)^*\}$;
    end
end
Output $\hat{N}(t)^*$ as a proper sample size;
```

The results from bound-based sampling method in Algorithm 1 are compared with those from some existing exterior and interior sampling methods. For exterior sampling method, in [46], the lower bound of sample size for ϵ-optimal solution to original problem with $1 - \alpha$ probability is:

$$N \geq \frac{3\sigma^2_{\max}}{(\epsilon - \delta)^2} \log \left(\frac{|\mathcal{S}|}{\alpha} \right) \tag{3.35}$$

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 $X, Y, \boldsymbol{a}, \boldsymbol{b}$. Since $\boldsymbol{a}, \boldsymbol{b}$ depend on X, Y, so the values of X, Y is under concern. 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, $s_i + w_i$ 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:

$$|\mathcal{S}| \leq 2^{m(s_i + w_i)} \tag{3.36}$$

Using this method, let $(\epsilon - \delta)^2 \approx 3$, $\alpha = 0.01$, we get $|\mathcal{S}| \approx 2^{250}$, so the lower bound 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 the clinic decides to accept all patients requested in block 1 and reject all patients in block 1 which is $(s + w)^2 = 625$. Under this situation, the lower bound of sample size is larger than $38 \times 625 = 23750$ which is still too large for CPLEX to handle. Therefore, the method in [46] does not fit DEP-i.

For interior sampling method, the method proposed by [49, 50] does not apply to the DEP-i model since it works on the basis of the stage-wise decomposition method of SP. So here sampling method in Algorithm 1 is only compared with the method from [48]. Figure 3.1 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 section, I 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. Values for the parameters are: $n_1 = 20, n_2 = 10, \bar{n} = 10, M = 20, \bar{M} = 2000, \bar{N} = 50, s = 12, w = 13, m = 10, \xi = 5, c_1 = c_2 = c_3 = c_4 = 1$. Unless specially mentioned, the computational experiments in the following context in this section will use this setting. For the method in [48], the step length is changed from 1 to 10 to keep it consistent with the sampling method. Their procedure after modification is illustrated in Algorithm 2.

Algorithm 2: Interior Sampling Algorithm for King and Wets [48]

```
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
```

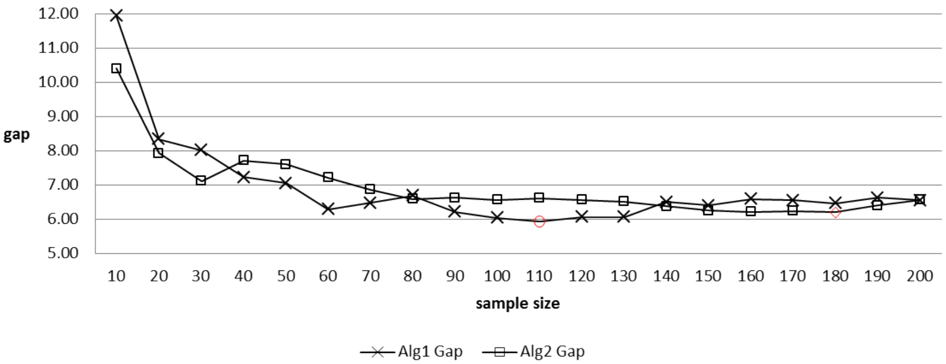


Figure 3.1: Comparison of Sampling Methods

Figure 3.1 shows the gap D_t of averaged bounds as calculated in Algorithm 1 and the averaged objective function value F_t as calculated in Algorithm 2 over different number of samples. It is obvious that the bound-based sampling method in the algorithm suggests implies a 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 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.

3.2.2 The Aggregate Assignment Method

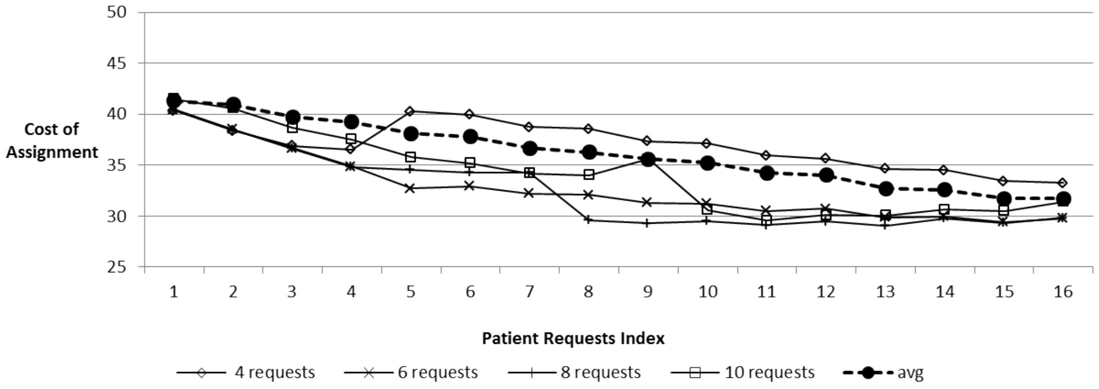
The aggregate assignment method distinguishes itself from the one-at-a-time assignment in [6] by the feature of scheduling multiple patients at the decision step in each block. The fundamental step of the aggregate assignment is to estimate how many same-day requests the clinic receives in each block. 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. Algorithm 3 shows how the aggregate assignment works using DEP-i. The one-at-a-time assignment procedure is shown in Algorithm 4.

Theoretically, from an overall perspective, the aggregate assignment makes a better

Algorithm 3: The Aggregate Assignment Method

```
Initialization: $\acute{q}^i = 0, \beta^i = 0, \forall i = 1, \cdots, m$ , choose value for $M, \hat{N}, \bar{M}, \bar{N}$ where
  $\bar{M} \gg M$;
for $i = 1, \cdots, m$ do
    Step 1: the current block is block $i$, $\acute{q}^i = q^{i-1}_{m+1}$;
    Step 2: estimate $\hat{s}$ and $\hat{w}$ according to Section 3.3.1 or other prediction methods;
    Step 3: choose proper sample size and run DEP-i model following Algorithm 1,
    obtain the optimal assignment solution;
    Step 4: assign received requests one by one following the optimal assignment
    solution, update $\bar{\boldsymbol{a}}, \bar{\boldsymbol{b}}$ accordingly;
    Step 5:
    if number of requests go beyond the estimated value then
        for each additional request do
            set DEP-i model for one request;
            implement Step 3 and assign the request using the obtained solution ;
            update $\bar{\boldsymbol{a}}, \bar{\boldsymbol{b}}$ accordingly;
        end
    end
    Step 6: when all the requests of the current block are handled, update $\beta^i$,
    $i = i + 1$ go to Step 1;
end
```

Figure 3.2: Aggregate Underestimation Costs vs. Average One-at-a-time Costs



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 as-

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

```
Initialization: $\acute{q}^i = 0, \beta^i = 0, \forall i = 1, \cdots, m$ , choose value for $M, \hat{N}, \bar{M}, \bar{N}$ where $\bar{M} \gg M$;
for $i = 1, \cdots, m$ do
    Step 1: the current block is block $i$, $\acute{q}^i = q^{i-1}_{m+1}$;
    Step 2: for each request do
        set DEP-i model for this request properly ;
        choose proper sample size and run DEP-i model following Algorithm 1, obtain the optimal assignment solution;
        assign following the result of the solution, update $\bar{\boldsymbol{a}}, \bar{\boldsymbol{b}}$ accordingly;
    end
    Step 3: when all the requests of the current block are handled, update $\beta^i$, $i = i + 1$ go to Step 1;
end
```

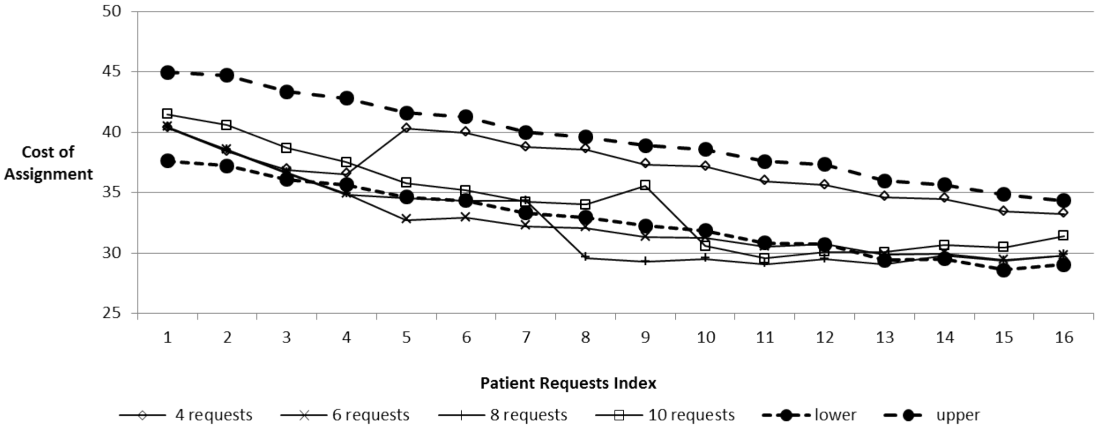
Table 3.1: Person-wise Assignment Cost Comparison

Probabilities	Under				Over
	4 requests	6 requests	8 requests	10 requests	16 requests
< Avg	0.25	1	1	0.9375	0.9375
< Lower Bnd	0	0.5	0.5625	0.1875	0.6875
> Upper Bnd	0	0	0	0	0

signment method is demonstrated through the following experiment. In the experiment, the two assignment methods under two cases of accuracy of request estimation are compared: underestimation and overestimation. Table 3.1 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 it is easy to see that when the estimated request is smaller than the real number of requests, the clinic needs 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

Figure 3.3: Aggregate Underestimation Costs vs. 95% C.I. of One-at-a-time Costs



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 3.1 shows the percentage of the assignment costs obtained from Algorithm 3 less than or greater than the average costs or bounds obtained from Algorithm 4. Figures 3.2 and 3.3 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, assume that the real number of requests is 16, and the estimation of requests ranges from 16 to 20. The DEP-1 model with one-at-a-time mode is run 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, the DEP-1 model is run

Figure 3.4: Aggregate Overestimation Costs vs. Average One-at-a-time Costs

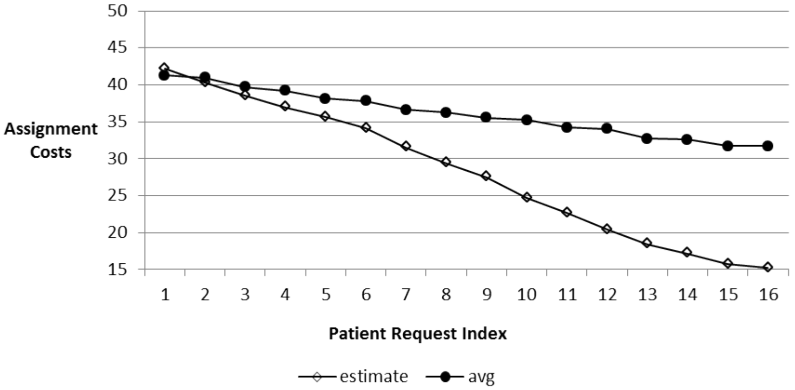
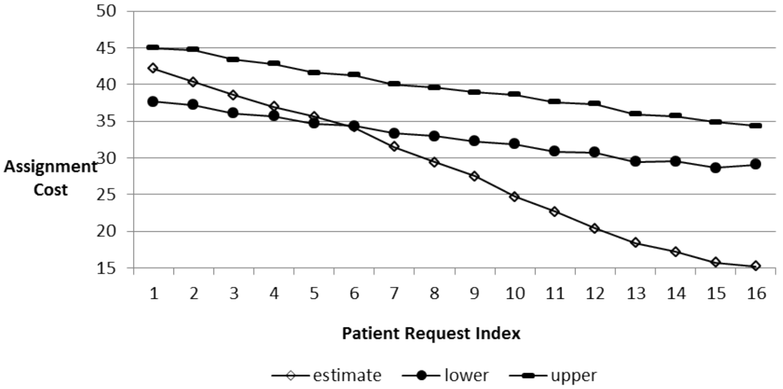


Figure 3.5: Aggregate Overestimation Costs vs. 95% C.I. of One-at-a-time Costs



under 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 3.1 as well as Figures 3.4 and 3.5 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 3.5, it can be seen 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.

3.3 Sensitivity Analysis and Value of SIP model

3.3.1 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 3.2 to 3.5. 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, the estimated requests $\hat{s}, \hat{w}$ of the first block are set to the same value increasing from 5 to 43, so the total number of requests goes from 10 to 86. Then solve DEP-1 for each of the estimations with 20 batches each and take average results. Figures 3.6 to 3.8 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. This bowl shape can be interpreted 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 3.8 it is easy to observe 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 in the 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: $c_1 = c_2 = c_3 = 1, \tau \sim$ Poisson$(5),\ m = 10,\ r_i \sim$ uniform$(0, 5)$, if the estimation falls in the second interval [34,68], then a low level overall cost can be guaranteed.

Figure 3.6: Objective Value and Assigned Requests of DEP-1 with Different Request Estimations

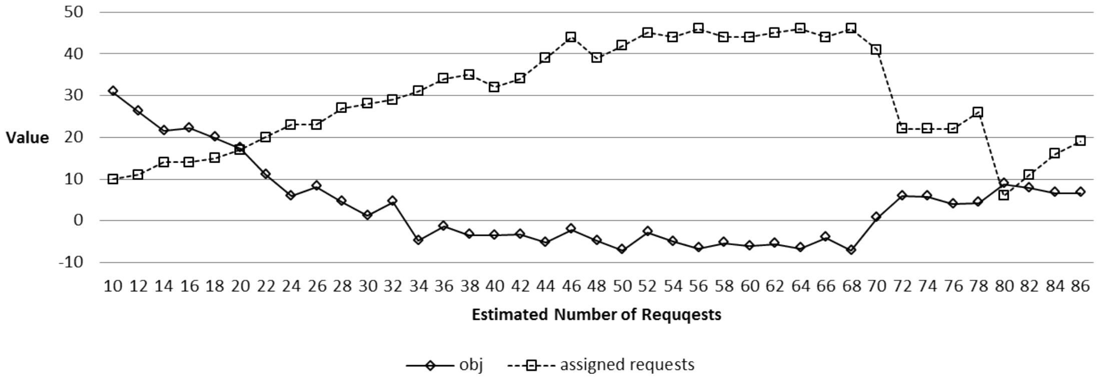


Figure 3.7: Objective Value and Overflow Cost ($\boldsymbol{q}$) of DEP-1 with Different Request Estimations

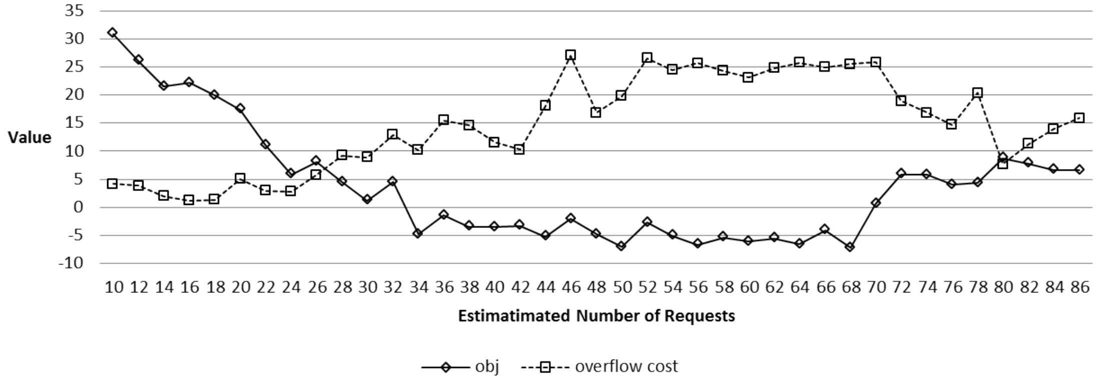
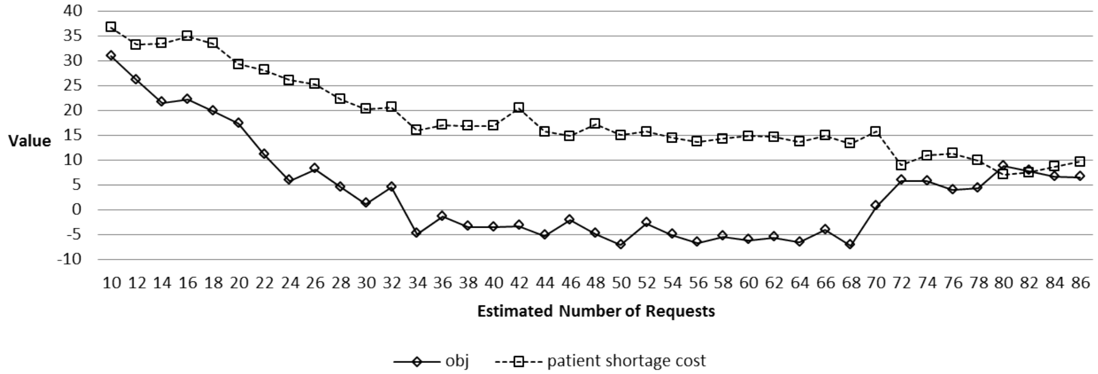


Figure 3.8: Objective Value and Patient Shortage Cost ($\boldsymbol{g}$) of DEP-1 with Different Request Estimations



3.3.2 Further Sensitivity Analysis

Besides the number of requests, the 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 six factors are chosen 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 3.2. Since the objective function coefficient is involved as a factor, the magnitude of the objective function is no longer a proper metric. So the components of the objective function are utilized: the average number of assigned request, the average sum of $\boldsymbol{q}$ and the average sum of $\boldsymbol{g}$, as the metrics for the evaluation. The third to eighth columns of Table 3.2 present the ranks of the impact of factors under the metrics using the entropy-based analysis addressed in [24]. 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. Columns 3 to 5 are the results obtained under the distribution of $\tau \sim$ Poisson$(\frac{l}{\xi})$, and Columns 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 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 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 3.2. 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 remain consistent under the two different distributions.

