Dynamic Capacity Allocation for Integrated Online and Offline Outpatient Services with Stochastic Multiple Revisits

Xiaoxiao Shen, *Member, IEEE*, Jun Lv, Shi-Chang Du, *Member, IEEE*, Andrea Matta, *Member, IEEE*

***Abstract*—Internet healthcare provides a new access way for revisits through texts or videos, which can lighten the service load on offline hospitals. In this study, we investigate the integrated capacity allocation problem while incorporating multiple revisit transitions between online and offline outpatient systems. The heterogeneity of revisit patients in terms of their chronic conditions and disease progression is also thoroughly considered. We formulate the problem as a multistage stochastic mixed-integer programming (SMIP) model. Dynamic decisions of matching service capacities with stochastic demands for first visit and multitype multiple revisits are made at each stage, which ensure that each type of revisit is assigned appointment on the patient preferred day as much as possible under meeting the interval restrictions of two consecutive revisits. We reformulate the model into an equivalent formulation that can be solved directly and develop a decomposition-based adaptive capacity allocation with revisit priority algorithm to solve the model. Numerical results illustrate the superiority of our proposed approach in terms of computation time and solution quality compared with commercial solver. In addition, to aid practitioners in applying the decision-making model more effectively, we also provide managerial insights into key factors such as capacity and demand patterns, revisit intervals, and cost coefficients. For instance, managers can identify and implement the optimal capacity pattern for a given demand pattern, as well as the best combinations of demand and capacity patterns, to minimize operation costs.**

***Note to Practitioners*—We develop a multi-stage stochastic optimization model for allocating capacities to first visits and multitype multiple revisits in integrated online-offline outpatient systems. Practitioners should update the occupied capacity information at the beginning of the decision period. At the end of the decision period, they should run the optimization model to assign this period realized demands. Stochastic demands in future periods, which have an impact on assignment decisions for realized demands, are optimized simultaneously. Then, only assignment decisions for realized demand are executed. The above procedures are repeated for each decision period to generate rolling plans with minimum operation costs. The proposed adaptive allocation algorithm is easy-implemented. It can be adopted especially for large-scale practical problems to obtain rather good solutions.**

***Index Terms*—Multistage stochastic programming, capacity allocation, multiple revisits, online and offline healthcare, decomposition-based adaptive algorithm.**

This work was supported by the National Natural Science Foundation of China under Grant Nos. 52275499 and 92467101, and by the *AutoTwin* project EU GA No. 101092021 (https://www.auto-twin-project.eu/). The authors also acknowledge the support of the China Scholarship Council Program (Grant No. 202306230016) for providing the scholarship. In addition, we thank Prof. Na Li for helpful discussions related to the topic of this study and acknowledge the support of the National Natural Science Foundation of China (Grant No. 72171144) for the overall research direction. *(Corresponding authors: Shi-Chang Du, Andrea Matta).*

Xiaoxiao Shen is with the School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China; the Department of Mechanical Engineering, Politecnico di Milano, Milan 20156, Italy; and the School of Mechanical and Electronic Engineering, Wuhan University of Technology, Wuhan 430070, China. (e-mail: xiaoxiao.shen@polimi.it)

Jun Lv is with Faculty of Economics and Management, East China Normal University, Shanghai 200241, China (e-mail: jlv@dbm.ecnu.edu.cn).

Shi-Chang Du is with the School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: lovbin@sjtu.edu.cn).

Andrea Matta is with the Department of Mechanical Engineering, Politecnico di Milano, Milan 20156, Italy (e-mail: andrea.matta@polimi.it).

I. Introduction

In recent years, Internet healthcare has developed rapidly and has been accepted by increasing number of doctors and patients. The advantages of Internet healthcare are prominent, which can reduce the service burden of offline hospitals [1,2]. Long-term diseases such as high blood pressure, diabetes treatment, cardiovascular disease, asthma, and Alzheimer's disease need a series of revisits for continuous treatment. Research shows that Internet healthcare significantly affects chronic patients in terms of quality-adjusted life years and adherence to physician instructions [3]. Bavafa et al. [4] conducted an empirical study on the impact of online appointments on revisit intervals. However, the operational management challenges remain unresolved in the practice of diverting revisit patients to online services, which is crucial for fully leveraging the advantages of online healthcare.

In one of Internet outpatient clinics we investigated, online treatments are available in many departments, such as oncology and internal medicine. The hospital managers are faced with the issue of properly arranging a series of chronic revisits to ensure continuity of care. The current practice of capacity allocation is as follows: each department has multiple doctors to choose from, and patients can make an appointment for the following week. The doctor's morning or afternoon is reserved for online appointments. New available appointments for the seventh day will be released at 4 pm every day, and the available appointments for the first to sixth days will be updated simultaneously. The allocation is based on manual or experience, and hospitals often have the problem of mismatching supply and demand between doctors and patients. For example, sometimes the demand for online revisits is insufficient, and it cannot be released to offline patients to make appointments, resulting in a waste of service capacities. Sometimes the arranged online appointments are insufficient to meet the demand of online revisits in a timely manner.

The integrated online and offline outpatient system investigated in this paper have several crucial features. First, it is

characterized by strong coupling correlations from both patient flow and service resource perspectives. From the standpoint of service resource, it cannot create new resources absolutely, but share existing medical resources with online appointments to a certain extent. On the other hand, patients in the system require multiple long-term revisits and exhibit heterogeneity in disease conditions and progression patterns. The demand from a type of revisit patient has three natures: a fixed number of appointments, revisit modality (i.e., online or offline), and time interval. These classifications are developed based on a combination of clinical practice guidelines and expert knowledge from senior physicians and hospital administrators at our collaborating hospital. Specifically, we consulted with specialists from departments such as cardiology, endocrinology, and rehabilitation medicine. They explained that the condition of chronic patients can change during the course of treatment. Physicians respond to disease progression by adjusting both the treatment approach and the revisit frequency. This serves as the basis for us to identify representative revisit patterns that align with standard protocols for chronic disease management and revisit care. Multiple times return visits from online services shift to offline, and multiple revisits from offline services transfer to online. As a result, from the patient flow perspective, online and offline patient flows creates strong temporal and spatial coupling across planning periods and both channels, and cannot be treated separately.

Second, patient demand arrives in a dynamic and uncertain manner. Third, for chronic disease patients requiring multiple revisits, the time interval between any two consecutive visits must fall within a medically reasonable range, and the appointment time for each revisit is initially unknown and must be determined through optimization. This renders the online and offline outpatient integrated capacity allocation problem particularly challenging and motivated us to study it exhaustively. To address the challenges discussed above, our research integrates online appointments into offline appointments and determines the appropriate plan for both new and multiple times returning patients over a finite planning horizon. A multistage capacity allocation model is developed to dynamically assign appointments for first visit and multiple revisits once the new demand information is obtained. We propose a decomposition-based adaptive capacity allocation with revisit priority (DB-ACARP) algorithm that is computationally fast, even for large-scale examples, to dynamically match appointment capacity and patient demand.

The major contributions of this study are threefold. First, to the best of our knowledge, this study is the first to investigate the integrated capacity allocation of both online and offline services for first-visit patients and multitype, multiple revisit patients to ensure continuous chronic care. While the efficient coordination of online and offline medical operations to balance healthcare demand and supply has received increasing attention in practice, it remains largely unexplored in the existing literature. Second, we develop a new multistage SMIP model for the problem. The model is the first to optimize multitype, multiple revisit loops under online and offline coupling manner. Our model has several features that are uncommon in earlier models, such as the considerations of future uncertain demands at current decision stage, the online and offline capacity allocation decisions, the revisit time interval constraints, and the preferred revisit day constraints. Third, this study integrates a rolling-horizon re-optimization framework with a Monte Carlo sampling-based, scenario-driven stochastic programming model to approximate an exact multistage stochastic program with a full scenario tree. This approach mimics real-world online operations and provides a computationally tractable solution method. It is widely adopted in large-scale practical problems due to its feasibility and scalability, and it typically yields improved performance. Then, we examine the structure of the nonlinear optimization model and reformulate it as an equivalent directly solved formulation. We also develop the DB-ACARP algorithm for solving this formulation. The algorithm innovatively introduces adaptive allocation operations, such as real-time tracking of allocated and unallocated demand, full-capacity utilization-driven patient allocation, and dynamic capacity updates. Numerical results indicate that in dynamic "online" capacity allocation settings, the DB-ACARP algorithm outperforms the commercial solver in both short-term and long-term costs, delivering rather great solutions in a short time. In addition, some insight findings are presented to help practitioners better apply the decision-making model. For example, when the patient's condition gets better, practitioners can adjust revisit interval or the allowable span of the revisit interval to reduce operational costs. Besides, by analyzing the combination effect of demand and capacity patterns, it has been indicated that adopting an optimal combination of these two patterns can minimize total operational costs.

The rest of this article is divided into the following sections. In Section II, the pertinent literature is reviewed. The next section describes the problem and establishes the optimization model for the integrated allocation of capacity to first visits and revisits from both online and offline channels. Section IV reformulates the model and proposes the DB-ACARP algorithm to solve this reformulation. To evaluate the effectiveness of the solution algorithm and the performance of the optimization model, numerical experiments are conducted in Section V. Finally, Section VI summarizes our major conclusions and offers directions for future research.

II. Literature review

Capacity allocation problem is typically addressed based on decisions for different patient categories. Such decisions often include determining how many patients of each type to accept and how many patients of each type, with different target dates, should be scheduled for booking on specific days. Typically, the goal in this class of problems is to minimize the overall cost or to maximize expected net profit. A stream of papers has studied capacity allocation in the application areas of diagnostic tests such as MRI or CT scans [7,9–11], primary care clinics [12,15], coarse-grained scheduling and resource allocation for operating rooms, anesthesiologists, and surgeries [6,8,13,16–18,20–22,24], inpatient admission and room allocation problem [25–28] and therapy planning [49]. Other research studies a general application scenario for capacity allocation problems [29–31]. Note that

the studies [17–18,20–22,24] in the field of surgery scheduling can be categorized as generalized capacity allocation problems. Although these studies are based on individual patient-level decisions, most of the scheduling decisions involved are limited to the date level, such as determining on which day a surgery should be performed, and the surgery start times are assumed to be known (e.g., assigning a surgery to a fixed block on a specific day). These types of problems can be viewed as coarse-grained scheduling, also referred to as multi-day scheduling in the literature. In other words, in terms of modeling structure and decision focus, they can be interpreted as generalized capacity allocation problems. This perspective is valuable for our study, which addresses the classical capacity allocation problem at the patient group level, as it helps bridge the concepts between scheduling and capacity allocation, thereby enabling a more unified modeling framework. The aforementioned research mainly focuses on the operation of offline hospitals, without accounting for online medical services. And they classify patients based on different priorities or wait time targets [7,10,30]. In addition, the assignment of series revisits is not involved in their models.

For healthcare services of radiotherapy, physical therapy, diabetes treatment, patients need repeated visits. Some works have addressed the capacity planning problem for a reentry system. Nguyen et al. [32] establish a mixed-integer programming model for capacity planning with the goal of minimizing the maximum required capacity, subject to constraints on the patient appointment lead-times. For an outpatient system with patient reentry, Nguyen et al. [33] further consider the uncertainties of first visit and revisit demands to determine the required number of physicians. To maximize long-run average earnings, Yu and Bayram [34] plan the capacity required for office and virtual appointments using a newsvendor model. Few studies focus on capacity allocation for series patients that need to be assigned at the time of admission. With the assumption that the number of appointments is fixed and known at the time of the initial appointment, Sauré et al. [35] provide a dynamic model for assigning patients with multiple radiation therapy appointments. Unlike Sauré et al. [35], Yu et al. [36] schedule a series of appointments considering random number of visits needed by a patient and a constant inter-visit time. Ding et al. [53] analyze the optimal policy that reserves capacity for potential revisit appointment right after the customer's previous visit. There is a growing body of literature, including Khorasanian et al. [14], Biggs and Perakis [19], and Heching et al. [23], that investigates routing and scheduling problem at home care programs where the patients involved typically require multiple visits. In [14], a Markov decision process (MDP) model is developed for a single nurse, assuming that both the number of referrals per day and the number of visits for each referral are uncertain. Biggs and Perakis [19] efficiently solve the online version of this problem using approximate dynamic programming (ADP) and machine learning techniques. Heching et al. [23] design an exact logic-based Benders decomposition to solve the problem with deterministic number of visits for each visit over a time period. The number of revisits for each type of patient is deterministic, consistent with [23] and our study, but differing from [14]. Additionally, Gao et al. [59] and Li et al. [60] investigated a team orienteering problem characterized by two required visits per patient, aiming to maximize total profit.

The above review of related studies reveals several areas of unresolved research. Our work focuses on capacity allocation with strong coupled online and offline multitype multiple revisit loops. Due to the rapid development of Internet healthcare over the years and the lack of systematic operation management research that limits to revisit intervals decision-making [5], online doctor-patient matching [52], single time revisit [54], and single day appointment scheduling [55] etc., integrated online and offline healthcare operation management is a new area of study. Second, while the current related studies typically classified patients according to various priorities, severity or wait time targets, such approaches could potentially limit their practical applicability in certain contexts. Our model characterizes patients by revisit frequency, the number of revisits, and the revisit modality, reflecting their conditions and progression patterns. Third, the model considers the practical requirements of the time interval between two consecutive revisits and patient preferred revisit day, which are important factors related to service quality. We formulate the series revisit assignment constraints that meet the requirements of the reentry interval. Also, we model the restrictions of meeting patient preferred reentry time as soft constraints to achieve a balance between service quality of patient demand-side and service cost of hospital supply-side.

The most common approach used in papers to formulate capacity allocation problems is MDP [7–10,15–16,29,31,35–39]. However, high-dimensional state and action spaces make exact solution methods intractable. Different approaches are developed to tackle such a problem. Patrick et al. [7] resort to a linear-programming-based ADP method. Liu et al. [8] apply a simple yet innovative variable transformation to reveal the monotonicity of the number of patients allowed in the system with respect to the state variable and downstream capacity. Zhou et al. [10] propose a modified Benders decomposition algorithm to solve the multi-stage stochastic programming reformulation. Yu et al. [36] and Astaraky and Patrick [16] use the policy iteration algorithm to design a heuristic policy. Parizi and Ghate [29] apply an ADP method to solve the MDP formulation considering uncertainty in demand and budget. Unlike the above studies, the structural properties of the finite-horizon MDP model and optimal policy are established in Dai et al. [15]. They also design two efficient heuristic policies from the theoretical results.