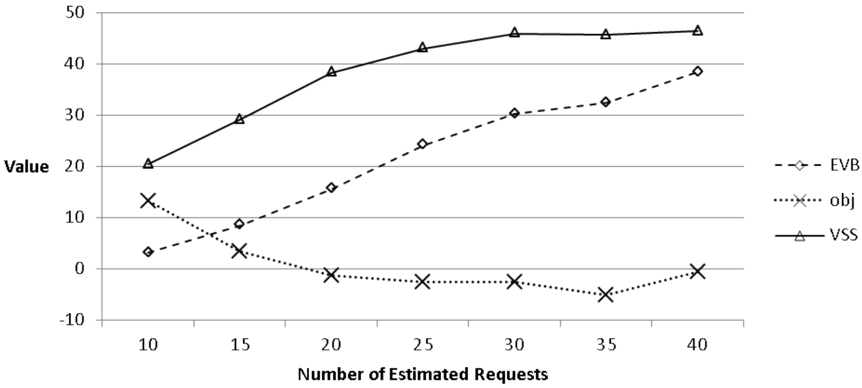
Table 3.2: Rank of Importance of Parameters under Different Distributions of τ

Factors	Levels	Poisson			Uniform		
		$\sum\sum X + \sum\sum Y$	$\sum \boldsymbol{q}$	$\sum \boldsymbol{g}$	$\sum\sum X + \sum\sum Y$	$\sum \boldsymbol{q}$	$\sum \boldsymbol{g}$
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

3.3.3 Value of SP Model

One highlighted feature of stochastic programming is to make decision involving data uncertainty. The trade-off between gaining more information before a decision and incurring less cost on information leads to a question: how much we know about the uncertainty is sufficient for a good decision? Stochastic programming practitioners embrace the Expected Value of Perfect Information (EVPI) and the Value of the Stochastic Solution (VSS) to measure information cost and benefit. EVPI is the difference in objective value between solution with complete information and the current SP solution with partial information [75]. VSS is the the difference in objective value between the current SP solution and the expected value problem solution [75]. Since the sample space of DEP-i model is too large to be calculated, so it is hard to get EVPI. Instead of EVPI, the Expected Value of Better Information (EVBI) can be used. To obtain EVBI, we draw a large sample with large batches to approximate the solution of full sample space. For each of the sample, the DEP-i model is run and take the sample mean of the optimal objective function values, this mean value is named as “Wait-and-See" (WS) value [75]. Let the average optimal objective function value obtained by solving DEP-i with batches of 110 samples be the “Here-and-Now" (HN) value [75]. Then we have EVBI = HN - WS. To obtain VSS, plug in expectation value of the random variables into DEP-i, and obtain the expected optimal solution. After that, we put this solution into the batches of 110 samples drawn for HN, and calculate the corresponding objective value for each of the scenario using expected optimal solution. The average value of the objectives is the “Expected Results using Expected Value" (EEV) [75]. Then we have VSS = EEV - HN. Figure 3.9 presents the EVBI and VSS for different request estimations. Apparently, in most cases, both EVBI and VSS are more than double of the objective function value. This manifests the value of the current SIP solution in delivering useful result efficiently. It is obvious that the VSS is always higher than EVBI which demonstrates that if the uncertainty of data is ignored, a solution with a significant difference from the current SIP model may be obtained, while if better information are purchased with higher cost, the deviation of the results from the current model is not significant. Hence the current model is a better choice than WS in saving calculation cost, and is better than EEV in obtaining better solution. Increase of VSS is relatively stable compared with the convex trend of EVBI as the increment of number of estimated requests. This observation implies that the cost of utilizing better information will increase faster if the model size becomes larger. The gap between the two values stays almost constant at low request number, then starts to decrease after request of 30. This implies that the SIP model can give a more valuable solution compared with first-order statistics (the expected value problem), and the advantage of SIP model over expected value problem may reach a stable status beyond a certain model size.

Figure 3.9: EVBI and VSS for Different Request Estimations



3.4 Conclusions of the Section

This section 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, the bound-based sampling method is developed 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, sensitivity analysis is performed on a few parameters and settings which in practice can offer meaningful suggestions for clinic cost control as well as key factor identification and m onitoring. The advantage of the SIP model over the first-order statistics model is demonstrated through entropy analysis for different distributions of τ. Demonstration of value of the SIP model is also enhanced through VSS and EVBI value evaluation.

4. OUTPATIENT CLINIC ONLINE SCHEDULING: STOCHASTIC INTEGER PROGRAMMING WITH ENDOGENOUS UNCERTAINTY

4.1 Introduction

In the outpatient clinic online scheduling problem, patient assignment is planned before actual patient attendance. The decision is made without full information regarding how many patients will make the appointments on time, and how many patients with appointments will arrive at the clinic and will be served in a certain time interval. This Modified L-shaped formulates the clinic scheduling problem in a two-stage stochastic integer programming with endogenous uncertainty. The endogenous uncertainty arises from the dependence of uncertain information on the decision, which in this problem is reflected by the fact that the assigned number of patients always gives an upper bound on the number of patients that arrive with appointments. Modified L-shaped and aggregated multicut L-shaped methods are designed to solve the model. Based on the computational experiments, the aggregated multicut L-shaped method needs less number of iterations than the modified L-shaped method to achieve the same optimal solution. Distinctive optimality cut generation schemes are proposed under three types of distributions for decision dependent random variables; namely Poisson distribution, discrete uniform distribution and empirical distribution. The first two types of distributions are handled with explicit form of population mean, while the empirical distribution is handled via the central limit theorem. It is shown in computational experiments that the optimality cuts generated through central limit theory and those cuts generated with explicit distribution lead to similar optimal objective costs. These methods offer generic resolution for decision-uncertainty relationships in SP with the following features: 1) decision variables act as parameters in the random variable distributions and 2) decision variables decide the population mean of the random variables. An alternative formulation of the problem with simple recourse function is provided, based on which, a mixed integer programming model is established as a convenient method to obtain an approximation of the stochastic integer programming model. This approximation method is applicable to any stochastic programming problem with endogenous uncertainty and simple recourse function.

In this section, most assumptions in previous section are adopted expect that Type 2 patients have no-show rates. In the two-stage SIP model, the first stage makes decisions about assignment with respect to patient preferences and FCFS rule. The second stage evaluates the consequence of the first-stage decision based on the estimation of resulted overflow and patient shortage cost. It is obvious that the assigned number of patients will affect the number of patients arrived with appointments. To be precise, number of Type 2 patients that are assigned in each block is a decision variable, if the no-show probability of Type 2 patients is considered, then the number of Type 2 patients who will make the visits becomes a random variable depending on the decision variable. This type of uncertainty where a random variable depends on decision variable is called *endogenous uncertainty* [15]. In this problem, the decision variable impacts the upper bound of the random variable, which means that the number of Type 2 patients who attend the appointment in a block cannot exceed the number of Type 2 patients who are assigned to this block. Although stochastic programming (SP) with endogenous uncertainty is well studied in the literature, relevant papers are designed for their own specified types of endogenous uncertainty which are not the same as in this section. Therefore, no existing method can be applied to handle the uncertainty in this scheduling problem. Two sets of two-stage SIP formulations for the clinic scheduling problem are developed. In the first set of formulations, different situations about the decision dependent random distributions are considered. A modified L-shaped algorithm and an aggregated multicut L-shaped algorithm to solve the SIP model are devised. Both algorithms provide generic ways to solve SP problems with decision dependent distribution parameters, such as mean. In the second set of formulations, the second stage is modeled as a simple recourse problem. The lower bound of the second stage are derived leveraging population means and sample means of random variables. Using the bound, a mixed-integer programming model combining the first-stage and second-stage models is constructed. The objective function value of the model can be used as an approximation of expected result of using the expected value solution (EEV).

The remainder of this section is arranged in the following way: followed by the properties and formulations of the two-stage SIP model for clinic scheduling in Section 2. Section 3 presents the modified L-shaped method and the aggregated multicut L-shaped method. After that, numerical examples are provided for comparison of the methods. Different distributions of the random variables and adaptive solution methods are also discussed in Section 3. Section 4 provides alternative two-stage SIP formulations with simple recourse function. The last section is about the conclusions.

4.2 Problem Formulations

4.2.1 Decision Model

4.2.1.1 Formulations

Assumptions of the model in this section are the same as previous section except for that no-show rates of Type 2 patients are considered here. SIPE-i below has the same first-stage formulations as Section 3. The second-stage constraints in Equations (4.2b) to (4.2e) are different from previous section.

$$\text{(SIPE-i)} \quad \min \; -c_2 \textstyle\sum_{j=1}^{h} \sum_{k=1}^{\hat{s}} x_{jk} - c_3 \sum_{j=1}^{h} \sum_{t=1}^{\hat{w}} y_{jt} + \mathbb{E}[Q(X, Y, \boldsymbol{a}, \boldsymbol{b}, \omega)] \tag{4.1a}$$

$$\text{s.t.} \quad \textstyle\sum_{j=1}^{h} x_{jk} \leq 1, \; \forall k = 1, \cdots, \hat{s} \tag{4.1b}$$

$$\textstyle\sum_{j=1}^{h} y_{jt} \leq 1, \; \forall t = 1, \cdots, \hat{w} \tag{4.1c}$$

$$x_{jk} \leq A_{jk}, \quad j = 1, \cdots, h, \; k = 1, \cdots, \hat{s} \tag{4.1d}$$

$$y_{jt} \leq B_{jt} z_t, \quad j = 1, \cdots, h, \; t = 1, \cdots, \hat{w} \tag{4.1e}$$

$$a_j = \bar{a}_{j+i} + \textstyle\sum_{k=1}^{\hat{s}} x_{jk}, \quad j = 1, \cdots, h \tag{4.1f}$$

$$b_j = \bar{b}_{j+i} + \textstyle\sum_{p=1}^{\hat{w}} y_{jt}, \quad j = 1, \cdots, h \tag{4.1g}$$

$$\textstyle\sum_{t=1}^{\hat{w}} (i + j) y_{jt} \geq \beta^{i-1} \sum_{p=1}^{\hat{w}} z_t, \quad j = 1, \cdots, h, \quad t = 1, \cdots, \hat{w} \tag{4.1h}$$

$$x_{jk}, y_{jt} \in \{0, 1\}, \quad j = 1, \cdots, h, \quad k = 1, \cdots, \hat{s}, \quad t = 1, \cdots, \hat{w} \tag{4.1i}$$

$$a_j, b_j \in \mathbb{Z}^+, \quad j = 1, \cdots, h, \quad k = 1, \cdots, \hat{s}, \quad t = 1, \cdots, \hat{w} \tag{4.1j}$$

where

$$Q(X, Y, \boldsymbol{a}, \boldsymbol{b}, \omega) \quad = \quad \min \quad c_f \textstyle\sum_{j=1}^{h} q_j + c_s \sum_{j=1}^{h} g_j \tag{4.2a}$$

$$\text{s.t.} \quad q_{j+1} - q_j \geq \eta(\omega)_j + \alpha_j + b_j, \quad j = 1, \cdots, h \tag{4.2b}$$

$$g_j + q_j \geq -\eta(\omega)_j - \alpha_j - b_j, \quad j = 1, \cdots, h \tag{4.2c}$$

$$q_1 = \acute{q} \tag{4.2d}$$

$$q_j, g_j \in \mathbb{Z}^+, \quad j = 1, \cdots, h \tag{4.2e}$$

Objective function (4.1a) 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. In order to have less accumulative overflows, the optimization solver may give a solution with more assignments to the last block. This will lead to a large "overtime" cost which is usually calculated as the difference between the real finish time of a clinic day and the end time of the last block. The total number of patients assigned to the last block can also explain the “overtime”, since the more patients are assigned to the last block, the later the clinic day will be finished. So the clinic can control the number of patients assigned to the last block in order to reduce the “overtime” cost. Let c_o be the unit overtime cost of assigning one patient to the last block, then the objective function can be written as:

$$\min \quad -c_2 \sum_{j=1}^{h} \sum_{k=1}^{\hat{s}} x_{jk} - c_3 \sum_{j=1}^{h} \sum_{t=1}^{\hat{w}} y_{jt} + c_o(a_h + b_h) + \mathbb{E}[Q(X, Y, \boldsymbol{a}, \boldsymbol{b}, \omega)] \tag{4.3}$$

where $\mathbb{E}[Q(X, Y, \boldsymbol{a}, \boldsymbol{b}, \omega)]$ is the expectation of the second-stage objective function value.

Second-stage objective function (4.2a) 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. Constraints (4.2c) and (4.2d) are 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, p_j, α_j, b_j, the number of patients 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:

$$q_{j+1} = \max\{\nu_j + q_j + \alpha_j + b_j - \tau_j, 0\} \tag{4.4}$$

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

$$g_j = \max\{\tau_j - \nu_j - q_j - \alpha_j - b_j, 0\} \tag{4.5}$$

Constraint (4.2b) and (4.2c) are derived through these two equations and the definition of η_j, i.e.

$$\eta_j := \nu_j - \tau_j \tag{4.6}$$

4.2.2 Properties of SIPE-i

The SIPE-i model in this section has some interesting properties which facilitate obtaining the optimal solution.

- Property I: the SIPE-i model has complete recourse.
- Property II: the coefficient matrix of the second-stage model is totally unimodular.
- Property III: the second-stage model can be reformulated as a simple integer recourse.

Proving Property I is trivial since for any $\boldsymbol{\eta}(\omega)+\boldsymbol{\alpha}+\boldsymbol{b} \in \mathbb{Z}^h$, there alway exist vectors $\boldsymbol{q}, \boldsymbol{g}$ such that (4.2b) to (4.2e) are satisfied. With this property, no feasibility cuts are needed for the L-shaped method. Proof of Property II is illustrated below. Property III is addressed later in Section 4.4 of this section.

Proof:

The coefficient matrix is like:

$$\begin{array}{ccccccccccc}
q_1 & q_2 & q_3 & \cdots & q_{n-1} & q_n & q_{n+1} & g_1 & g_2 & \cdots & g_n \\
-1 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & -1 & 1 & \cdots & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & \cdots & 0 & -1 & 1 & 0 & 0 & \cdots & 0 \\
1 & 0 & 0 & \cdots & 0 & 0 & 0 & 1 & 0 & \cdots & 0 \\
0 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 & 1 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \cdots \\
0 & 0 & 0 & \cdots & 0 & 1 & 0 & 0 & 0 & \cdots & 1
\end{array} \tag{4.7}$$

This coefficient matrix can be decomposed in the following way :

$$\begin{array}{c|c|c}
\multicolumn{2}{c|}{D^{n\times(n+1)}} & 0^{n\times n} \\
\hline
I^{n\times n} & 0^{n\times 1} & I^{n\times n}
\end{array} \tag{4.8}$$

where

$$D^{n\times(n+1)} = \begin{pmatrix}
-1 & 1 & 0 & \cdots & 0 & 0 & 0 \\
0 & -1 & 1 & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \\
0 & 0 & 0 & \cdots & 0 & -1 & 1
\end{pmatrix} \tag{4.9}$$

D has only two non-zero entries 1 and -1 in each column, all its sub squared nonsingular matrices have only two structures:

- all -1 diagonals with all 0 lower triangular
- all 1 diagonals with all 0 upper triangular

For both structures the determinant of the sub nonsingular matrices are either -1 or 1. So D is a totally unimodular. Since the coefficient matrix is a combination of D and identity matrices and all 0 matrices, so the coefficient matrix is also totally unimodular. (The end of proof)

Property II is very useful because if integral restrictions of the second stage model is relaxed, it can still get integral optimal solutions. This is an advantage for the following reason: in the L-shaped algorithm, the dual solution of the second stage is necessary in generating optimality cut for the master problem. The cut can be generated only if the second-stage model follows the duality theorem of linear programming. Without Property II, the linear relaxation of the second-stage can not retain all the information of the original problem. With Property II, the dual solution of the relaxation of the second-stage model can be directly used in the optimality cut.

4.3 Modified L-shaped Method and Aggregated Multicut L-shaped Method for Endogenous Uncertainty

4.3.1 Modified L-shaped Method for Endogenous Uncertainty

The first-stage decision variable $\boldsymbol{a}$ and the second-stage random variable $\boldsymbol{\alpha}$ introduce endogenous uncertainty into the problem since the distribution of uncertain random variable vector $\boldsymbol{\alpha}$ depends on vector $\boldsymbol{a}$. Let the random variables which are associated with endogenous uncertainty be the “Linked Variables", the random variables which are irrelevant to endogenous uncertainty be the “Unlinked Variables". This section discusses the solution method for the decision model (SIPE-i) with an additional assumption: the distribution of the linked variables are not empirical. Recall that $\boldsymbol{\alpha}$ denotes the vector of numbers of patients that will arrive with appointments for the remaining blocks. In relevant literature, binomial distribution and discrete uniform distribution for patient arrivals are widely used [6, 18]. Solution methods for these two distributions are demonstrated accordingly in this section. N samples are drawn from the sample space of $\eta_j(\omega)$, let the nth scenario be $\eta_j(\omega)^n$. suppose there are K different scenarios for $\boldsymbol{\alpha}$. The value of kth scenario α_j^k comes from integer points in $[0, a_j]$, Therefore, there are NK scenarios in this scheduling problem with respect to vector $\boldsymbol{a}$. The second-stage model with indices k and n is:

$$
\begin{aligned}
(\mathrm{sub}_{n,k}) \quad f_{n,k} \quad = \quad & \min \quad c_f \textstyle\sum_{j=1}^h q_j^{n,k} + c_s \sum_{j=1}^h g_j^{n,k} && \text{(4.10a)} \\
\text{s.t.} \quad & q_{j+1}^{n,k} - q_j^{n,k} \geq \eta(\omega)_j^n + \alpha_j^k + b_j, \quad j = 1, \cdots, h && \text{(4.10b)} \\
& g_j^{n,k} + q_j^{n,k} \geq -\eta(\omega)_j^n - \alpha_j^k - b_j, \quad j = 1, \cdots, h && \text{(4.10c)} \\
& q_1^{n,k} = \acute{q} && \text{(4.10d)} \\
& q_j^{n,k}, g_j^{n,k} \in \mathbb{Z}^+, \quad j = 1, \cdots, h && \text{(4.10e)}
\end{aligned}
$$

Define $\lambda_j^{n,k}$ as the dual variable associated with jth overflow constraint in (4.10b) and $\mu_j^{n,k}$ as the dual variable associated with jth patient shortage constraint in (4.10c). According to Weak Duality Theorem, the primal sub problem defines a upper bound of the dual sub problem as illustrated below:

$$
\begin{aligned}
f_{n,k} \geq & \max \sum_{j=1}^{h} [(\eta_j^n + \alpha_j^k + b_j)\lambda_j^{n,k} + (-\eta_j^n - \alpha_j^k - b_j)\mu_j^{n,k}] \\
= & \max \sum_{j=1}^{h} [\eta_j^n(\lambda_j^{n,k} - \mu_j^{n,k}) + \alpha_j^k(\lambda_j^{n,k} - \mu_j^{n,k}) + b_j(\lambda_j^{n,k} - \mu_j^{n,k})]
\end{aligned}
\tag{4.11}
$$

Let λ_j^*, μ_j^* be the optimal solution of the dual problem associated with scenario n, k, p_n be the probability of scenario n of η_j, $\tilde{p}_k$ be the probability of kth scenario of $\boldsymbol{\alpha}$. Define

$u = E[f_{n,k}]$, the following inequality can be derived:

$$\begin{aligned} u &\geq \mathbb{E}\Big[\sum_{j=1}^{h}[\eta_j^n(\lambda_j^* - \mu_j^*) + \alpha_j^k(\lambda_j^* - \mu_j^*) + b_j(\lambda_j^* - \mu_j^*)]\Big] \\ &= \sum_{n=1}^{N}\sum_{k=1}^{K} p_n \tilde{p}_k \sum_{j=1}^{h}[\eta_j^n(\lambda_j^* - \mu_j^*) + \alpha_j^k(\lambda_j^* - \mu_j^*) + b_j(\lambda_j^* - \mu_j^*)] \end{aligned} \tag{4.12}$$

Using sampling method, p_n is replaced with $\frac{1}{N}$. If it is assumed that α_j follows binomial distribution with individual no-show probability p_2, i.e. $\alpha_j \sim$ binomial$(a_j, 1 - p_2)$, then in the inequalities above, the sample mean of α_j is replaced with the population mean $a_j(1 - p_2)$, so that a_j can be used as a first-stage variable in the optimality cut shown in (4.12). The cut has the formulation as:

$$\begin{aligned} u - \tfrac{1}{N}\textstyle\sum_{n=1}^{N}\sum_{k=1}^{K}\tilde{p}_k\sum_{j=1}^{h} b_j(\lambda_j^* - \mu_j^*) - \tfrac{1}{N}\sum_{n=1}^{N}\sum_{j=1}^{h} a_j(1 - p_2)(\lambda_j^* - \mu_j^*) \\ \geq \tfrac{1}{N}\textstyle\sum_{n=1}^{N}\sum_{k=1}^{K}\tilde{p}_k\sum_{j=1}^{h}\eta_j^n(\lambda_j^* - \mu_j^*) \end{aligned} \tag{4.13}$$

If it is assumed that α_j follows discrete uniform distribution in $[0, a_j]$, i.e. $\alpha_j \sim$ unif$(0, a_j)$, then the sample mean of α_j can be replaced with population mean $\frac{1}{2}a_j$, then the explicit form of the cut becomes:

$$\begin{aligned} u - \tfrac{1}{N}\textstyle\sum_{n=1}^{N}\sum_{k=1}^{K}\tilde{p}_k\sum_{j=1}^{h} b_j(\lambda_j^* - \mu_j^*) - \tfrac{1}{N}\sum_{n=1}^{N}\sum_{j=1}^{h}\tfrac{1}{2}a_j(\lambda_j^* - \mu_j^*) \\ \geq \tfrac{1}{N}\textstyle\sum_{n=1}^{N}\sum_{k=1}^{K}\tilde{p}_k\sum_{j=1}^{h}\eta_j^n(\lambda_j^* - \mu_j^*) \end{aligned} \tag{4.14}$$

The master problem with optimality cuts is presented below. Here only one cut with all NK scenarios is added to the master problem per iteration. Modified L-shaped method with aggregated multicuts is addressed in the next section.

$$\begin{aligned} &\text{(Master)} \quad \min \quad -c_2 \textstyle\sum_{j=1}^{h} \sum_{k=1}^{\hat{s}} x_{jk} - c_3 \sum_{j=1}^{h} \sum_{t=1}^{\hat{w}} y_{jt} + u && \text{(4.15a)} \\ &\qquad\qquad\quad \text{s.t.} \quad \text{All first-stage Constraints in (4.1b) to (4.1j)} && \text{(4.15b)} \\ &\qquad\qquad\qquad\qquad \text{Optimality cuts in (4.13) or (4.14)} && \text{(4.15c)} \end{aligned}$$

Based on the master and subproblems, a modified L-shaped method aiming at solving SIP with Endogenous Uncertainty is proposed. Algorithm 5 describes the procedure of the method. This method distinguishes itself from regular L-shaped method for SP in two aspects:

- It divides the random variables in the second stage into two subsets: linked variables and unlinked variables, draw samples from the two subsets separately and independently.
- It replaces the sample mean of the linked variables with population mean so that the linked decision variables appear in the cut formulation.

Initial solution $\boldsymbol{a}_0, \boldsymbol{b}_0$ can be obtained by ignoring the no-shows of Type 2 patients and solving the deterministic equivalent program (DEP) for the resulting regular two-stage SIP. Due to the relative complete recourse property of the problem, the feasibility cut is not considered in this method.

4.3.2 Aggregated Multicut L-shaped Method for Endogenous Uncertainty

Based on the division of linked variables and unlinked variables and the modified L-shaped method in Algorithm 5, the aggregated multicut L-shaped method is proposed. This method aggregates K scenarios of the linked variables and generates one cut for each scenario of the unlinked variables.

Algorithm 5: Modified L-shaped Method for Endogenous Uncertainty

```
**Initialization**: set iteration index $d = 0, lb = \infty, ub = \infty, \epsilon = 0.001$ ;
get initial solution $\boldsymbol{a}_0, \boldsymbol{b}_0$;
let $F^d$ be the objective value of master problem, $u^d$ be the optimal value of $u$;
**Step 1:**
$d = d + 1$;
**for** *each scenario* $n = 1, , N$ **do**
    draw $K$ samples from $\boldsymbol{a}_{d-1}$ **for** *each value of* $\boldsymbol{\alpha}_k, k = 1, \cdots, K$ **do**
        solve the subproblem, let $f_d^{n,k}$ be the objective value of the subproblem with
        scenario $n, k$ at iteration $d$ ;
    **end**
**end**
generate optimality cut in (4.15c);
compute upper bound $ub = \min\{F^{d-1} - u^{d-1} + \sum_{n=1}^{N} \sum_{k=1}^{K} p_n \tilde{p}_k f_d^{n,k}, ub\}$;
if the upper bound is updated, update the incumbent solution;
**Step 2:**
add the optimality cut to Master with Uni-cut and solve the master problem;
set lower bound $lb = \max\{F^d, lb\}$;
**Step 3:**
**if** $ub - lb < \epsilon|ub|$ **then**
    stop, the current solution of the master problem is $\epsilon$ - optimal;
**else**
    return to Step 1;
**end**
```

The sub problems remain the same as in (4.10a) to (4.10e). For the master problem, let $\boldsymbol{u} := \{u_n\}, n = 1, \cdots, N$ be the decision variable vector associated with N scenarios of the unlinked variables, $\lambda_j^{n*}, \mu_j^{n*}$ be the optimal dual solution for the nth scenario. For scenario n the sub dual problem satisfies:

$$u_n \geq \sum_{k=1}^{K} p_n \tilde{p}_k \sum_{j=1}^{h} [\eta_j^n(\lambda_j^{n*} - \mu_j^{n*}) + \alpha_j^k(\lambda_j^{n*} - \mu_j^{n*}) + b_j(\lambda_j^{n*} - \mu_j^{n*})] \tag{4.16}$$

If α_j follows binomial distribution, the cut for scenario n is:

$$\begin{aligned} u_n - \textstyle\sum_{k=1}^{K} \tilde{p}_k \sum_{j=1}^{h} b_j(\lambda_j^{n*} - \mu_j^{n*}) - \sum_{j=1}^{h} a_j(1-p_2)(\lambda_j^{n*} - \mu_j^{n*}) \\ \geq \textstyle\sum_{k=1}^{K} \tilde{p}_k \sum_{j=1}^{h} \eta_j^n(\lambda_j^{n*} - \mu_j^{n*}) \end{aligned} \tag{4.17}$$

If it is assumed that α_j follows discrete uniform distribution, the cut for scenario n is:

$$\begin{aligned} u_n - \textstyle\sum_{k=1}^{K} \tilde{p}_k \sum_{j=1}^{h} b_j(\lambda_j^{n*} - \mu_j^{n*}) - \sum_{j=1}^{h} \frac{1}{2}a_j(\lambda_j^{n*} - \mu_j^{n*}) \\ \geq \textstyle\sum_{k=1}^{K} \tilde{p}_k \sum_{j=1}^{h} \eta_j^n(\lambda_j^{n*} - \mu_j^{n*}) \end{aligned} \tag{4.18}$$

$$\begin{aligned} \text{(Master with multicut)}\ \min\ & - c_2 \sum_{j=1}^{h}\sum_{k=1}^{\hat{s}} x_{jk} - c_3 \sum_{j=1}^{h}\sum_{t=1}^{\hat{w}} y_{jt} + \sum_{n=1}^{N} u_n \\ \text{s.t.}\ & \text{All first-stage Constraints in (4.1b) to (4.1j)} \\ & \text{Optimality cuts in (4.17) or (4.18)}\ , n = 1, \cdots, N \end{aligned} \tag{4.19}$$

The aggregated multicut L-shaped method is described in Algorithm 6.

4.3.3 Comparison of Modified L-shaped Method and Aggregated Multicut L-shaped Method

In this section, a simplified numerical example is adopted to compare the performance of the two proposed methods. In this numerical example, let the number of remaining blocks $h = 2$ or 3 or 5, length of each block is $l = 30$ min, estimated number of Type 2 requests received the current block is $\hat{s} = 1$ to 20, estimated number of Type 3 patients arrived at the current block is $\hat{w} = 2$ to 20, number of patients overflow to the current block is $\acute{q} = 1$ and all the objective coefficients are set to 1. Assume that there are two scenarios of the random variable $\boldsymbol{\eta}(\omega)$, each of them are equally likely to occur. The first scenario is each of the remaining block has $\nu_j = 1$ Type 1 patient who will arrive for service. The second scenario is $\nu_j = 2$. In this numerical example, patients have no

Algorithm 6: Aggregated Multicut L-shaped Method for Endogenous Uncertainty

```
Initialization: set iteration index $d = 0, lb = \infty, ub = \infty, \epsilon = 0.001$ ;
get initial solution $\boldsymbol{a}_0, \boldsymbol{b}_0$, let $u_n = -\infty, n = 1, \cdots, N$;
let $F^d$ be the objective value of master problem with multicut , $\boldsymbol{u}^d$ be the optimal
  value of $\boldsymbol{u}$;
Step 1:
$d = d + 1$;
for each scenario $n = 1, \cdots, N$ do
    draw $K$ samples from $\boldsymbol{a}_{d-1}$ for each value of $\boldsymbol{\alpha}_k, k = 1, \cdots, K$ do
        solve the subproblem, let $f_d^{n,k}$ be the objective value of the subproblem with
        scenario $n, k$ at iteration $d$ ;
    end
    if $u_n^{d-1} \leq \frac{1}{N} \sum_{k=1}^{K} \tilde{p}_k f_d^{n,k}$ then
        generate optimality cut in (4.16) for $n$ ;
    end
    compute upper bound $ub = \min\{F^{d-1} - \sum_n u_n^{d-1} + \sum_{n=1}^{N} \sum_{k=1}^{K} p_n \tilde{p}_k f_d^{n,k}, ub\}$;
    update incumbent solution if upper bound is updated;
end
Step 2:
if $\exists n,\ s.t.\ u_n^{d-1} > f_d^{n,k}$ then
    stop;
    current incumbent solution is optimal;
else
    add the generated optimality cuts to Master with multicut and solve the master
    problem;
    set lower bound $lb = \max\{F^d, lb\}$;
    return to Step 1;
end
```

assignment preferences or restrictions. No Type 2 patients or Type 3 patients from previous blocks have been assigned to the remaining blocks ($\bar{a} = 0, \bar{b} = 0$). Table 4.1 shows the optimization results under the modified L-shaped and aggregated multicut L-shaped method. It is obvious that both methods can solve the simplified small scale problem within a few iterations and produce the same objective function value. As the size of the problem increases, the L-shaped method tends to take more iterations than the multicut, as well explained in [75, 76].