Some papers apply mathematical programming approaches to formulate capacity allocation problems in literature [17,20–23,50–51]. Most of the literature uses sample average approximation (SAA) [19,22]. Few papers develop distributional robust optimization [17,21], robust optimization [6,30], and conditional value-at-risk constraints [13] to address the uncertainties. Rath et al. [6] propose a data-driven robust optimization approach for allocating anesthesiologists and operating rooms, scheduling surgery sequences and start times, and minimizing overtime in large multispecialty hospitals. Sun et al. [13] adopt conditional value-at-risk constraints to model uncertain demand in the investigated anesthesiologist

scheduling problem. The SMIP models established in the above reviewed studies mainly apply static decision-making procedures that run the optimization model once in advance to obtain the unchanging results for a given planning horizon. In "online" allocation settings, where the parameter information is updated in real-time, as shown in the literature [40–43], multistage dynamic decision-making is a more suitable approach. Our model is effectively integrated into a rolling horizon planning to react to new data from patient first visits and multitype multiple revisits, where the planning can be rolling optimized whenever new information is obtained.

Solving the multistage dynamic SMIP model that contains all the scenario-based constraints is very time-consuming and impractical. The solving burden of SMIP can be significantly lessen if the problem is decomposed into a number of scenario-based subproblems. Approaches based on scenario decomposition are frequently used to solve SMIP models [44–48]. Benders decomposition method [19] and column-generation-based heuristic algorithm [21–22] are proposed to solve the two-stage stochastic programming models. Adopting the decomposition technique, we propose the DB-ACARP algorithm for allocating capacities to offline and online first visit and revisit patients. DB-ACARP delivers high-quality solutions within an extremely short time, even for large-scale problems with 2000 scenarios, achieving results in as little as 60 seconds. Notably, it demonstrates significant advantages in both short-term and long-term cost optimization, making it highly valuable for practical applications.

III. Stochastic Optimization Model for Integrated Capacity Allocation in Online and Offline Outpatient Services

A. Problem Description

In this subsection, we describe the capacity allocation problem in integrated offline and online outpatient system. The system has a given capacity level. These capacities need to be allocated to both online and offline clinics to serve patients. We assume that all patients are appointment-based. We assume patients adhere to prescribed revisit intervals once assigned, without considering no-shows or cancellations. Some patients are first-visit patients, meaning they are receiving medical care at this hospital for the first time. Others are revisit patients who have previously visited the hospital. First visits are typically conducted offline due to the need for necessary examinations. Revisit patients receive continuous care according to a designated medical plan. They return to the hospital at a certain interval for multiple revisits. Both online and offline services can accommodate revisits. Revisit patients with mild conditions may only require an online consultation for prescription refills, whereas those with more severe conditions need in-person visits. We assume that revisit patients can be categorized into different types based on their medical conditions and progression patterns, as reflected in revisit frequency and mode of consultation. The more severe the condition, the shorter the revisit interval, the higher the revisit frequency, and the greater the tendency for offline visits, and vice versa. For instance, a diabetic patient in stable condition may require multiple consecutive online revisits with longer intervals to prescribe medication. In contrast, a diabetic patient who was previously well-controlled but has recently experienced significant discomfort may need alternating online and offline revisits to address disease progression. Additionally, a newly diagnosed diabetic patient typically requires frequent revisits, often involving multiple consecutive offline appointments, and so on. The appointment demand for each patient type is uncertain. For the appointment requests received at each period, the scheduler must determine the number of patients to accept for each type and allocate them to a specific period. First visits and first revisits need meet maximum waiting time targets (MWTTs), meaning their appointments must be scheduled before a certain deadline. Patients have preferred dates for each revisit based on their availability, work, transport, or treatment cycles, with preferences collected via online booking, telephone, or onsite registration, while medical guidelines impose constraints on the time intervals between consecutive revisits. Specifically, a revisit cannot be scheduled earlier than a certain date after the previous one, nor later than another specified date. Our objective is to generate a rolling plan that minimizes the total cost, including patient rejection costs, overtime and idle capacity costs, and penalty costs related to deviations from patients' preferred dates. The patient flows involving multiple revisit loops in the problem are illustrated as Fig. 1.

B. Problem Formulation

Before formulating the problem, we summarize the model's notations in Table I. The arrival horizon is defined as the first N days, and is indexed by $n \in \mathcal{N} = \{1, \ldots, N\}$. The terms "period" and "day" are used interchangeably in this study. Stochastic first visits and revisits come to request appointments on future days in each period n of the horizon. We need to allocate a certain number of first and revisit appointments to a specific time period over an L-day booking window. There are I types of revisits indexed by $i \in \mathcal{I} = \{1, \ldots, I\}$ with different number of revisits, revisit intervals, and revisit mode. Let set $\mathcal{K}_i = \{1, \ldots, K_i\}$ denote the number of appointments needed by different types of revisit patients. Whether type $i \in \mathcal{I}$ revisit kth appointment needs to be set offline (online) is indicated by binary parameter $\alpha_{ik}(\beta_{ik})$, which equals one if yes and zero otherwise. The medically required revisit interval depends on the next revisit status and is restricted within [$\underline{a}^o$, $\overline{a}^o$]([$\underline{a}^e$, $\overline{a}^e$]), where $\underline{a}^o(\underline{a}^e)$ and $\overline{a}^o(\overline{a}^e)$ are the allowances for earliness and lateness in the offline (online) revisit interval. Let c_t represent the total capacities available on day t. r_t decides the proportion of assigned offline capacity on day t. b_1, b_2, and b_3 represent the fixed-length service times of an offline first visit, an offline revisit, and an online revisit, respectively. If regular capacity is insufficient or surplus, the resources are not used efficiently, incurring an overtime or idle cost. Let c_b^o and c_b^e denote unit overtime or idle cost on offline and online service capacities, respectively. In addition, the preferred visit time of type $i \in \mathcal{I}$ revisit patient kth appointment is τ_{ik}. The cost of a revisit not being assigned on τ_{ik} is considered and denoted as c_i^w. This delay or early cost provides an incentive to arrange the patient on τ_{ik} as much as possible.

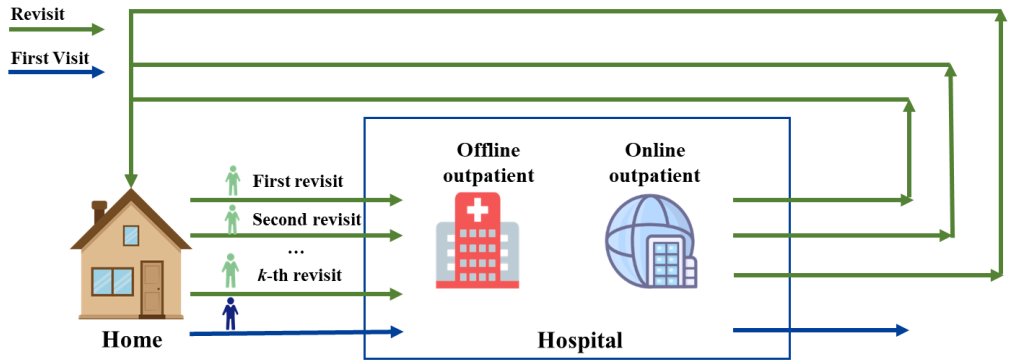


Fig. 1: The patient flows in the problem

TABLE I: Model Notation

Indexes and Sets	Description
$n \in \mathcal{N} = \{0, \ldots, N\}$	Set of periods within a planning horizon of N days, indexed by n
$l \in \mathcal{L} = \{1, \ldots, L\}$	Set of periods within a booking horizon of L days, indexed by l
$i \in \mathcal{I} = \{1, \ldots, I\}$	Set of revisit patient types, indexed by i
$k_i \in \mathcal{K}_i = \{1, \ldots, K_i\}$	Set of the number of appointments needed by type $i \in \mathcal{I}$ revisit patient, indexed by k_i
$s \in \mathcal{S} = \{1, \ldots, S\}$	Set of scenarios, indexed by s
Parameters	**Description**
F_0, R^k_{i0}	The number of first visit patients and type $i \in \mathcal{I}$ revisit patient' kth appointment that need to be assigned at current decision point
f^s_n, r^{ks}_{in}	The number of first visit patients and type $i \in \mathcal{I}$ revisit patient' kth appointment that arrive on day $n \in \mathcal{N}\backslash\{0\}$ in scenario s
W_1, W_2	MWTTs of revisit patient first appointment and first visit patient
M	A large number
τ_{ik}	The patient-preferred visit day of type $i \in \mathcal{I}$ revisit patient kth appointment
$\underline{a}^o(\underline{a}^e),\ \overline{a}^o(\overline{a}^e)$	Minimum and maximum time interval if revisit patients access care offline (online)
c_t	Regular total service capacities on day t
$\alpha_{ik}(\beta_{ik})$	The offline (online) care access status of type $i \in \mathcal{I}$ revisit patient kth appointment
$\underline{v}_{ik}(\overline{v}_{ik})$	Minimum(maximum) time interval of type $i \in \mathcal{I}$ revisit patient kth appointment, which equals to $\alpha_{ik}\underline{a}^o + \beta_{ik}\underline{a}^e(\alpha_{ik}\overline{a}^o + \beta_{ik}\overline{a}^e)$
b_1, b_2, b_3	Fixed service length of an offline first visit, offline revisit and online revisit
c^w_i	One day delay or early cost of assigning a type i revisit patient
c^o_b, c^a_b	Unit overtime and idle cost
$c^r_d(c_d)$	Rejection or diverting cost per revisit (first visit) patient
p_s	Probability of scenario s
γ	Discount factor
Decision variables	**Description**
$u_t,\ x^t_{ik}$	Integer variables, the number of first visit patients and type-i revisit patients' kth appointment that arrive at current decision point and are assigned on day t
y^{ts}_n, z^{ts}_{ikn}	Integer variables, the number of first visit patients and type-i revisit patients' kth appointment that arrive on day $n \in \mathcal{N}\backslash\{0\}$ and are assigned on day t in scenario s
$a^r_{i0}, a_0, a^{rs}_{in}$, and a^s_n	Continuous variables, the unmet patient demand at current decision point and on day $n \in \mathcal{N}\backslash\{0\}$ in scenario s. The demand includes first visit patients and type-i revisit patients, where r denotes revisit patients.
r_t	Continuous variables, the proportion of assigned offline capacity on day t
$o^o_t(o^e_t), a^o_t(a^e_t)$	Continuous variables, the offline(online) overtime or idle time on day t
$o^{os}_t(o^{es}_t), a^{os}_t(a^{es}_t)$	Continuous variables, the offline(online) overtime or idle time on day t in scenario s
$\theta^t_{ik}, h^{ts}_{ikn}$	Binary variables, auxiliary variables

To obtain dynamic capacity allocation decisions, we formulate the problem as a multistage SMIP model. A series of random demands $((f_0,\ r^k_{i0}),\ (f_1,\ r^k_{i1}),\ \ldots,\ (f_N, r^k_{iN}))$ are progressively indicated over the course of N stages in the multistage programming. To accommodate to this process, at each stage, once the stochastic demands are revealed, decisions regarding patient scheduling are made. Random parameters are realized by gradual observation. The scenarios set $\mathcal{S}$ is considered, including countable number of realizations S of random parameters. Let $(f^s_n,\ r^{ks}_{in})$ represent the first visit and revisit demands under scenario s arriving on day $n \in \mathcal{N}$, which takes place with a probability p_s. The number of first visit and type i revisit arriving on day $n \in \mathcal{N}$ is characterized by two different random variables that follow different Poisson distributions with unequal expectation. We assume that decision points correspond to the end of the day since patients arrive for their appointments throughout the day, and their status is not fully known until that time. At the decision point, the stochastic demands $(f^s_0,\ r^{ks}_{i0})$ are observed to obtain deterministic values $(F_0,\ R^k_{i0})$ without scenario index s. It is also possible to observe the capacity plan prior to the decision point, including details on the number of online and offline capacities already allocated to first visit and revisit appointments on day t $(t = 1, \ldots, L-1)$. Then, the available online and offline capacities on day t is updated. The optimization decisions of capacity allocation are made at current decision point over an L-day booking window by determining the number of first visit patients and type-i revisit patient's kth appointments that arrive on day $n \in \mathcal{N}$ and are assigned on day t under scenario s. We denote these decision variables by y^{ts}_n and z^{ts}_{ikn}. The first revisit and the first visit patient must be given on $t \in n+1, n+2, \ldots, n+W_1$ and

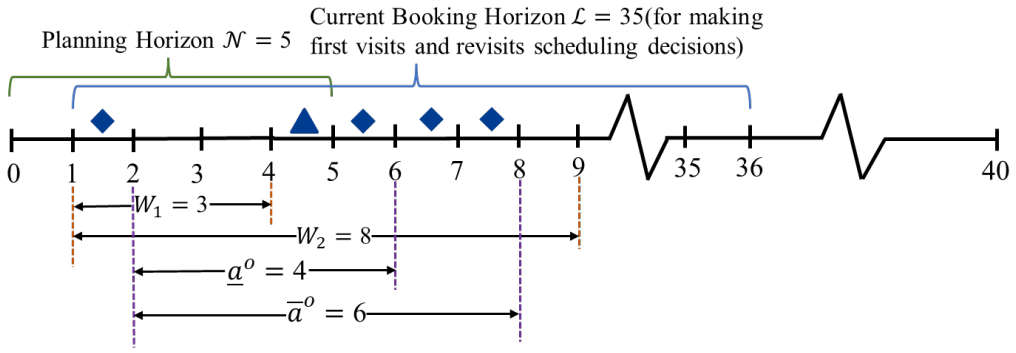


Fig. 2: The timeline of planning, current booking horizon and allocation windows. In this example, triangles indicate first visits, and diamonds indicate revisits. Demands generated in period 0–1 must be allocated. The first revisit can only be assigned to periods 1–4; if allocated to periods 1–2, the second revisit is restricted to periods 5–8. First visits can be scheduled in periods 1–8.