Table 4.1: Comparison of Performance and Outputs of Modified L-shaped and Aggregated Multicut L-shaped Methods

parameters				# iterations		obj value	
h	$\hat{s}$	$\hat{w}$	τ_1, τ_2	L-shaped	Multicut	L-shaped	Multicut
2	1	2	4,3	2	1	-1.5	-1.5
2	2	3	4,3	2	2	-3.5	-3.5
2	4	4	4,3	3	2	-3.2777	-3.2777
2	5	5	4,3	2	2	-4.0833	-4.0833
2	6	6	4,3	3	3	-4.7666	-4.7666
2	7	7	5,4	2	2	-6.375	-6.375
2	8	8	5,4	2	2	-7	-7
3	8	8	5,4	2	3	-9.1666	-9.1666
3	9	9	5,4	3	3	-8.1111	-8.1111
3	10	10	5,4	2	5	-9.4762	-9.4762
5	10	10	5,4	6	4	-8.25	-8.25
5	12	12	5,4	6	6	-6.25	-6.25
5	15	15	8,6	4	4	-13.5	-13.5
5	16	16	8,6	5	4	-14.2	-14.2
5	20	20	8,6	7	5	-16.3571	-16.3571

4.3.4 Linked Random Variables Without Closed-form Expectations

For the situation when the linked variables have empirical distribution, they do not have close-form expectations, cuts in (4.13) or (4.14) are no longer valid. To overcome this difficulty, assume that there is some relationship between each linked variable and its sample mean as well as sample standard deviation. In this problem, the linked variable is $\boldsymbol{a} = \{a_j\}$. Let K be the sample size, $\tilde{a}_j$ be the sample mean, μ_j be the population mean and s_j be the sample standard deviation. From Central Limit Theory, $\frac{\sqrt{K}}{s_j}(\tilde{a}_j - \mu_j) \to \text{Normal}(0, 1)$. Let δ be the accuracy control parameter, the value of a_j is supposed to satisfy:

$$\sqrt{K}\big|a_j - \tilde{a}_j\big| \leq \delta s_j \tag{4.20}$$

From the property of standard normal distribution, the distances of 99.7% of the data are within 3 times of standard deviation. So here one can choose δ to be 3. In (4.20) the population mean of α_j is approximated by a_j and therefore obtain the relationship between a_j and the sample data α_j. This approximation is based on the assumption that the clinic expects Type 2 patients arrive for their service with zero no-show rates. The deviation of this approximation from real result depends on the underlying distribution of α_j. For example if the real distribution of $\boldsymbol{\alpha}$ is $\alpha_j \sim \text{binomial}\big(a_j, (1-p_2)\big)$, but empirical distribution and approximation are used above, the average deviation is around $a_j p_2$ which will be small if p_2 is low. With this approximation, the master problem in modified L-shaped algorithm becomes:

$$\min \qquad -c_2 \textstyle\sum_{j=1}^{h} \sum_{k=1}^{\hat{s}} x_{jk} - c_3 \sum_{j=1}^{h} \sum_{t=1}^{\hat{w}} y_{jt} + u \tag{4.21}$$

$$\text{s.t.} \qquad \text{All first-stage Constraints in (4.1b) to (4.1j)} \tag{4.22}$$

$$u - \tfrac{1}{N} \textstyle\sum_{n=1}^{N} \sum_{k=1}^{K} \tilde{p}_k \sum_{j=1}^{h} b_j(\lambda_j^* - \mu_j^*) - \tfrac{1}{N} \sum_{n=1}^{N} \sum_{j=1}^{h} a_j(\lambda_j^* - \mu_j^*)$$

$$\geq \tfrac{1}{N} \textstyle\sum_{n=1}^{N} \sum_{k=1}^{K} \tilde{p}_k \sum_{j=1}^{h} \eta_j^n(\lambda_j^* - \mu_j^*) \tag{4.23}$$

$$\sqrt{K}(a_j - \tilde{a}_j) \leq \delta s_j \tag{4.24}$$

$$\sqrt{K}(a_j - \tilde{a}_j) \geq -\delta s_j \tag{4.25}$$

In order to verify the accuracy of this approximation method, the results of this method are compared with Algorithm 5 on the same test sets. In this experiment, it is assumed that the true distribution of α_j is binomial, i.e. $\alpha_j \sim \text{binomial}\big(a_j, (1 - p_2)\big)$. In Method 1 empirical data and formulations from (4.20) to (4.25) are used, in Method 2 Algorithm 5 is used. Table 4.2 shows the difference between objective function values of Method 2 and Method 1 over 15 test sets with the same sample size of 110 but different number of same-day requests. The direct observation from the table is that when the model size is not large, the differences between the two methods are within 1. Since integer unit cost coefficient is used and all the decision variables are integers, the difference of two methods in assigning patients is not significant. When the model size increases, a larger sample size is needed to be used to reduce the deviation.

Table 4.2: Objective Value Differences Between Method 2 and Method 1

# requests	Batch 1	Batch 2	Batch 3	Batch 4	Batch 5	Average
20	1.49	-0.26	1.12	-0.22	-0.52	0.4
30	-0.84	-1.32	-1.23	-3.21	-0.77	0.322
40	0.86	1.485	0.915	0.84	-2.1	-1.474

4.4 Alternative Formulations with Simple Recourse

According to Property III of SIPE-i, the second stage can be formulated in the simple recourse format:

$$q_{j+1} = \max\{\acute{q} + \sum_{k=1}^{j} \nu_k + \sum_{k=1}^{j} \alpha_k + \sum_{k=1}^{j} b_k - \sum_{k=1}^{j} \tau_k, 0\} \tag{4.26a}$$

$$g_j = \max\{-\acute{q} - \sum_{k=1}^{j} \nu_k - \sum_{k=1}^{j} \alpha_k - \sum_{k=1}^{j} b_k + \sum_{k=1}^{j} \tau_k, 0\} \tag{4.26b}$$

Note that equations (4.26a) and (4.26b) show the accumulative output and input of blocks with indices from i to $j+i$. From the comparison between the models with different second stage formulations, we can see that neither the overflow constraints in (4.4) and (4.26a) nor the patient shortage constraints in (4.26b) and (4.5) are the same. For any future block, if the patient shortage of previous block is positive, then calculation in (4.4) and (4.26a) will be different, so does (4.26b) and (4.5). These differences result from the definition of patient shortage in two models. In model (4.1), g_j is shortage at the particular block j, but in model (4.26), g_j is the total shortage of the remaining blocks 1 to j. Under the simple recourse formulation, the third component of the first stage model can be written in the following way:

$$\begin{aligned}\mathbb{E}[Q(X,Y,\boldsymbol{a},\boldsymbol{b},\omega)] &= \mathbb{E}[c_f\sum_{j=2}^{h+1} q_j + c_s\sum_{j=1}^{h} q_j] \\ &= c_f\sum_{j=1}^{h}\mathbb{E}[\max\{\acute{q} + \sum_{k=1}^{j}\nu_k + \sum_{k=1}^{j}\alpha_k + \sum_{k=1}^{j}b_k - \sum_{k=1}^{j}\tau_k, 0\}] \\ &+ c_s\sum_{j=1}^{h}\mathbb{E}[\max\{-\acute{q} - \sum_{k=1}^{j}\nu_k - \sum_{k=1}^{j}\alpha_k - \sum_{k=1}^{j}b_k + \sum_{k=1}^{j}\tau_k, 0\}]\end{aligned} \tag{4.27}$$

Let $\hat{\eta}_j := \sum_{k=1}^{j}\nu_k + \sum_{k=1}^{j}\alpha_k - \sum_{k=1}^{j}\tau_k$, $\hat{z}_j := -\acute{q} - \sum_{k=1}^{j}b_k$, so $\hat{\eta}_j$ is linear combination of random variables. Then q_j, g_j can written in the following way:

$$\begin{aligned}q_{j+1} &= \max\{\hat{\eta}_j - \hat{z}_j, 0\} \\ g_j &= \max\{\hat{z}_j - \hat{\eta}_j, 0\}\end{aligned} \tag{4.28}$$

Given these definitions, it is easy to derive the following equations.

$$\begin{aligned}\mathbb{E}[q_{j+1}] =& \mathbb{E}[\max\{\hat{\eta}_j - \hat{z}_j, 0\}] \\ =& \sum_{n=n_1^j}^{N} p_n(\hat{\eta}_j - \hat{z}_j) \\ =& \sum_{n=n_1^j}^{N} p_n\hat{\eta}_j - \sum_{n=n_1^j}^{N} p_n\hat{z}_j \\ =& -\hat{z}_j(1 - \sum_{n=1}^{n_1^j} p_n) + \sum_{n=n_1^j}^{N} p_n\hat{\eta}_j\end{aligned} \tag{4.29}$$

$$\begin{aligned}\mathbb{E}[g_j] =& \mathbb{E}[\max\{\hat{z}_j - \hat{\eta}_j, 0\}] \\ =& \sum_{n=1}^{n_1^j} p_n(\hat{z}_j - \hat{\eta}_j) \\ =& \sum_{n=1}^{n_1^j} p_n\hat{z}_j - \sum_{n=1}^{n_1^j} p_n\hat{\eta}_j\end{aligned} \tag{4.30}$$

where n_1^j is the largest index such that $\hat{\eta}_{j(n_1^j)} \leq \hat{z}_j$. Use the equations above, we have:

$$\begin{aligned}
&\mathbb{E}[g_j] - \mathbb{E}[q_{j+1}] \\
=& \sum_{n=n_1^j}^{N} p_n \hat{z}_j - \sum_{n=1}^{n_1^j} p_n \hat{\eta}_j + \hat{z}_j (1 - \sum_{n=1}^{n_1^j} p_n) - \sum_{n=n_1^j}^{N} p_n \hat{\eta}_j \\
=& \hat{z}_j - \sum_{n=1}^{N} p_n \hat{\eta}_j \\
=& \hat{z}_j - \mathbb{E}[\hat{\eta}_j]
\end{aligned} \tag{4.31}$$

$$\begin{aligned}
\mathbb{E}[g_j] + \mathbb{E}[q_{j+1}] =& \mathbb{E}[\max\{\hat{z}_j - \hat{\eta}_j, 0\}] + \mathbb{E}[\max\{\hat{\eta}_j - \hat{z}_j, 0\}] \\
=& \mathbb{E}[|\hat{z}_j - \hat{\eta}_j|] \\
\geq& \big|\mathbb{E}[\hat{z}_j - \hat{\eta}_j]\big|
\end{aligned} \tag{4.32}$$

From the inequality above, it is easy to obtain the following formulas.

$$\mathbb{E}[g_j] - \mathbb{E}[q_{j+1}] = \hat{z}_j - \mathbb{E}[\hat{\eta}_j] \tag{4.33}$$

$$\mathbb{E}[q_{j+1}] - \mathbb{E}[g_j] = \mathbb{E}[\hat{\eta}_j] - \hat{z}_j \tag{4.34}$$

$$\mathbb{E}[g_j] + \mathbb{E}[q_{j+1}] \geq \hat{z}_j - \mathbb{E}[\hat{\eta}_j] \tag{4.35}$$

$$\mathbb{E}[g_j] + \mathbb{E}[q_{j+1}] \geq \mathbb{E}[\hat{\eta}_j] - \hat{z}_j \tag{4.36}$$

From (4.34) and (4.36), we get:

$$\begin{aligned}
&\mathbb{E}[g_j] \geq 0 \\
&\mathbb{E}[q_{j+1}] \geq \mathbb{E}[\hat{\eta}_j] - \hat{z}_j
\end{aligned} \tag{4.37}$$

From (4.35) and (4.36), we can obtain:

$$\begin{aligned} &\mathbb{E}[q_{j+1}] \geq 0 \\ &\mathbb{E}[g_j] \geq \hat{z}_j - \mathbb{E}[\hat{\eta}_j] \end{aligned} \tag{4.38}$$

Assume the numbers of Type 1 and 2 patients who make their visits all follow Binomial distribution, and the block throughput follows Poisson distribution. Under this assumption, all the uncertain data has closed-form expectation, i.e.

$$\mathbb{E}[\alpha_k] = a_k(1 - p_2) \tag{4.39}$$

$$\mathbb{E}[\nu_k] = r_k(1 - p_1) \tag{4.40}$$

$$\mathbb{E}[\tau_k] = \frac{l_i}{\xi} \tag{4.41}$$

Let $u_j := c_s\mathbb{E}[g_j] + c_f\mathbb{E}[q_{j+1}]$ be a decision variable, using inequality pairs in (4.37) and (4.38), we have:

$$\begin{aligned} u_j &\geq c_f\big(\mathbb{E}[\hat{\eta}_j] - \hat{z}_j\big) \\ &= c_f\bigg(\acute{q} + \sum_{k=1}^{j} b_k + \sum_{k=1}^{j} \mathbb{E}[\nu_k] + \sum_{k=1}^{j} \mathbb{E}[\alpha_k] - \sum_{k=1}^{j} \mathbb{E}[\tau_k]\bigg) \\ &= c_f\bigg(\acute{q} + \sum_{k=1}^{j} b_k + \sum_{k=1}^{j} r_k(1 - p_1) + \sum_{k=1}^{j} a_k(1 - p_2) - \sum_{k=1}^{j} \frac{l_i}{\xi}\bigg) \end{aligned} \tag{4.42}$$

$$\begin{aligned} u_j &\geq c_s\big(\hat{z}_j - \mathbb{E}[\hat{\eta}_j]\big) \\ &= c_s\bigg(-\acute{q} - \sum_{k=1}^{j} b_k - \sum_{k=1}^{j} \mathbb{E}[\nu_k] - \sum_{k=1}^{j} \mathbb{E}[\alpha_k] + \sum_{k=1}^{j} \mathbb{E}[\tau_k]\bigg) \\ &= c_s\bigg(-\acute{q} - \sum_{k=1}^{j} b_k - \sum_{k=1}^{j} r_k(1 - p_1) - \sum_{k=1}^{j} a_k(1 - p_2) + \sum_{k=1}^{j} \frac{l_i}{\xi}\bigg) \end{aligned} \tag{4.43}$$

Let $v_j := \acute{q}+\sum_{k=1}^{j} r_k(1-p_1)-\sum_{k=1}^{j} \frac{l_i}{\xi}$, v_j which can be calculated as a parameter before decision. The SIP model considering no-shows of Type 2 patients can be transformed into the following integer programming model IN-i.

$$\text{(IN-i)} \qquad \min\ -c_2 \sum_{j=1}^{h} \sum_{k=1}^{\hat{s}} x_{jk} - c_3 \sum_{j=1}^{h} \sum_{t=1}^{\hat{w}} y_{jt} + \sum_j u_j \tag{4.44}$$

$$\text{s.t.} \qquad \sum_{j=1}^{h} x_{jk} \leq 1,\ \forall k = 1, \cdots, \hat{s} \tag{4.45}$$

$$\sum_{j=1}^{h} y_{jt} \leq 1,\ \forall t = 1, \cdots, \hat{w} \tag{4.46}$$

$$x_{jk} \leq A_{jk}, \quad j = 1, \cdots, h,\ k = 1, \cdots, \hat{s} \tag{4.47}$$

$$y_{jt} \leq B_{jt} z_t, \quad j = 1, \cdots, h,\ t = 1, \cdots, \hat{w} \tag{4.48}$$

$$a_j - \sum_{k=1}^{\hat{s}} x_{jk} = \bar{a}_{j+i}, \quad j = 1, \cdots, h \tag{4.49}$$

$$b_j - \sum_{p=1}^{\hat{w}} y_{jt} = \bar{b}_{j+i}, \quad j = 1, \cdots, h \tag{4.50}$$

$$\sum_{t=1}^{\hat{w}} (i+j) y_{jt} - \beta^{i-1} \sum_{p=1}^{\hat{w}} z_t \geq 0, \quad j = 1, \cdots, h, \quad t = 1, \cdots, \hat{w} \tag{4.51}$$

$$u_j - c_f \sum_{k=1}^{j} a_k (1-p_2) - c_f \sum_{k=1}^{j} b_k \geq c_f v_j, \quad j = 1, \cdots, h, \tag{4.52}$$

$$u_j + c_s \sum_{k=1}^{j} a_k (1-p_2) + c_s \sum_{k=1}^{j} b_k \geq -c_s v_j, \quad j = 1, \cdots, h, \tag{4.53}$$

$$x_{jk}, y_{jt}, z_t \in \{0,1\}, \quad a_j, b_j, u_j \in \mathbb{Z}^+,$$

$$j = 1, \cdots, h, \quad k = 1, \cdots, \hat{s}, \quad t = 1, \cdots, \hat{w} \tag{4.54}$$

It is obvious that IN-i model takes advantage of the simple recourse structure. It leverages expected values of random variables and the relationship between linked decision variables and linked random variables to achieve a bound for the original SIP-i problem. This reminds us about the Expected Results of Using Expected Value (EV) Solution (EEV) of the SP model, which measures the average objective function value among scenarios under EV solution.