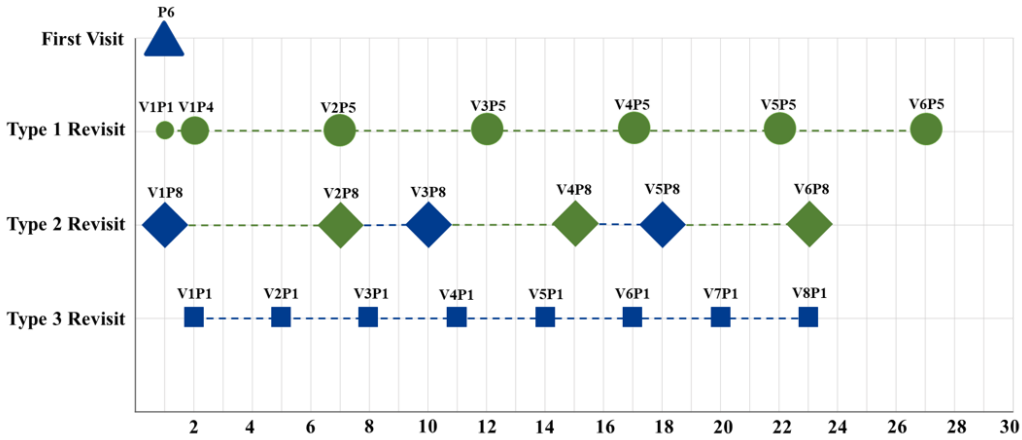


Fig. 3: A small illustrative case for allocating the current realized demand: 6 first visits (Triangle), 5 Type 1 revisits (Circle) with 6 loops, 8 Type 2 revisits (Diamond) with 6 loops, and 1 Type 3 revisit (Square) with 8 loops. Blue indicates offline visits, and green indicates online revisits. The notation V[k]P[n] denotes that n patients are allocated for the kth revisit. Shape size reflects the number of patients allocated.

day $t \in n+1, n+2, \ldots, n+W_2$, respectively, where W_1 and W_2 are their appointment maximum wait time targets (MWTTs). To account for the maximum appointment lead-time permitted for the last revisit in various types of revisits, we must set L to be significantly long. Fig. 2 is the visualization of the timeline of planning, current booking horizon and allocation windows. Fig. 3 visualizes the allocation plans for the current realized demand in a small illustrative case. The schedule generated for next day is put into action. The random demands (f_1^s, r_{i1}^{ks}) are observed at next stage, and the information of available capacities is updated before applying the model for making decisions at next stage. This procedure is repeated for the remaining decision optimization stages. The event sequence throughout the decision process is illustrated in Fig. 4.

For the first visits and revisits capacity allocation problem in the integrated online and offline healthcare system, we now present a multistage stochastic mixed-integer formulation. The model is used to optimize capacity allocation decisions for the next L days at the specific decision point $d \in \mathcal{D} = \{0, \ldots, D\}$. The immediate cost (incurred by patient demands that arrive at current decision point, and by the first period) and the future cost (incurred by patient demands that arrive on day $n \in \mathcal{N}\backslash\{0\}$, and by period $t \in \mathcal{L}\backslash\{1\}$) both consist of three types of costs:

(1) Penalty cost for revisits cannot be assigned appointments on their preferred visit day.

Immediate cost:
$TC_{w0} = \sum_{i\in\mathcal{I},k\in\mathcal{K}\backslash\{1\}} \sum_{t=1}^{L} c_i^w \left|t - \tau_{ik}\right| x_{ik}^t$
Future cost:
$TC_w^s = \sum_{n=1}^{N} \sum_{i\in\mathcal{I},k\in\mathcal{K}\backslash\{1\}} \sum_{t=1}^{L} \gamma^n c_i^w \left|t - \tau_{ik}\right| z_{ikn}^{ts}$

(2) Rejection or diverting cost for revisits and first visits.

Immediate cost:
$TC_{d0} = \sum_{i\in\mathcal{I}} c_d^r a_{i0}^r + c_d a_0$
Future cost:
$TC_d^s = \sum_{n=1}^{N} \sum_{i\in\mathcal{I}} \gamma^n c_d^r a_{in}^{rs} + c_d a_n^s$

(3) Overtime and idle cost for online and offline services.

Immediate cost:
$TC_{oa}^1 = c_b^o(o_1^o + o_1^e) + c_b^a(a_1^o + a_1^e)$
Future cost:
$TC_{oa}^s = \sum_{t=2}^{L} \gamma^{t-1}(c_b^o(o_t^{os} + o_t^{es}) + c_b^a(a_t^{os} + a_t^{es}))$

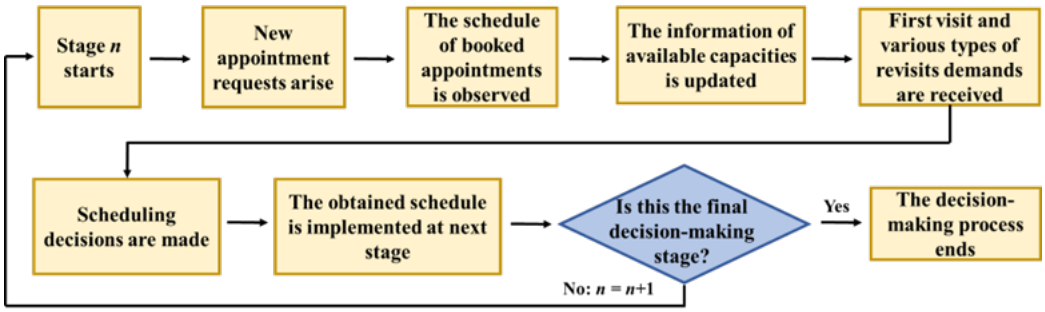


Fig. 4: The event sequence throughout the decision process.

The objective function (1a) is designed to minimize the overall expected costs for violating the patient preferred revisit day, rejecting or transferring patients, capacity overtime, and capacity idle time. Constraints (1b)–(1e) state that each admitted revisit and first visit patient is assigned an appointment at a period. It is noted that the model does not strictly require all patient demands to be fulfilled. Instead, a certain level of unmet demand is permitted. Specifically, the first equation in constraints (1b) ensures that the first revisit of accepted, realized type i revisit patient is assigned an appointment within a period before their MWTTs. The second equation in constraints (1b) guarantees that the subsequent revisits of these patients are allocated within the designated booking horizon. Constraints (1c) enforce that accepted, realized first-visit patients receive appointments before their MWTTs. The formulation logic of constraints (1d) to (1e) is consistent with that of (1b) and (1c), but these constraints address the allocation of stochastic demand arriving in future periods $n \in \mathcal{N} \setminus \{0\}$. As new demand arrives over time, the booking horizon dynamically shifts forward to accommodate it. Constraints (1f)–(1i) stipulate that the assignment days of two adjacent revisits meet the revisit interval requirements. Here, 1(condition) is defined as an indicator function, which equals 1 if the condition is true, and 0 otherwise. Specifically, constraints (1f) and (1g) ensure that if the (k-1)th revisit of a realized type i revisit patient is assigned to period t, then the kth revisit must be assigned within the interval from period $t + \underline{v}_{ik}$ to $t + \bar{v}_{ik}$. In other words, any assignment of the kth revisit outside this specified interval, whether before or after period t, is strictly prohibited and set to zero. The formulation logic of constraints (1h) and (1i) follows the same structure as that of (1f) and (1g), but these constraints are designed for allocating stochastic demand arriving in future periods $n \in \mathcal{N} \setminus \{0\}$. Constraints (1j)–(1q) calculate the amount of offline and online overtime and idle time. Constraints (1r)–(1t) define the range of decision variables.