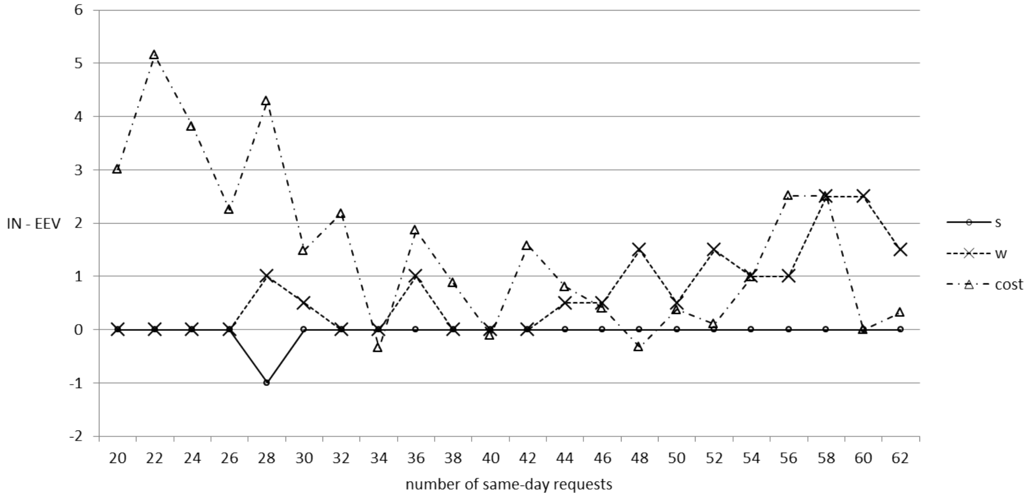
Comparing with procedure of obtaining EEV, IN-i model shows its strength in two aspects: (1) It is easy to operate, one does not need to iterate every scenario for obtaining

EEV; instead, the population mean is used in place of sample mean in IN-i for all the random variables. (2) It works smoothly for SIP-i with Endogenous uncertainty in this problem. If EEV is applied to this problem, due to the reason of endogenous uncertainty, some transformations need to be done to obtain the EV solution. Especially, there are different sample spaces of $\boldsymbol{\alpha}$ for different values of $\boldsymbol{a}$ which makes it nontrivial to calculate EEV. The population mean in IN-i and the relationship between linked decision variables and linked random variables overcome the difficulty brought by endogenous uncertainty. To compare IN-i and EEV, an additional assumption is adopted that Type 2 patients have zero no-show rates so that endogenous uncertainty is absent here. Figure 4.1 illustrates the difference between the two methods. The IN-i model and EEV are run to produce results over different number of requests under 20 batches of 110 samples, and compare average difference between the two methods on three objective components: number of assigned Type 2 patients (denoted with“s”), number of assigned Type 3 patients (denoted with “w”) and total cost of overflows and patient shortage (denoted with “cost”). On average, for all request numbers, the two methods produce very close results. The IN-i model shows a little higher costs but the average difference is around 2% of the cost of EEV. Hence IN-i can be used as a substitute of EEV for SP with endogenous uncertainty.

4.5 Conclusions of the Section

This section exploits two-stage SIP model with endogenous uncertainty to address the clinic scheduling problem. A modified L-shaped method and an aggregated multicut L-shaped method are designed to solve the model. Both methods perform well in producing optimal solutions. The underlying framework of these two methods offers a way to handle SP with decision dependent distribution parameters where the linked decision variable may not be involved in the second stage model. Here the potential relationship between decision variable and random variable is found and interpreted as equations or inequal-

Figure 4.1: Comparison of Computational Results of IN and EEV



ities which can be included in optimality cut. By doing this, the optimality cut retains results from scenarios of linked random variable and the participation of linked decision variable. This framework is flexible to deal with different types of distributions including empirical distribution where the relationship between decision variable and dependent random variable is not explicit. Patient cancellations, earliness and lateness can also be addressed under this framework. Introduction of these factors will increase the complexity of endogenous uncertainty. This section provides insights about the objective bounds of SP model with simple recourse function, based on which, the derived IN-i model can replace the EEV solution and overcome the calculation difficulty caused by endogenous uncertainty. Besides the IN-i model, another advantage of simple recourse function in this problem is to use a simplified version of optimality cut in modified / multicut L-shaped methods with prerequisite that the coefficient matrix in (4.26a) and (4.26b) is still unimodal. With this prerequisite, the simplified cut in [75] can be applied to Algorithms 5 and 6 as a good topic for future work.

5. ZIGZAG SORTING AND MAXIMUM INDEPENDENT SET BASED METHODS IN CLINIC SCHEDULING

5.1 Introduction

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. So when the clinic makes decision for the same-day requests, assignment of Type 1 patients is considered as known information. Since the traditional far-in-advance policy typically deals with chronic and follow-up care, so the service time of each Type 1 patients is more predictable than the same-day requests. With this information, the offline scheduling approach can be used to arrange the appointments of Type 1 patients. In practice, one policy for allocating Type 1 patients is to arrange the follow-up or chronic care patients into a limited portion of blocks leaving other blocks empty for the same-day requests. The occupied blocks for Type 1 patients are supposed to be fully utilized so that the same-day request decision maker can skip them, consequently, the same-day request assignment will have decision variables with fewer dimensions.

This section addresses the block-wise offline scheduling problem for Type 1 patients, where blocks are equal-length time intervals in a clinic day. The decision time horizon is not necessarily restricted to one clinic day, so the blocks under consideration may span several days. Each patient has preference on the blocks, so the individual assignment must obey the corresponding restrictions. Given the service time of each patient and the time limit of each block, the clinic assigns all these patients into the blocks following the assignment restrictions. The target is that the clinic uses the smallest possible numbers of blocks to serve the Type 1 patients, so that more blocks are available for the same-day requests patients.

The remainder of this section is arranged in the following way: Section 5.2 describes the integer programming formulations of the problem considering different types of assignment targets. Section 5.3 analyzes the complexity of the problem and the relation with other classical problems. Section 5.4 proposes a heuristic method which can perform the assignment efficiently and effectively. Section 5.5 designs a meta-heuristic algorithm with maximum independent set based construction, neighborhood representation and local search methods. Section 5.6 compares the performance of the heuristic and meta-heuristic methods with the exact solution method. Section 5.7 compares the performance of the construction method in this dissertation with existing construction methods. Section 5.9 draws conclusions about the work.

5.2 Problem Statement and Formulations

Assignment of Type 1 patients is addressed under the following assumptions: (1) the number of Type 1 patients to be assigned is known; (2) the expected service time of each patient is known; (3) the number of blocks under consideration is fixed, blocks have fixed equal lengths; (4) no-shows of Type 1 patients is not considered here; (5) patient preferences (restrictions) on the blocks are known. With these assumptions, the offline scheduling for Type 1 patients is trying to assign n patients into m blocks following the patients' preferences. The clinic wants to use the least number of blocks to serve these patients and there are no overflows from these blocks. This scheduling problem is a clinic patient assignment problem. Notations of the formulations of the clinic patient assignment problem are listed below:

Indices:

- j: index for patients.
- i: index for blocks.

Parameters:

- n: number of patients to be assigned in a day.
- m: number of blocks of a clinic day.
- $A := \{a_{ij}\}$: 0-1 restrictions of patients choices on the blocks. $a_{ij} = 1$ means patient j can be assigned to block i, $a_{ij} = 0$ means patient j cannot be assigned to block i.
- $\boldsymbol{t} := \{t_j\}$: service time of patient j.
- l: length of each block.
- c_u: cost of utilizing one block.
- c_e: cost of unit time beyond block length.
- r_a: unit revenue of assigning one patient.

Variables:

- x_{ij}: 0-1 decision variable indicating whether patient j is assigned to block i.
- y_i: 0-1 decision variable indicating whether block i is used.
- z: maximum makespan of blocks.

Formulations of the clinic patients assignment problem can be addressed from different perspectives. If the clinic needs to handle all n patients, and set a soft threshold for the makespan, the following formulation named as Assign-All (AA) can be used:

$$
\begin{aligned}
\text{(AA)} \quad \min \quad & f = c_u \textstyle\sum_{i=1}^{m} y_i + c_e(z-l) && \text{(5.1a)}\\
\text{s.t.} \quad & x_{ij} - y_i a_{ij} \leq 0,\ i = 1,\cdots,m, j = 1,\cdots,n && \text{(5.1b)}\\
& \textstyle\sum_{i=1}^{m} x_{ij} = 1,\ j = 1,\cdots,n && \text{(5.1c)}\\
& \textstyle\sum_{j=1}^{n} x_{ij} t_j - z \leq 0,\ i = 1,\cdots,m && \text{(5.1d)}\\
& x_{ij}, y_i \in \{0,1\}, z \in \mathbb{Z}^+,\ i = 1,\cdots,m, j = 1,\cdots,n && \text{(5.1e)}
\end{aligned}
$$

Objective function (5.1a) is designed to minimize the total cost generated from number of blocks used and the length of the maximum makespan. Constraint (5.1b) is the assignment restriction, (5.1c) assures one patient is assigned to only one block, (5.1d) defines the max makespan.

Suppose the clinic can reject the requests of some patients, but stick to the rule that the maximum makespan cannot go beyond the block length, and the clinic wants to assign as many patients as possible, then the following formulation named as Makespan-Restriction (MR) can be used:

$$
\begin{aligned}
\text{(MR)} \quad \min \quad & f = c_u \textstyle\sum_{i=1}^{m} y_i - r_a \textstyle\sum_{i=1}^{m} \textstyle\sum_{j=1}^{n} x_{ij} && \text{(5.2a)}\\
\text{s.t.} \quad & x_{ij} - y_i a_{ij} \leq 0,\ i = 1,\cdots,m, j = 1,\cdots,n && \text{(5.2b)}\\
& \textstyle\sum_{i=1}^{m} x_{ij} \leq 1,\ j = 1,\cdots,n && \text{(5.2c)}\\
& \textstyle\sum_{j=1}^{n} x_{ij} t_j \leq l,\ i = 1,\cdots,m && \text{(5.2d)}\\
& x_{ij}, y_i \in \{0,1\}, z \in \mathbb{Z}^+,\ i = 1,\cdots,m, j = 1,\cdots,n && \text{(5.2e)}
\end{aligned}
$$

If the clinic needs to handle all the n patients and the block length is a hard threshold, the Assign-All-Makespan-Restriction (AAMR) model can be used. The advantage of this model is that it satisfies all the requirements of the clinic patient assignment, but the disadvantage is that under strict restrictions, the feasible set may be empty or very small.

$$
\begin{aligned}
\text{(AAMR)} \quad \min \quad & f = c_u \textstyle\sum_{i=1}^{m} y_i + z && \text{(5.3a)}\\
\text{s.t.} \quad & x_{ij} - y_i a_{ij} \leq 0,\ i = 1, \cdots, m, j = 1, \cdots, n && \text{(5.3b)}\\
& \textstyle\sum_{i=1}^{m} x_{ij} = 1,\ j = 1, \cdots, n && \text{(5.3c)}\\
& \textstyle\sum_{j=1}^{n} x_{ij} t_j - z \leq 0,\ i = 1, \cdots, m && \text{(5.3d)}\\
& z \leq l && \text{(5.3e)}\\
& x_{ij}, y_i \in \{0, 1\}, z \in \mathbb{Z}^+,\ i = 1, \cdots, m, j = 1, \cdots, n && \text{(5.3f)}
\end{aligned}
$$

5.3 Complexity and Transformations of The Problem

5.3.1 Complexity Analysis

The analysis can be started from the AA model which has a soft threshold on makespan. The corresponding decision problem is:

- AA-Decision: Given n patients with individual service time t_j, m blocks and restriction matrix A, is there an assignment satisfying the assignment to assign the n patients into k of the blocks with makespan of z?

Theorem 5.3.1. *The AA-Decision problem is NP-complete.*

Proof:

It can be shown by reducing satisfiability problem (SAT) into AA-Decision. Let ϕ be an instance of conjunctive normal form (CNF) of the satisfiability problem with n clause $C_j, j = 1, \cdots, n$ and m variables, $X_i, i = 1, \cdots, m$ and the negations of the variables $\bar{X}_i, i = 1, \cdots, m$. Then one can obtain an instance of AA-Decision by mapping each clause to one patient and associating each variable X with one block. The literal X_i means that block i can be chosen, the literal $\bar{X}_i$ denotes that block i cannot be chosen. For example, let ϕ in (5.4) indicate the 6 patients and 5 blocks with restriction matrix A in (5.5).

$$\begin{aligned}\phi =&(X_1 \vee X_2 \vee \bar{X}_3 \vee \bar{X}_4 \vee X_5) \wedge (X_1 \vee X_2 \vee X_3 \vee \bar{X}_4 \vee \bar{X}_5) \\ &\wedge (X_1 \vee \bar{X}_2 \vee X_3 \vee X_4 \vee X_5) \wedge (X_1 \vee X_2 \vee X_3 \vee X_4 \vee X_5) \\ &\wedge (X_1 \vee X_2 \vee X_3 \vee X_4 \vee \bar{X}_5) \wedge (X_1 \vee \bar{X}_2 \vee X_3 \vee X_4 \vee \bar{X}_5)\end{aligned} \tag{5.4}$$

$$A^T = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \end{pmatrix} \tag{5.5}$$

This transformation costs $O(nm)$ time. It is easy to detect from the one-one mapping of the two that ϕ is true if and only if the AA-Decision in (5.5) is “Yes”. The one-one mapping also implies that AA-Decision is a special case of SAT, since SAT is in NP, so does AA-Decision. Hence AA-Decision problem is NP-complete.

5.3.2 Relation with Bin Packing Problem

If the restriction on assignment and the minimization of the maximum makespan are relaxed, then the problem becomes a BPP without unit bin capacity. To obtain the standard BPP with unit bin capacity, the service time of each patient can be worked out using the time limit of the max makespan, i.e., $\frac{t_j}{l}$. BPP has been proved to be NP-complete [79, 80]. To transform the input of instance of BPP with assign restriction into the input of basic BPP, the processing time can be changed from a vector $\boldsymbol{t}$ into a matrix T such that

$T := \{t_{ij}\}$ where

$$t_{ij} = \begin{cases} \frac{t_j}{a_{ij}l}, & \text{if } a_{ij} = 1, i = 1, \cdots, m, j = 1, \cdots, n \\ +\infty, & \text{if } a_{ij} = 0, i = 1, \cdots, m, j = 1, \cdots, n \end{cases} \tag{5.6a}$$

This transformation can be implemented in polynomial time $O(nm)$. The advantage of this transformation is that a feasible BPP solution will satisfy all the strict restrictions: makespan and full assignment. The disadvantage of this representation is that, if approximation method of BPP is adopted to solve it, it is easy to run into a situation where the full assignment cannot be satisfied in the end. So this transformation is close to the MR problem.

Using this transformation, it can be shown that MR is NP-complete The decision version of MR problem is:

- MR-Decision: Given n patients with individual service time t_j, m blocks and restriction matrix A, is there an assignment satisfying the assignment and makespan restriction to assign the s $(s \leq n)$ patients into k of the blocks?

Theorem 5.3.2. *The MR-Decision problem is NP-complete.*

Proof:

To show, one can reduce a partition problem into the MR-decision. In partition, get n integer numbers $t_1, t_2, \cdots, t_n$ and decide if there is a set $S \subset \{1, \cdots, n\}$ such that $\sum_{j \in S} t_j = \sum_{j \notin S} t_j$. A transformation similar to (5.6a) can be used to get the corresponding t_{ij} and construct the MR-decision instance with 2 blocks. Let $t_{ij} = 2\frac{t_j}{a_{ij}\sum_{j=1}^{n} t_j}$, so if there is an S that makes $\sum_{j \in S} t_j = \sum_{j \notin S} t_j$ true, then $1 \leq \sum_{i=1}^{2} \sum_{j \in S} t_{ij} \leq 2$ which means the MR-decision is true. If $\sum_{i=1}^{2} \sum_{j \in S} t_{ij} = 1$ which implies MR-decision is true, then $\sum_{j \notin S} t_j = 1$ which also means $\sum_{j \in S} t_j = \sum_{j \notin S} t_j$, so the partition instance is true.

Verifying whether an assignment of an instance of MR-Decision is true costs polynomial time, so it is easy to show that MR-Decision is in NP. Thus, the MR-Decision problem is NP-complete.

5.3.3 Relation with Maximum Independent Set Problem

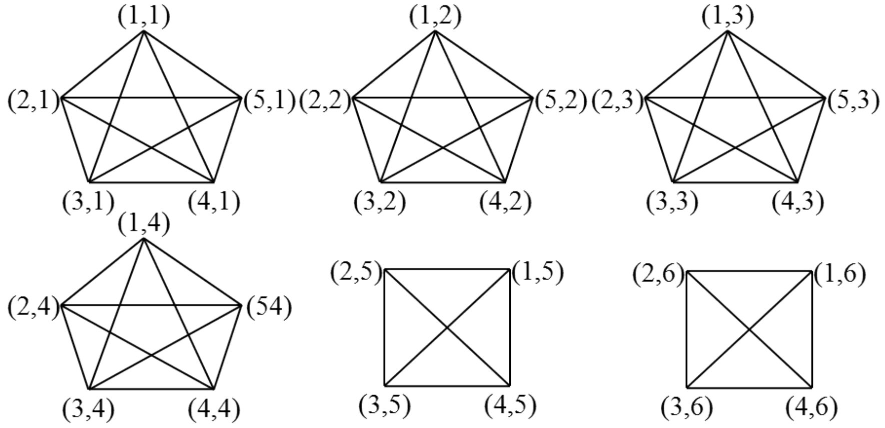
The method which transforms clinic assignment problem into a maximum independent set problem is modified on the basis of the work from Gabrel [90]. One can construct a graph $G = (V, E)$ in the following way: let $\sum_{i=1}^{n} \sum_{j=1}^{m} a_{ij} = b$ and $\sum_{i=1}^{n} a_{ij} = b_j$. Let a vertex denote the pair $v = (i, j)$ for patient j and block i with $a_{ij} = 1$, hence $|V| = b$. Connect the vertices which share the same patient, i.e., $(i, j) \to (k, j) \in E, j = 1, \cdots, n;\; i, k = 1, \cdots, m, i \neq k$. Graph G constructed in the way above has n cliques. A feasible solution for the clinic assignment problem would be a maximum independent set I of G. Let $s(I)$ be the number of blocks in the maximum independent set I, $p(I)$ be the length of the maximum makespan of these blocks. The target is minimum number of blocks to complete the service of all the patients within a short time no more than l. The objective function in (5.1a) is equivalent to:

$$f(I) = c_u s(I) + c_e(p(I) - l) \tag{5.7}$$

An example is given by the assignment restriction matrix A shown below and the corresponding graph G presented in Figure 5.1.

$$A^T = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \end{pmatrix} \tag{5.8}$$

Figure 5.1: Transformation from Clinic Patients Assignment Problem to Maximum Independent Set Problem



The AA problem is an optimization problem which aims at finding a maximum independent set I of size n such that the objective function value f in (5.7) is minimized. The transformation will take $O(n^3)$ time if using each column of A is constructed by each clique of G. The advantage of this transformation is that any feasible solution will preserve the all-assigning restriction. However, the makespan restriction is not guaranteed. Therefore this transformation is close to the AA model.

5.4 Approximation Methods of BPP-based MR

The existing approximation methods for BPP mentioned in literature review do not work well due to the assignment restrictions. Under these methods, some patients which are assigned late cannot find proper blocks. The unsuccessful trial on using the existing approximation methods of BPP shows that the feasibility of assignment can be achieved through multiple adjustments of an assignment. Specifically, if the procedure runs into an infeasible solution, it can alter the current assignment to find a way out. Let the term “degree of patient” denote the number of blocks in which a patient can be served according to restriction matrix A. The “degree of blocks" is defined in a similar way. The degree of each patient and each block will be updated during the assignment where the assigned patients and blocks with insufficient residual capacity will be removed from the candidate sets. To reduce the risk of running into an infeasible solution during the procedure, give highest priority to those patients who currently have only one choice on the blocks (call them 1-degree patients). The decreasing processing time order of item in NFD, FFD, BFD leads to a better approximation ratio since it tries to place “small” items into the residual space. When it applies to the patient assignment problem, the “large” item will take the bin so that “smaller” items have no suitable bins when they are to be assigned. If the patients are sorted increasingly according to their service time, then no enough “small” items are left to take the residual capacity of blocks which make the assignment away from “optimal”. So it is suggested to rearrange the patients order in a “zigzag” way. Suppose the clinic has an even number of patients. Sort the patients in non-decreasing order with respect to service time, i.e. $j_{(1)}, j_{(2)}, \cdots, j_{(n)}$, then rearrange the order based on the sorted order. The “zigzag" order based on non-decreasing sorted index is $j_{(\frac{n}{2})}, j_{(\frac{n}{2}+1)}, j_{(\frac{n}{2}-1)}, j_{(\frac{n}{2}+2)}, j_{(\frac{n}{2}-2)}, \cdots, j_{(n-1)}, j_{(2)}, j_{(n)}, j_{(1)}$. The advantage of this order is that the groups of patients whose sum service time can fit a capacity limit of a block are close to each other, i.e. patients with medium service length are close to patients with medium service length, patients with long service time are close to patients with short service time. For the patients with degree larger than 1, it always finds the block with maximum degree of patients following the Zigzag sorting order. So the algorithm is named as Max Fit Based on Zigzag Sorting with Retained Feasibility as presented in Algorithm 7.

To avoid the procedure running into a dead-lock in the case where it is unable to assign all the patients, define maximum number of loops N that can be performed by the inner while loop. To implement this algorithm, four lists can be used for storage:

- a list of patients preserving the zigzag sorted order who need to be assigned with

Algorithm 7: Max Fit Based on Zigzag Sorting with Retained Feasibility

```
1  Initialization: Rearrange the order of jobs in the “zigzag” way, let $J$ be set of
   patients, $M$ be the set of blocks ;
2  while $J \neq$ do
3      while there exists 1-degree patient in $J$ do
4          assign the 1-degree patients based on their order;
5          remove the assigned patient from $J$;
6          remove the blocks from $M$ which has not enough capacity ;
7          if 1-degree patient does not find a block then
8              find all the target blocks which the 1-degree patient can use ;
9              take the patients which are assigned to the target blocks as candidate
               patients ;
10             swap the unassigned 1-degree patient with the candidate patient with
               highest degree;
11         end
12     end
13     for the first patient in the order remaining in $J$ do
14         find the blocks associated with it as candidate blocks;
15         find the candidate block with the maximum degree within the makespan
           limit ;
16         assign the all the patients available to this block properly, remove the
           assigned patients and the block with insufficient residual capacity ;
17         break;
18     end
19 end
```

and their adaptive degree.

- a list of blocks with sufficient adaptive residual capacity and adaptive degree.
- a list of block-based assignment showing all assignments that have been done.
- a patient-block adjacency matrix storing restriction matrix A.

Using the lists, Lines 4-6 take $O(m)$ time, Line 8 searches the fourth list using $O(m)$ time, getting the candidate patients list takes $O(mn)$ time, finding the one with max degree costs takes $O(mn)$. Since it is necessary to go over the fourth list for each candidate, there are at most N loops between Lines 3-11, so the inner **while** loop costs $O(Nmn)$. Lines 14-17 cost time $O(m)$, Lines 13 -17 are executed only once per outer **while** loop. So the total computational complexity of Algorithm 7 is $O(Nmn^2)$.

Table 5.1 shows the performance of different heuristic methods where the length of each block is set to the maximum service time of patients. They are: I: Max fit with zigzag sorting; II: First fit with zigzag sorting; III: Best fit with zigzag sorting; IV: Mixed first fit and best fit with zigzag sorting; V: First fit without zigzag sorting; VI: Best fit without zigzag sorting; VII: Mixed first fit and best fit without zigzag sorting. The zigzag sorting is performed for patients based on their service time. The first fit with zigzag sorting is to assign patients one by one following their order to the first block that is able to serve them. The best fit with zigzag sorting is to assign patients one by one following their order to the block with minimum residual capacity that is able to serve them.

Table 5.1: Comparison of Heuristics Algorithms

Algorithms	with Zigzag Sorting				without Zigzag Sorting		
Cases	I	II	III	IV	V	VI	VII
1	0	1	2	1	1	2	1
2	0	0	1	1	0	1	1
3	0	0	1	1	0	1	0
4	0	0	1	0	0	1	1
5	0	0	1	2	1	1	1
6	0	0	1	0	0	1	0
7	0	0	1	0	0	1	0
8	0	0	1	0	0	1	1
9	0	0	1	1	0	1	1
10	0	0	1	1	2	3	3
sum	0	1	11	7	4	13	9

The comparison shows that except for the proposed max fit with zigzag sorting, all traditional methods are not reliable in providing assignment for all the patients. The zigzag sorting method improves the assignment ratio. From the performance perspective, first fit is better than best fit.

Theorem 5.4.1. *Zigzag sorting based algorithm has approximation ratio 2 in finding minimum number of blocks used.*

Proof. Let s be the number of blocks used given by the zigzag sorting based algorithm , s^* be the optimum number of blocks needed to be used. Assume that every block can serve all patients. Let k be the index of the last patient to be assigned whose service time is shorter than half of the block length. So for those patients that are already assigned with service time shorter than half of the block length, their block must have residual less than the service time of k. Let the number of occupied blocks be s_1. For the rest of patients who are not assigned, their service time is larger than half of the block length. Let the number of these patients be s_2. Then $s = s_1 + s_2 + 1$ or $s = s_1 + s_2$. For each case, the relationship is $\frac{l}{2}s_1 \leq \sum_{i \in s_1} t_i$, $\frac{l}{2}s_2 \leq \sum_{i \in s_2} t_i$, thus $\sum_i t_i \geq \frac{sl}{2}$. In the optimal solution, $s^* l \geq \sum_i t_i$. Thus, $s \leq 2s^*$.

□

5.5 Meta-heuristic for MIS-based AA

In this section, the patient scheduling problem is reduced into a maximum independent set problem on a graph with n cliques as mentioned previously. This transformation provides a new perspective to develop representation methods for meta-heuristics. It is obvious that a combination of one vertex from each clique of the graph is a feasible solution to the problem. Local search in neighborhood can be used to improve the solution for shorter makespan of blocks. In literature, heuristic methods like FFD and NFD are usually used to construct initial solutions for meta-heuristics for bin-packing problem. A procedure for Greedy Randomized Construction of feasible solutions is presented in Algorithm 8. The construction algorithm runs in time $O(nm \log m)$ where the sorting costs $O(m \log m)$ and dominates inside the **for** loop. The solution constructed this way may violate the max makespan limit, so the meta-heuristic will select a better solution over the iterations.

Algorithm 8: Greedy Random Construction

```
Initialize $\alpha$, let $J$ be set of patients, $M$ be the set of blocks ;
for each patient $j = 1, \cdots, n$ do
    get the subset of blocks associated with patient $j$, denoted as
    $M_j \in M, M_j = \{i \mid i \in M, a_{ij} = 1\}$ ;
    sort blocks in $M_j$ in a non-increasing order according to their degree $|M_j|$, let
    $p = |M_j|$ ;
    let $k = \lfloor 1 + \alpha p \rfloor$, choose the first $k$ blockes in sorted $M_j$ to form a Restricted
    Candidate List (RCL);
    randomly pick one block in the RCL, assign patient $j$ to it; remove patient $j$
    from set $J$;
end
Return the solution;
```

The neighborhood of a solution can be defined in the following way: replace one vertex in the solution say (i, j) with another vertex (k, j) , these two vertices share the same patient but different blocks. Denote the current solution as I, and the neighborhood of it as $N(I)$, then we have

$$N(I) = \{I' \mid I' = I \backslash (i, j) \cup (k, j)\} \tag{5.9}$$

The local search procedure is to find a solution which is optimum in its neighborhood as shown in Algorithm 9.

Algorithm 9: Local Search

```
Input: solution $I_0$ ;
$I = I_0$;
while there exists $I' \in N(I)$ such that $f(I') < f(I)$ using (5.7) do
    I = I';
end
Return $I$;
```

The greedy Randomized Adaptive Search Procedure (GRASP) in Algorithm 10 and Simulated Annealing (SA) in Algorithm 11 are adopted. For SA, the initial temperature is the maximum difference between two neighbor solutions. The final temperature is the minimum difference between two neighbor solutions.

Algorithm 10: GRASP for MIS-based AA

```
Initialize the maximum number of iterations $N$, $i = 0, I^* =$ ; while $i < N$ do
    $i = i + 1$;
    $I$ = Greedy Random Construction();
    Local Search(I) ;
    if $f(I^*) > f(I)$ then
        $I^* = I$;
    end
end
Return $I^*$;
```

5.6 Accuracy and Efficiency of the Designed Heuristic and Meta-heuristic Methods

In this section, accuracy and efficiency of the designed algorithms are demonstrated through numerical experiments. The proposed max fit zigzag construction, GRASP, and Simulated Annealing algorithms were developed in Matlab and then empirically evaluated by 13 instances of the clinic assignment problem which were randomly generated. The in-

Algorithm 11: SA for MIS-based AA

```
Initialize the temperature control parameter $c$, $i = 0$, $t_0$ is the initial temperature, $t_f$ is the final temperature, $f(I^*) = \infty$;
$I$ = Greedy Random Construction();
while $i < N$ do
    $i = i + 1$;
    $t = t_0$;
    $I = I_0$;
    while $t < t_f$ do
        take a solution $I'$ randomly from $N(I)$;
        if $f(I') < f(I)$ then
            $I = I'$ ;
        else
            if $e^{\frac{f(I')-f(I)}{t}} >$ random(0,1) then
                $I = I'$
            end
        end
        if $f(I) < f(I^*)$ then
            $I^* = I$
        end
        $t = ct$;
    end
end
Return $I^*$;
```

put test instances are shown in Table 5.2. Optimum results under AA model using CPLEX are shown in Table 5.3. Table 5.4 shows the results using max fit zigzag method. Table 5.5 summarizes the results from GRASP, Table 5.6 summarizes results from SA. The codes for GRASP and Simulated Annealing were run on MATLAB program through the Virtual Open Access Lab at our institution.

First, the accuracy of the results are discussed. It is obvious that all 3 proposed heuristic algorithms can produce solutions with decent quality. Approximate solutions generated by the max degree based on the zigzag sorting method are guaranteed to be within the approximation ratio of 2. For small-scale instances, such as instance 1 to 9, GRASP and Simulated Annealing were able to generate the same solutions as obtained by using CPLEX. Note that in the proposed Simulated Annealing procedure, the approximate solutions are obtained with only one set of cooling process. For even larger-scale instances, repeated cooling process with the best solution obtained by the last cooling process as a initial solution may be introduced to improve approximation rate. All the experiments in the remaining sections are performed using the same computer with Intel Core i7-2640 and 4 GB RAM.