IV. Solution Approach

In this section, we first transform the model into a formulation that can be solved directly by a commercial solver. Specifically, introducing the auxiliary variables to linearize the objective function with absolute value and the constraints with indicator functions. Then, since large-scale stochastic problems with many scenarios are difficult to solve by commercial solvers, a DB-ACARP algorithm is proposed to obtain solutions to large scale stochastic problems.

A. Reformulation of the Proposed Model

The proposed model in Section III includes absolute values in the objective function and indicator functions with continuous variables in the constraints, which makes the model impossible to be solved directly. First, we linearize the objective function with absolute values. We define auxiliary variables d^t_{ik}, dn^{ts}_{ikn}. The following constraints are added to the original model:

$$d^t_{ik} \geq (t - \tau_{ik})\, x^t_{ik}, \forall i \in \mathcal{I}, k \in \mathcal{K}\backslash\{1\}, t \in \mathcal{L}, \tag{4a}$$

$$d^t_{ik} \geq -(t - \tau_{ik}) x^t_{ik}, \forall i \in \mathcal{I}, k \in \mathcal{K}\backslash\{1\}, t \in \mathcal{L}, \tag{4b}$$

$$dn^{ts}_{ikn} \geq (t - \tau_{ik}) z^{ts}_{ikn}, \forall i \in \mathcal{I}, k \in \mathcal{K}\backslash\{1\}, n \in \mathcal{N}\backslash\{0\}, t \in \mathcal{T}, s \in \mathcal{S}, \tag{4c}$$

$$dn^{ts}_{ikn} \geq -(t - \tau_{ik}) z^{ts}_{ikn}, \forall i \in \mathcal{I}, k \in \mathcal{K}\backslash\{1\}, n \in \mathcal{N}\backslash\{0\}, t \in \mathcal{T}, s \in \mathcal{S} \tag{4d}$$

After replacing the absolute value term with new auxiliary variables, TC_{w0}, TC^s_w in the objective function is then reformulated as follows:

$$TC_{w0} = \sum_{i \in \mathcal{I}} \sum_{k \in \mathcal{K}\backslash\{1\}} \sum_{t=1}^{L} c^w_i d^t_{ik} x^t_{ik}$$

$$TC^s_w = \sum_{n=1}^{N} \sum_{i \in \mathcal{I}, k \in \mathcal{K}\backslash\{1\}} \sum_{t=1}^{L} \gamma^n c^w_i dn^{ts}_{ikn} z^{ts}_{ikn}$$

After that, we reformulate the constraints with the indicator functions. Since the judgment condition of the indicator function in constraints (1f)–(1i) is not binary, it cannot be solved directly by commercial solvers. Auxiliary binary variables θ^t_{ik}, $h^{ts}_{ikn,}$ are introduced. Then, we use these variables to formulate the following constraints (4e)–(4j). Constraints (4e)–(4f) are equivalent to the constraints (1f) in the original model. Let

$$\min \quad TC_{w0} + TC_{d0} + TC_{oa}^1 + \sum\nolimits_{s\in\mathcal{S}} p^s (TC_w^s + TC_d^s + TC_{oa}^s) \tag{1a}$$

$$s.t. \quad \sum\nolimits_{t=1}^{W_1} x_{i1}^t + a_{i0}^r = R_{i0}^1, \ \ \sum\nolimits_{t=1}^{L} x_{ik}^t + a_{i0}^r = R_{i0}^k, \qquad \forall i \in \mathcal{I}, k \in \mathcal{K}_i \backslash \{1\} \tag{1b}$$

$$\sum\nolimits_{t=1}^{W_2} u_t + a_0 = F_0, \ \ u_t = 0, \qquad \forall\ t = W_2 + 1,\ ...,\ L \tag{1c}$$

$$\sum\nolimits_{t=n+1}^{n+W_1} z_{i1n}^{ts} + a_{in}^{rs} = r_{in}^{1s}, \ \sum\nolimits_{t=1}^{n} z_{i1n}^{ts} = 0,$$
$$\sum\nolimits_{t=n+1}^{n+L} z_{ikn}^{ts} + a_{in}^{rs} = r_{in}^{ks}, \ \sum\nolimits_{t=1}^{n} z_{ikn}^{ts} = 0 \qquad \forall i \in \mathcal{I}, k \in \mathcal{K}_i \backslash \{1\}, n \in \mathcal{N} \backslash \{0\}, s \in \mathcal{S} \tag{1d}$$

$$\sum\nolimits_{t=n+1}^{n+W_2} y_n^{ts} + a_n^s = f_n^s, \ \ \sum\nolimits_{t=n+1}^{n+L} y_n^{ts} + a_n^s = f_n^s, \qquad \forall n \in \mathcal{N} \backslash \{0\}, s \in \mathcal{S} \tag{1e}$$

$$x_{ik}^{t+m} - M\left[1 - 1\left(x_{i,k-1}^t > 0\right)\right] \le 0, \qquad \forall i \in \mathcal{I}, k \in \mathcal{K}_i \backslash \{1\}, t = 1, \ldots, L - \bar{v}_{ik},$$
$$1 \le m < \underline{v}_{ik}, \bar{v}_{ik} < m \le L - t \tag{1f}$$

$$x_{ik}^{t-m} - M\left[1 - 1\left(x_{i,k-1}^t > 0\right)\right] \le 0, \qquad \forall i \in \mathcal{I}, k \in \mathcal{K}_i \backslash \{1\}, t = 1, \ldots, L - \bar{v}_{ik},$$
$$0 \le m \le t - 1 \tag{1g}$$

$$z_{ikn}^{(t+m)s} - M\left[1 - 1\left(z_{i,k-1,n}^{ts} > 0\right)\right] \le 0, \qquad \forall i \in \mathcal{I}, k \in \mathcal{K}_i \backslash \{1\}, n \in \mathcal{N} \backslash \{0\}, t = n+1, \ldots, n$$
$$+ L - \underline{v}_{ik}, s \in \mathcal{S}, 1 \le m < \underline{v}_{ik}, \bar{v}_{ik} < m \le L - t \tag{1h}$$

$$z_{ikn}^{(t-m)s} - M\left[1 - 1\left(z_{i,k-1,n}^{ts} > 0\right)\right] \le 0, \qquad \forall i \in \mathcal{I}, k \in \mathcal{K}_i \backslash \{1\}, n \in \mathcal{N} \backslash \{0\}, t = n+1, \ldots,$$
$$n + L - \bar{v}_{ik}, s \in \mathcal{S},\ 0 \le m \le t - 1 \tag{1i}$$

$$o_t^o \ge b_1 u_t + \sum\nolimits_{i\in\mathcal{I}} \sum\nolimits_{k\in\mathcal{K}} \alpha_{ik} b_2 x_{ik}^t - r_t c_t, \qquad t = 1 \tag{1j}$$

$$o_t^{os} \ge b_1 u_t + \sum\nolimits_{i\in\mathcal{I}} \sum\nolimits_{k\in\mathcal{K}} \alpha_{ik} b_2 x_{ik}^t +$$
$$\sum\nolimits_{n\in\mathcal{N}\backslash\{0\}} \sum\nolimits_{k\in\mathcal{K}} \sum\nolimits_{i\in\mathcal{I}} b_1 y_n^{ts} + \alpha_{ik} b_2 z_{ikn}^{ts} - r_t c_t, \qquad \forall\ \ t = 1, \ldots, L, s \in \mathcal{S} \tag{1k}$$