Table 5.2: Test Instances

Test index	No. of blocks	No. of patients
1	10	12
2	8	9
3	5	6
4	12	12
5	14	15
6	8	9
7	10	12
8	14	15
9	13	12
10	16	18
11	20	21
12	50	51
13	30	30

From the running time of the algorithms, the max degree based on Zigzag approximation method maintained an obvious advantage regarding time efficiency, especially when dealing with large-scale instances. Its running time was the smallest among the three proposed algorithms. GRASP performed the worst with large-scale instances regarding runtime. Since the neighborhood is exhausted in each iteration of the local search of GRASP, the runtime of GRASP heavily depends on the size of instance and the number of itera-

Table 5.3: Best Solutions of AA Model from CPLEX

Test index	Max makespan	No. of used blocks
1	21	6
2	19	3
3	41	2
4	16	6
5	13	6
6	28	4
7	14	4
8	14	7
9	14	4
10	10	7
11	11	7
12	9	10
13	8	11

Table 5.4: Results from Approximation Method

Test index	Max makespan	No. of used blocks	Runtime(seconds)
1	21	6	0.00742
2	19	3	0.0094
3	41	2	0.003801
4	16	7	0.0091
5	13	7	0.0147
6	28	4	0.0037
7	14	5	0.0042
8	14	8	0.0037
9	14	4	0.00796
10	10	8	0.0657
11	11	8	0.011079
12	9	11	0.012572
13	8	12	0.010109

tions. The runtime of SA mainly depends on the number of iterations, which is affected by the choice of temperatures. However, when the problem size is small, GRASP is still a good tool to generate local optimal solutions.

Table 5.5: Results from GRASP

Test index	Max makespan	No. of used blocks	Runtime(sec)	Best iteration
1	21	6	2.4484	3
2	19	3	1.1532	3
3	41	2	0.4655	1
4	16	6	2.4526	1
5	13	6	4.6111	3
6	28	4	1.2337	1
7	14	5	1.8321	1
8	14	7	4.7488	5
9	14	4	3.6318	1
10	10	7	32.2791	145
11	11	7	94.5149	69
12	7	14	1368.3354	76
13	8	11	209.5082	20

Table 5.6: Results from SA

Test index	Max makespan	No. of used blocks	Runtime(sec)
1	21	6	0.7551
2	19	3	0.8896
3	41	2	0.865
4	16	6	0.872
5	13	6	0.6921
6	28	4	0.6799
7	14	5	0.7204
8	14	7	0.7971
9	14	4	0.7686
10	10	7	1.0423
11	11	7	0.985
12	7	14	1.5029
13	8	12	1.0657

5.7 Comparison with Other Construction Method in Meta-Heuristics

The performance of different construction methods for GRASP are compared. In the following experiments, we keep the neighborhood representation in Equation (5.9) and local search in Algorithm 9 but change the construction method. Ten construction methods are implemented. They are: 1. Clique-based greedy randomized construction with zigzag sorting of patients. 2. Clique-based greedy construction with zigzag sorting. 3. First fit construction with zigzag sorting. 4. Best fit construction with zigzag sorting. 5. Randomly-mixed first fit and best fit construction with zigzag sorting. Algorithms 6 to 10 are modified on the basis of 1 to 5 by removing the zigzag sorting of patients.

Table 5.7 shows the absolute distance of the output of the methods to the optimal solution under the metric of number of blocks used. 11 test cases are evaluated under these methods, the last row shows the average distance of the output of the methods to the optimal solution. Method 1 has the least distance among zigzag based methods. Method 6 shows the least distance among methods without zigzag sorting. These demonstrate the advantage of the proposed clique based greedy randomized construction methods. Best fit methods have better performance than first fit in these experiments.

Table 5.7: Comparison of Meta-heuristic Algorithms on Number of Blocks Used

Algorithms		with Zigzag sorting					without Zigzag sorting				
Test Cases	opt	1	2	3	4	5	6	7	8	9	10
1	6	5	4	6	6	6	6	4	5	4	4
2	3	4	3	3	3	3	4	4	3	3	3
3	2	2	2	2	2	2	2	2	2	2	2
4	6	6	5	5	4	4	6	3	5	5	5
5	6	7	6	6	4	6	7	3	6	5	6
6	4	4	4	4	5	4	5	4	4	5	5
7	4	6	5	6	5	6	5	4	6	6	6
8	7	7	3	6	4	5	8	4	6	7	7
9	4	5	6	5	4	4	5	4	5	5	5
10	7	6	5	5	4	4	7	4	5	5	5
11	7	7	4	6	5	6	7	3	5	5	5
Avg dist to opt		0.64	1.36	0.73	1.27	0.91	0.55	1.73	0.91	1.09	1.00

Table 5.8 shows the rank of the 10 methods within the two subsets: zigzag based methods or non-zigzag methods with respect to the metric of makespan of the blocks. The methods are supposed to produce solutions with shorter makespan, thus Rank 1 implies the shortest makespan and Rank 5 implies the longest makespan. Each row in the table records number of times for each rank over the 11 test cases of one method. For example, Method 1 has Rank 1 nine times, Rank 3 once and Rank 5 once in the 11 test cases. The last column shows the accumulative rank score for the methods. For example, the score 16 for Method 1 is obtained by $1 \times 9 + 3 \times 1 + 4 \times 1$. The lower the accumulative rank score is, the better the method is in producing a better solution. Again, Method 1 (clique-based greedy randomized construction with zigzag sorting of patients) and Method 2 (clique-based greedy randomized construction without zigzag sorting of patients) dominate over other methods.

Table 5.8: Comparison of Meta-heuristic Algorithms on Rank of Makespan (Increasing Order)

	Ranks					
Algorithms	1	2	3	4	5	accum rank
1	9	0	1	1	0	16
2	1	6	1	0	3	31
3	1	2	5	3	0	32
4	0	2	1	7	1	40
5	0	1	3	0	7	46
6	11	0	0	0	0	11
7	0	3	1	0	7	44
8	0	6	3	1	1	30
9	0	1	6	4	0	36
10	0	1	1	6	3	44

Table 5.9 shows the rank of the 10 methods within the two subsets but with respect to the objective function value, which consists both number of blocks used and the makespan

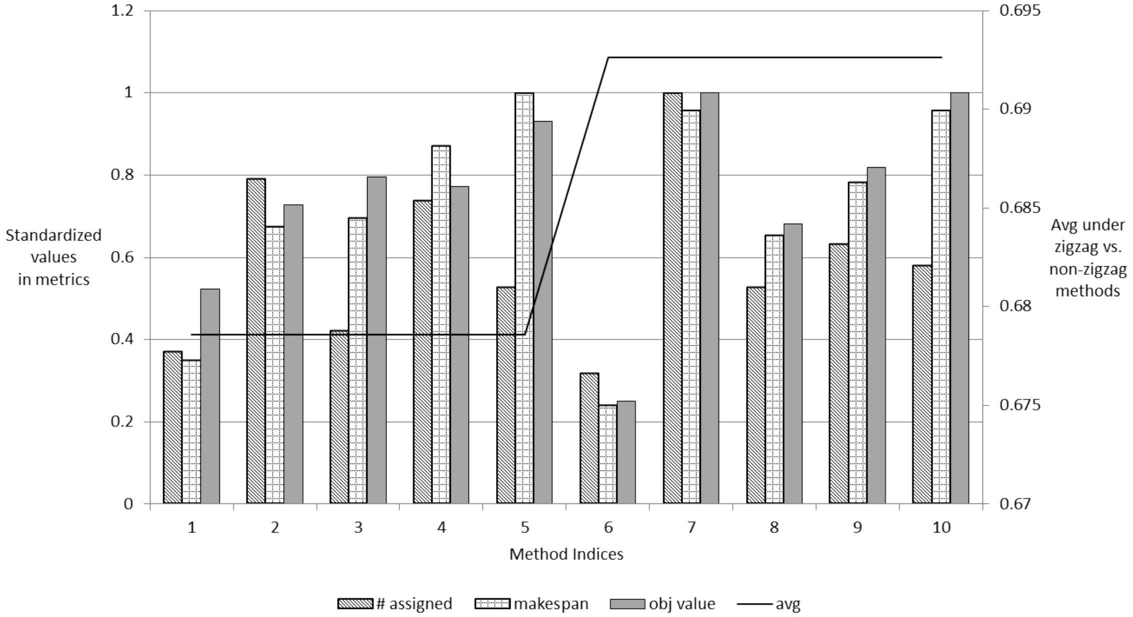
of blocks. The accumulative rank is calculated in a similar manner as in Table 5.8. Method 1 and Method 6 are still outstanding among these methods.

Table 5.9: Comparison of Meta-heuristic Algorithms on Rank of Objective Function Value (Increasing Order)

	Ranks					
Algorithms	1	2	3	4	5	accum rank
1	6	1	2	1	1	23
2	3	3	1	0	4	32
3	1	2	3	4	1	35
4	1	3	2	4	1	34
5	0	2	3	2	4	41
6	11	0	0	0	0	11
7	0	3	1	0	7	44
8	0	6	3	1	1	30
9	0	1	6	4	0	36
10	0	1	1	6	3	44

Figure 5.2 shows the standardized ranks of the 10 methods under the three metrics: number of assigned blocks, makespan of blocks and objective value. The advantage of clique based construction methods is quite obvious. Additionally, on average, the performance of zigzag methods is better than non-zigzag methods as shown by the line in the figure.

Figure 5.2: Comparison of Different Construction Methods for Meta-heuristics



5.8 Comparison with CPLEX on Larger Size Problems

In this section, performance of meta-heuristics proposed in this section is compared with CPLEX on AA problem. As shown in Table 5.7, CPLEX can deliver global optimal solution in short time when the size of the problem is not large. Based on extra experiments, CPLEX keeps this advantage until the number of blocks or patients increases to around 100. Table 5.10 shows the comparison of performance between the methods in limited time for large size problems. Method (a) is to use regular GRASP, method (b) is through GRASP with zigzag sorting, and method (c) is to use CPLEX. Test cases 1 to 9 have 1-minute time limit while test cases 10 to 18 have 5-minute time limit. Test cases 1 to 3 share the same input data with 100 blocks and 120 patients. Test cases 4 to 6 share the same input data with 200 blocks and 220 patients. Test cases 7 to 9 share the same input data with 300 blocks and 320 patients. Both the meta-heuristic methods are implemented in Java, and the AA model is also built in Java and solved using Concert Library of CPLEX

12.6. All the experiments are run on the same computer as mentioned in previous section. We can see that as the size of problem increases, performance of CPLEX drops faster than meta-heuristics. This is reflected by the large objective function values of CPLEX and the large relative gaps. It implies that when the size of the problem is large, meta-heuristic methods developed in this section can be a better choice than CPLEX to generate better solution efficiently.

Table 5.10: Comparison of Meta-heuristic Algorithms and CPLEX Performance on Large Size AA Problems

Test Case No.	Method	No. used blocks	Max makespan	Obj value	Gap
1	(a)	37	31.0	68.0	-
2	(b)	32	35.0	67.0	-
3	(c)	14	46.6	60.6	67.90%
4	(a)	16	157.0	173.0	-
5	(b)	15	156.0	171.0	-
6	(c)	8	427.1	435.1	98.02%
7	(a)	15	216.0	231.0	-
8	(b)	16	208.0	224.0	-
9	(c)	11	521.8	532.8	98.50%
10	(a)	23	44.0	67.0	-
11	(b)	32	28.0	60.0	-
12	(c)	15	42.4	57.4	66.23%
13	(a)	57	43.0	100.0	-
14	(b)	58	44.0	102.0	-
15	(c)	17	98.0	115.0	92.68%
16	(a)	89	49.0	138.0	-
17	(b)	82	61.0	143.0	-
18	(c)	10	594.6	604.6	98.64%

5.9 Conclusions to the Section

This section aims at assigning a certain number of patients with deterministic service time and individual preferences into a limited number of blocks where the sum of patients' service time in a block does not exceed the block length. This optimization problem is associated with three classical NP complete problems: the bin packing problem (BPP), minimum makespan problem and the maximum independent set (MIS) problem. A heuristic algorithm based on Zigzag sorting and feasibility restore policy is proposed to get an approximation solution of this assignment. Unlike the traditional heuristic methods which easily encounter the situation of infeasibility, this method guarantees to find a 2-approximate feasible solution at a fast speed. A meta-heuristic algorithm based on MIS transformation is designed. Performance of these two algorithms are compared with (c) mixed integer programming solver and with traditional approximation and meta-heuristic methods. The designed clique based algorithms exhibit advantages in giving better solution than traditional construction methods, and they are even better than (c) when the problem size is large. Although these methods are designed, implemented and evaluated under the topic of clinic scheduling, they can also be applied to other scheduling problems like job-shop problem with scheduling restrictions. The advantage of performing a good scheduling of Type 1 patients offers a clinic the flexibility to plan the scheduling of the same-day request patients as they arrive or call the clinic for an appointment.

6. CONCLUSIONS

As stated earlier, this dissertation provides insights about the hybrid clinic scheduling policy that handles both far-in-advance requests and same-day requests. For each topic, models are established on the basis of assumptions obeying the real rules, solution methods are developed so that exact or approximately optimal solutions can be worked out efficiently. Advanced and detailed analysis can enhance the impact of the research to real clinic management. For the same-day requests, this dissertation 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. For the chronic and follow-up requests, the clinic can try to assign them into aggregated blocks so that there are empty blocks to handle the same-day requests for reduced estimation and reduced uncertainty.

In implementation of the proposed method, the clinic is suggested 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 an appropriate decision. So one of the future work can be determining how to collect data from a clinic and developing reasonable and reliable estimators of the parameters. This work falls into the category of statistic inference and statistical learning.

The online scheduling part of the dissertation focuses on decision in each block, however, some block-wise features may pop up if the clinic performs the assignment throughout all the blocks. This can be a good topic for further research. A multi-stage SIP model based on block-wise request estimation can be exploited to handle the overall scheduling.

Analysis in Topic II will be useful for developing the multi-stage SIP solution method. Additionally, the objective function of SIP models in Topics I and II are risk-neutral. To consider the risk of a decision, a risk measure can be introduced into the model to obtain a mean-risk SIP model which tries to measure the cost and risk of the model. Risk measure can be chosen properly so that the objective function of the model is amenable to be optimized. There are ample problems that can be addressed about the offline scheduling. Further work can try to explore the connections between the clinic scheduling and vehicle routing problem with distance limitations and time window restrictions. If there are different physicians in different blocks, the minimum makespan with parallel non-identical machines may shed light on the solution method. What's more, for both online and offline part, the clinic can leverage simulation experiments for comparison or as a complementary tool to yield estimations of the hybrid scheduling policy. Simulation optimization is a good choice to handle the decision problem facing uncertainty during the assignment procedure.

The proposed methods do 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 $\boldsymbol{X}, \boldsymbol{Y}$ 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 a request later. In implementation of the proposed method, the clinic is suggested 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, it has been shown 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.

Although all the computational experiments in Section 3 and 4 are conducted for the first block, it is easy to apply the established modeling and solution methods to the remaining blocks. To do so, one only need to update the initial overflow number, and the accumulative same-day assignment of each block. Although the model is designed to cater to special situations like punctuality and cancellations, numerical studies on them are not provided here. One can also compare two ways of dispersion of capacity for the same-day request assignment in the future: (1) the scattered capacity for the same-day request where the Type 1 patients are assigned evenly into all the blocks. (2) the gathered capacity for the same-day requests where the Type 1 patients are assigned to a part of blocks leaving other blocks empty for the same-day requests. This comparison will provide insights about the traditional far-in-advance assignment as well as cooperation of the far-in-advance and open-access policies. Additionally, multi-stage model with endogenous uncertainty can be worked out for this problem in the future. Effort can be dedicated towards exploring methods to solve the multi-stage model with endogenous uncertainty with decision dependent random bounds.

Finally, the author will be pleased if any work from this dissertation can contribute to improve the well-being of sick people.

REFERENCES

[1] N. Liu, S. Ziya, and V. G. Kulkarni, “Dynamic scheduling of outpatient appointments under patient no-shows and cancellations,” *Manufacturing and Services Operations Management*, vol. 12, pp. 347–365, 2010.

[2] K. Phan and S. R. Brown, “Decreased continuity in a residency clinic: A consequence of open access scheduling,” *Family Medicine*, vol. 41, no. 1, pp. 46 –50, 2009.

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

[4] L. W. Robinson and R. R. Chen, “A comparison of traditional and open-access policies for appointment scheduling,” *Manufacturing & Service Operations Management*, vol. 12, no. 2, pp. 330–346, 2010.