$$a_t^o \ge r_t c_t - b_1 u_t - \sum\nolimits_{i\in\mathcal{I}} \sum\nolimits_{k\in\mathcal{K}} \alpha_{ik} b_2 x_{ik}^t, \qquad t = 1 \tag{1l}$$

$$a_t^{os} \ge r_t c_t - b_1 u_t - \sum\nolimits_{i\in\mathcal{I}} \sum\nolimits_{k\in\mathcal{K}} \alpha_{ik} b_2 x_{ik}^t -$$
$$\sum_{n\in\mathcal{N}\backslash\{0\}} \sum\nolimits_{k\in\mathcal{K}} \sum\nolimits_{i\in\mathcal{I}} \left(b_1 y_n^{ts} + \alpha_{ik} b_2 z_{ikn}^{ts}\right), \qquad \forall\ \ t = 1, \ldots, L, s \in \mathcal{S} \tag{1m}$$

$$o_t^e \ge \sum\nolimits_{i\in\mathcal{I}} \sum\nolimits_{k\in\mathcal{K}} \beta_{ik} b_3 x_{ik}^t - (1 - r_t) c_t, \ t = 1 \tag{1n}$$

$$o_t^{es} \ge \sum_{i\in\mathcal{I}} \sum_{k\in\mathcal{K}} \beta_{ik} b_3 x_{ik}^t + \sum_{i\in\mathcal{I}} \sum_{n\in\mathcal{N}\backslash\{0\}} \sum_{k\in\mathcal{K}} \beta_{ik} b_3 z_{ikn}^{ts} - (1 - r_t) c_t, \quad \forall\ \ t = 1, \ldots, L, s \in \mathcal{S} \tag{1o}$$

$$a_t^e \ge (1 - r_t) c_t - \sum\nolimits_{i\in\mathcal{I}} \sum\nolimits_{k\in\mathcal{K}} \beta_{ik} b_3 x_{ik}^t, \ t = 1 \tag{1p}$$

$$a_t^{es} \ge (1 - r_t) c_t - \sum_{i\in\mathcal{I}} \sum_{k\in\mathcal{K}} \beta_{ik} b_3 x_{ik}^t - \sum_{i\in\mathcal{I}} \sum_{n\in\mathcal{N}\backslash\{0\}} \sum_{k\in\mathcal{K}} \beta_{ik} b_3 z_{ikn}^{ts}, \quad \forall\ \ t = 1, \ldots, L, s \in \mathcal{S} \tag{1q}$$

$$u_t, x_{ik}^t, y_n^{ts}, z_{ikn}^{ts}, o_t^o, a_t^o, o_t^e, a_t^e, o_t^{os}, a_t^{os}, o_t^{es}, a_t^{es} \ge 0, \qquad \forall i \in \mathcal{I}, k \in \mathcal{K}_i, n \in \mathcal{N} \backslash \{0\}, t = n+1, \ldots,$$
$$n + L, s \in \mathcal{S} \tag{1r}$$

$$u_t, x_{ik}^t, y_n^{ts}, z_{ikn}^{ts}\ integer, \qquad \forall i \in \mathcal{I}, k \in \mathcal{K}_i, n \in \mathcal{N} \backslash \{0\}, t = n+1, \ldots,$$
$$n + L, s \in \mathcal{S} \tag{1s}$$

$$\theta_{ik}^t, h_{ikn,}^{ts} \in \{0, 1\}, 0 \le r_t \le 1, a_{i0}^r - R_{i0}^1 \le -1,$$
$$a_{in}^{rs} - r_{in}^{1s} \le -1, a_{i0}^r, a_0, a_{in}^{rs}, a_n^s \ge 0 \qquad \forall i \in \mathcal{I}, k \in \mathcal{K}_i, n \in \mathcal{N} \backslash \{0\}, t = n+1, \ldots,$$
$$n + L, s \in \mathcal{S} \tag{1t}$$

$\theta^t_{i,k-1} == 1\left(x^t_{i,k-1} > 0\right)$, which means if $x^t_{i,k-1} > 0$, then $\theta^t_{i,k-1} = 1$. Its inverse negative proposition is if $\theta^t_{i,k-1} = 0$, then $x^t_{i,k-1} \leq 0$, which can be formulated by the large M constraints (4f). Constraints (4e)–(4f) act as follows: if $x^t_{i,k-1} > 0$, that is, when there is a certain number of the (k-1)th appointment for type i revisit patients assigned to day t, then kth appointment for type i revisit patients cannot be assigned to day $t + m$. That is, the number of revisit patients assigned to those days that do not meet the requirement of revisit intervals is 0. Constraints (1g) can be equivalently reduced to the constraints (4f)–(4g) in a similar manner, which guarantees that kth revisit cannot be assigned before (k-1)th revisit. Likewise, constraints (4h)–(4j) ensure that two consecutive revisits for type-i revisit patients that generate on day $n \in \mathcal{N}\backslash\{0\}$ under scenario s are assigned to the two days to satisfy the revisit interval. Finally, constraints (1f)–(1i) are reformulated as the equivalent constraints (4e)–(4j). The big-M parameters in (4e)–(4j) take the same value because the same linearization structure is used. They are chosen large enough to ensure correct linearization without unnecessary relaxation. This is feasible because the patient demand allocation decisions are tightly bounded by constraints (1b)–(1e) and (1r)–(1t).

$$x^{t+m}_{i,k} - M\left(1 - \theta^t_{i,k-1}\right) \leq 0, \forall i \in \mathcal{I}, k \in \mathcal{K}\backslash\{1\}, \\ t = 1, \ldots, L - \bar{v}_{ik}, 1 \leq m < \underline{v}_{ik}, \bar{v}_{ik} < m \leq L - t \quad (4e)$$

$$x^t_{i,k-1} - M\theta^t_{i,k-1} \leq 0, \forall i \in \mathcal{I}, k \in \mathcal{K}\backslash\{1\}, t \in \mathcal{L} \quad (4f)$$

$$x^{t-m}_{i,k} - M(1 - \theta^t_{i,k-1}) \leq 0, \forall i \in \mathcal{I}, k \in \mathcal{K}\backslash\{1\}, \\ t = 1, \ldots, L - \bar{v}_{ik}, 0 \leq m \leq t - 1 \quad (4g)$$

$$z^{(t+m)s}_{i,k,n} - M(1 - h^{ts}_{i,k-1,n}) \leq 0, \ \forall i \in \mathcal{I}, k \in \mathcal{K}\backslash\{1\}, \\ n \in \mathcal{N}\backslash\{0\}, t = n + 1, \ldots, n + L - \bar{v}_{ik}, s \in \mathcal{S}, \\ 1 \leq m < \underline{v}_{ik}, \bar{v}_{ik} < m \leq N + L - t \quad (4h)$$

$$z^{ts}_{i,k-1,n} - Mh^{ts}_{i,k-1,n} \leq 0, \forall i \in \mathcal{I}, k \in \mathcal{K}\backslash\{1\}, \\ n \in \mathcal{N}\backslash\{0\}, t = n + 1, \ldots, n + L, s \in \mathcal{S} \quad (4i)$$

$$z^{(t-m)s}_{i,k,n} - M\left(1 - h^{ts}_{i,k-1,n}\right) \leq 0, \ \forall i \in \mathcal{I}, k \in \mathcal{K}\backslash\{1\}, \\ n \in \mathcal{N}\backslash\{0\}, \ t = n + 1, \ldots, n + L - \bar{v}_{ik}, \\ s \in \mathcal{S}, 0 \leq m \leq t - 1 \quad (4j)$$