[5] 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*, vol. 6, no. 3, pp. 345–376, 2015.

[6] K. Muthuraman and M. Lawley, “A stochastic overbooking model for outpatient clinical scheduling with no-shows,” *IIE Transactions*, vol. 40, pp. 820–837, 2008.

[7] C. Yan, J. Tang, and B. Jiang, “Sequential appointment scheduling considering walk-in patients,” *Mathematical Problems in Engineering*, vol. 2014, no. 564832, 2014.

[8] C. Liao, C. D. Pegden, and M. Rosenshine, “Planning timely arrivals to a stochastic production or service system,” *IIE Transactions*, vol. 25, no. 5, pp. 63–73, 1993.

[9] P. P. Wang, "Static and dynamic scheduling of customer arrivals to single-server system," *Computers and Operations Research*, vol. 24, pp. 703–716, 1993.

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

[11] B. Denton and D. Gupta, "A sequential bounding approach for optimal appointment scheduling," *IIE Transactions*, vol. 35, pp. 1003–1016, 2003.

[12] S. A. Erdogan and B. Denton, "Dynamic appointment scheduling of a stochastic server with uncertain demand," *INFORMS Journal on Computing*, vol. 25, no. 1, pp. 116–132, 2013.

[13] 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*, vol. 72, pp. 282–296, 2014.

[14] 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*, vol. 239, pp. 427–436, 2014.

[15] J. R. Birge and A. H. Dempster, "Stochastic programming approaches to stochastic scheduling," *Journal of Global Optimization*, vol. 7, no. 3–4, pp. 417–451, 1996.

[16] 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*, vol. 14, pp. 185–199, 1952.

[17] D. V. Lindley, "The theory of queues with a single server," *Proceedings Cambridge Philosophy Society*, vol. 48, pp. 277–289, 1952.

[18] T. Cayirli and E. Veral, “Outpatient scheduling in health care: A review of literature,” *Production and Operations Management*, vol. 12, no. 4, pp. 519–549, 2003.

[19] D. Gupta and B. Denton, “Appointment scheduling in health care: Chanllenges and opportunities,” *IIE Transactions*, vol. 40, pp. 800–819, 2008.

[20] H. Lau and A. H. Lau, “A fast procedure for computing the total system cost of an appointment schedule for medical and kindred facilities,” *IIE Transactions*, vol. 32, no. 9, pp. 833–839, 2000.

[21] C. Ho and H. Lau, “Minimizing total cost in scheduling outpatient appointments,” *Management Science*, vol. 38, no. 12, pp. 1750–1764, 1992.

[22] C. Ho and H. Lau., “Evaluating the impact of operating conditions on the performance of appointment scheduling rules in service systems,” *European Journal of Operational Research*, vol. 112, pp. 542–553, 1999.

[23] K. J. Klassen and R. Yoogalingam, “Appointment system design with interruptions and physician lateness,” *International Journal of Operations & Production Management*, vol. 33, no. 1, pp. 394–414, 2013.

[24] 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.

[25] L. W. Robinson and R. R. Chen, “Scheudling doctor’s appointments: Optimal and empirically-based heuristic policies,” *IIE Transactions*, vol. 35, no. 3, pp. 295–307, 2003.

[26] G. G. Kaandorp and G. Koole, “Optimal outpatient appointment scheduling,” *Health Care Management Science*, vol. 10, no. 3, pp. 217–229, 2007.

[27] D. Gupta and L. Wang, “Revenue management for a primary-care clinic in the presence of patient choice,” *Operations Research*, vol. 56, no. 3, pp. 576–592, 2008.

[28] T. R. Rohleder and K. J. Klassen, “Using client-variance information to improve dynamic appointment scheduling performance,” *Omega*, vol. 28, no. 3, pp. 293–302, 2000.

[29] W. Wang and D. Gupta, “Adaptive appointment systems with patient preferences,” *Manufacturing & Service Operations Management*, vol. 12, no. 3, pp. 373–389, 2011.

[30] J. Feldman, N. Liu, H. Topaloglu, and S. Ziya, “Appointment scheduling under patient preference and no-show behavior,” *Operations Research*, vol. 62, no. 4, pp. 794–811, 2014.

[31] M. Brahimi and D. J. Worthington, “Queuing models for out-patient appointment systems: A case study,” *Journal of the Operational Research Society*, vol. 42, no. 9, pp. 733–746, 1991.

[32] B. Jansson, “Choosing a good appointment system: A study of queues of the type (d/m/1),” *Operations Research*, vol. 14, pp. 292–312, 1966.

[33] A. Mercer, “A queuing problem in which arrival times of the customers are scheduled,” *Journal of the Royal Statistical Society Series B*, vol. 22, pp. 108–113, 1960.

[34] C. D. Pegden and M. Rosenshine, “Scheduling arrivals to queues,” *Computers & Operations Research*, vol. 17, no. 4, pp. 343–348, 1990.

[35] M. Brahimi and D. J. Worthington, “The finite capacity multi-server queue with inhomogeneous arrival rate and discrete service time distribution and its application to continuous service time problems,” *European Journal of Operational Research*, vol. 50, no. 3, pp. 310–324, 1991.

[36] R. W. Day, M. D. Dean, R. Garfinkel, and S. Thompson, “Improving patient flow in a hospital through dynamic allocation of cardiac diagnostic testing time slots,” *Decision Support Systems*, vol. 49, no. 4, pp. 463–473, 2010.

[37] L. Liu and X. Liu, “Block appointment systems for outpatient clinics with multiple doctors,” *Journal of the Operational Research Society*, vol. 49, pp. 1254–1259, 1998.

[38] Z. C. Zhu, B. H. Heng, and K. L.Teow, “Simulation study of the optimal appointment number for outpatient clinics,” *International Journal of Simulation Model*, vol. 8, no. 3, pp. 156–166, 2009.

[39] B. Fries and V. Marathe, “Determination of optimal variable-sized multiple-block appointment systems,” *Operations Research*, vol. 29, no. 2, pp. 324–345, 1981.

[40] L. Liu and X. Liu, “Dynamic and static job allocation for multi-server systems,” *IIE Transactions*, vol. 30, pp. 845–854, 1998.

[41] J. K. Lin, Muthuraman, and M. Lawley, “Optimal and approximate algorithms for sequential clinical scheduling with no-shows,” *IIE Transactions on Healthcare Systems Engineering*, vol. 1, no. 1, pp. 20–36, 2011.

[42] P. M. Vanden Bosch, C. D. Dietz, and J. R. Simeoni, “Scheudling customer arrivals to a stochastic service system,” *Naval Research Logistics*, vol. 46, pp. 549–559, 1999.

[43] P. M. Vanden Bosch, C. D. Dietz, and J. R. Simeoni, “Minimizing expected waiting in a medical appointment systems,” *IIE Transactions*, vol. 32, no. 9, pp. 841–848, 2000.

[44] E. N. Weiss, “Models for determining estimated start times and case orderings in hospital operating rooms,” *IIE Transactions*, vol. 22, pp. 143–150, 1990.

[45] A. Shapiro and A. Nemirovski, “On complexity of stochastic programming problems,” *Continuous Optimization Applied Optimization*, vol. 99, pp. 111–146, 2005.

[46] A. J. Kleywegt, A. Shapiro, and T. H. de Mello, “The sample average approximation method for stochastic discrete optimization,” *SIAM Journal on Optimization*, vol. 12, no. 2, pp. 479–502, 2001.

[47] S. Ahmed, “Two-stage stochastic integer programming: A brief introduction,” *Wiley Encyclopedia of Operations Research and Management Science*, 2011.

[48] A. J. King and R. J. Wets, “Epi–consistency of convex stochastic programs,” *Stochastics and Stochstics Reports*, vol. 34, no. 1, pp. 83–92, 1991.

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

[50] J. L. Higle and S. Sen, “Stochastic decomposition: A statistical method fo rlarge scale stochastic linear programming,” *Kluwer Academic Publishers*, p. 220, 1996.

[51] L. Stougie, “Design and analysis of algorithms for stochastic integer programming,” PhD Dissertation. Centre for Mathematics and Computer Science, Amsterdam. 1985.

[52] R. Schultz, “Continuity properties of expectation functions in stochastic integer programming,” *Mathematics of Operations Research*, vol. 18, no. 3, pp. 578–589, 1993.

[53] R. Schultz, “On structure and stability in stochastic programs with random technology matrix and complete integer recourse,” *Mathematical Programming*, vol. 70, pp. 73–85, 1995.

[54] R. Schultz, L. Stougie, and M. H. van der Vlerk, “Two-stage stochastic integer programming: a survey,” *Statistica Neerlandica*, vol. 50, no. 3, pp. 404–416, 1996.

[55] C. C. Caroe and R. Schultz, “Dual decomposition in stochastic integer programming,” *Operations Research Letters*, vol. 24, pp. 37–45, 1997.

[56] C. C. Caroe and J.Tind, “L-shpaed decomposition of two-stage stochastic programs with integer recourse,” *Mathematical Programming*, vol. 83, pp. 451–464, 1998.

[57] H. D. Sherali and B. M. P. Fraticelli, “A modification of benders’ decomposition algorithm for discrete subproblems: An approach for stochastic programs with integer recourse,” *Journal of Global Optimization*, vol. 22, no. 1, pp. 319–342, 2002.

[58] V. I. Norkin, G. C. Pflug, and A. RuszczyÅĎski, “A branch and bound method for stochastic global optimization,” *Mathematical Programming*, vol. 83, no. 3, pp. 425 – 450, 1998.

[59] S. Sen, J. L. Higle, and L. Ntaimo, “A summary and illustration of disjunctive decomposition with set convexification,” *Network Interdiction and Stochastic Integer Programming in Operations Research /Computer Science Interfaces*, vol. 22, pp. 105–125.

[60] Z. Sen and J. Higle, “The c3 theorem and a d2 algorithm for large scale stochastic mixed-integer programming: Set convexification,” *Mathematical Programming*, vol. 104, no. 1, pp. 1–20, 2005.

[61] Z. Sen and H. D. Sherali, “Decomposition with branch-and cut approaches for two stage stochastic mixed-integer programming,” *Mathematical Programming*, vol. 106, no. 2, pp. 203–223, 2006.

[62] G. Laporte and F. V. Louveaux, “The integer l-shaped method for stochastic integer programs with complete recourse,” *Operations Research Letters*, vol. 13, pp. 133–142, 1993.

[63] T. W. Jonsbråten, “Optimization models for petroleum field exploitation,” PhD Dissertation. Norwegain School of Economics and Business Administration. 1998.

[64] T. W. Jonsbråten, R. J.-B. Wets, and D. L. Barton, “A class of stochastic programs with decision dependent random elements,” *Annals of Operations Research*, vol. 82, pp. 83–106, 1997.

[65] V. Goel and I. E. Grossmann, “A class of stochastic programs with decision dependent uncertainty,” *Mathematical Programming*, vol. 108, no. 2, pp. 355–394, 2006.

[66] L. Hellemo, A. Tomasgard, and P. I. Barton, “Stochastic programming with decision dependent probabilities.” http://strato.impa.br/videos/2014-festival-incerteza/09-AsgeirTomasgard.pdf. Accessed: 2016 –07–10.

[67] V. Gupta and I. E. Grossmann, “Solution strategies for multistage stochastic programming with endogenous uncertainties,” *Computers & Chemical Engineering*, vol. 35, no. 11, pp. 2235–2247, 2011.

[68] V. Gupta and I. E. Grossmann, “A new decomposition algorithm for multistage stochastic programs with endogenous uncertainties,” *Computers & Chemical Engineering*, vol. 62, pp. 62–79, 2014.

[69] S. Ahmed, “Strategic planning under uncertainty: Stochastic integer programming approaches,” PhD Dissertation. University of Illinois at Urbana-Champaign. 2000.

[70] K. Viswanath, S. Peeta, and S. F. Salman, “Investing in the links of a stochastic network to minimize expected shortest path length,” *Technical Report of Purdue University*, 2004.

[71] H. Held and D. L. Woodruff, “Heuristics for multi-stage interdiction of stochastic networks,” *Journal of Heuristics*, vol. 11, no. 5–6, pp. 483–1092, 2005.

[72] P. Vayanos, D. Kuhn, and B. Rustem, “Decision rules for information discovery in multi-stage stochastic programming,” *2011 50th IEEE Conference on Decision and Control and European Control Conference*, pp. 7368–7373, 2011.

[73] M. Laumanns, S. Prestwich, and B. Kawas, “Distribution shaping and scenario bundling for stochastic programs with endogenous uncertainty,” *International Conference on Operations Research*, 2014.

[74] R. M. Van Slyke and R. J. B. Wets, “L-shaped linear programs with applications to optimal control and stochastic programming,” *Journal of Applied Mathematics*, vol. 17, pp. 638–663, 1969.

[75] J. R. Birge and F. Louveaux, *Introduction to Stochastic Programming*. Springer Series in Operations Research and Financial Engineering, 2011.

[76] S. Trukhanov, L. Ntaimo, and A. Schaefer, “Adaptive multicut aggregation for two-stage stochastic linear programs with recourse,” *European Journal of Operational Research*, vol. 206, pp. 395–406, 2010.

[77] L. Hellemo, “Managing uncertainty in design and operation of natural gas infrastructure,” PhD Dissertation. Norwegain University of Science and Technology. 2016.

[78] D. S. Johnson, “Approximation algorithms for combinatorial problems,” *J. Comput. System Sci.*, vol. 9, pp. 256–278, 1974.

[79] J. Blazewicz and K. Ecker, “A linear time algorithm for restricted bin packing and scheduling problems,” *Operations Research Letters*, vol. 2, pp. 80–83, 1983.

[80] J. Blazewicz, J. K. Lenstra, and A. H. G. R. Kan, “Scheduling subject to resource constraints: classification and complexity,” *Discrete Applied Mathematics*, vol. 5, no. 1, pp. 11–24, 1993.

[81] S. Martello and T. Paolo, *Knapsack problems: algorithms and computer implementations*, ch. Bin-packing problem, pp. 221– 240. New York: Johnwiley& Sons Ltd, 1990.

[82] S. Eilon and N. Christofides, “The loading problem,” *Management Science*, vol. 17, pp. 259–267, 1971.

[83] M. S. Hung and J. R.Brown, “An algorithm for a class of loading problem,” *Naval Research Logistics Quarterly*, vol. 25, pp. 289–297, 1978.

[84] S. Martello and T. Paolo., “Lower bounds and reduction procedures for the bin packing problem,” *Discrete Applied Mathematics*, vol. 28, pp. 59–70, 1990.

[85] R. E. Korf, “A new algorithm for optimal bin packing,” *American Association for Artificial Intelligence 2002 Proceedings*, pp. 731–736, 2002.

[86] A. Layeb and S. Chenche., “A novel grasp algorithm for solving the bin packing problem,” *International Journal of Information Engineering and Electronic Business*, vol. 4, no. 2, pp. 8 –14, 2012.

[87] S. Martello and T. Paolo., “Application of genetic algorithm for the bin packing problem with a new representation scheme,” *Mathematical Sciences*, vol. 4, no. 3, pp. 253–266, 2010.

[88] B. Brugger, K. F. Doerner, R. F. Hartl, and M. Reimann, “Antpacking âĂŞ an ant colony optimization approach for the one-dimensional bin packing problem,” *Evolutionary Computation in Combinatorial Optimization*, vol. 3004, pp. 41–50, 2004.

[89] E. Falkenauer, “A hybrid grouping genetic algorithm for bin packing,” *Journal of Heuristics*, vol. 2, no. 1, pp. 5–30, 1996.

[90] V. Gabrel, “Scheduling jobs within time windows on identical parallel machines: New model and algorithm,” *European Journal of Operational Research*, vol. 83, pp. 230–329, 1995.

[91] J. Weinberg, L. D. Brown, and J. R. Stroud, “Bayesian forecasting of an inhomogeneous poisson process with applications to call center data,” *Journal of the American Statistical Association*, vol. 102, no. 480, pp. 185–1198, 2007.

[92] S. Ahmed and A. Shapiro, “The sample average approximation method for stochastic programs with integer recourse,” *SIAM Journal of Optimization*, vol. 12, pp. 479–502, 2002.