B. DB-ACARP Algorithm

Since the model contains general integer variables, the problem is a stochastic non-convex programming. As the size of the problem increases, especially the number of scenarios, the number of variables and constraints increases. In our initial attempts, a general-purpose optimization solver Gurobi was implemented, but it was observed that the computation speed was rather slow in the SMIP, even for instances with five scenarios. As the sample size grows, one would anticipate an increase in the computational burden and solution time required to solve the model. Our model test instances can involve up to tens of millions of variables and constraints, making it an extremely large-scale problem. Using "throwing a model into a solver" approach is infeasible under an online resource allocation setting, as it fails to provide a feasible solution within the required time. To address this challenge, we propose a decomposition-based adaptive capacity allocation with revisit priority algorithm. The decomposition of the original stochastic mixed-integer programming model is implemented in two key ways. First, it separates the assigning of current realized demand from future stochastic demand, prioritizing the current-stage demand. Second, it decomposes the future demand assignment problem into scenario-based subproblems. The adaptiveness of the allocation lies in real-time tracking of allocated and unallocated demand, full-capacity utilization driven patient allocation, and dynamically updating capacity during the process. For decision period d, the algorithm framework is illustrated in the Fig. 5. In the DB-ACARP algorithm, we propose a revisit priority rule to ensure continuous care for revisits. This design choice is grounded in both clinical and operational considerations. As emphasized in the guidelines [56-57], timely revisits are essential for chronic disease management to ensure treatment adherence, monitor disease progression, and adjust treatment plans. Furthermore, operational policies in several hospitals, such as those outlined in [58], prioritize revisits to prevent treatment interruptions and maintain patient retention. This includes adaptive allocation heuristics for both first revisits (H1) and non-first revisits (H2), aiming to maximize service capacity utilization and fulfill the expected service time window as much as possible. Finally, execute the adaptive allocation heuristics for first visits (H3). For the realized demand in the current period and the stochastic demand in future periods, sequentially execute H1, H2, and H3. Repeat this process until the current decision stage shifts to the final predetermined decision period D.

(1) Adaptive Allocation Heuristic for First Revisit

This heuristic takes the initial service capacity as input and aims to assign as many first revisits as possible while adhering to capacity constraints and MWTTs. The output includes the number of each type of revisit patients accepted by the hospital and their corresponding plans. By iterating over each period, it allocates revisits and adjusts the available capacity to ensure efficient utilization of resources. Starting from period $t = 1$, calculate the number of first revisits currently already assigned and unassigned in real time. If there are still unmet first revisit demands and available capacity in the current period, set $x^k_{i,1}$ to the smaller of the following two values: the maximum number of first revisits that can be served by the available capacity in the current period and the remaining unmet first revisit demands. Repeat this process until all the first revisits are allocated. Then update the available capacity for each period. This heuristic is denoted as H1.

(2) Adaptive Allocation Heuristic for Non-First Revisits

This heuristic takes the first revisit assignment results and

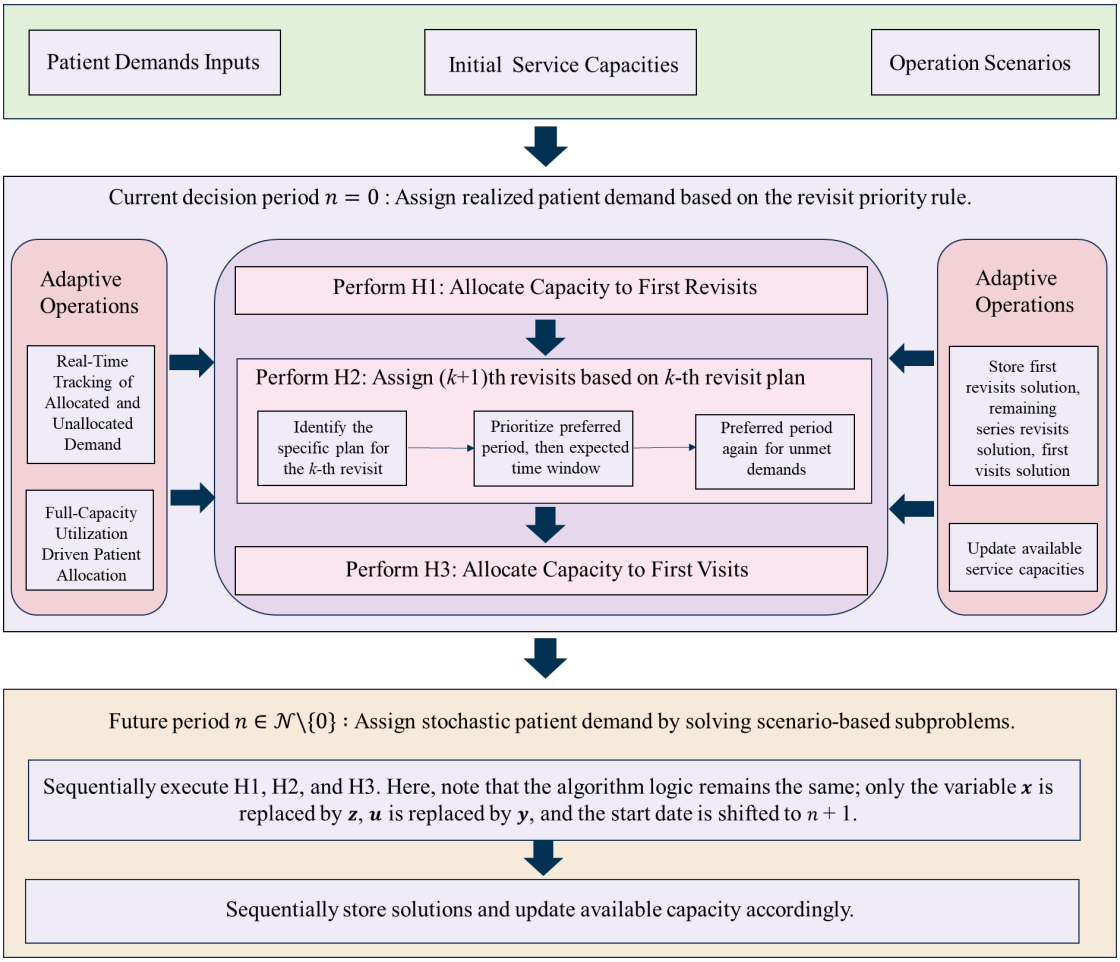


Fig. 5: Framework of the proposed DB-ACARP algorithm

Algorithm 1 Adaptive Allocation Heuristic for First Revisit

Input: Initial service capacity.
Output: The number of each type of revisit patients accepted by the hospital and their corresponding plans.
$\boldsymbol{C_{ava}} \leftarrow \boldsymbol{c}$
for $m = 1, \ldots, W_1$, **do**
  $temp \leftarrow \sum_{t=1}^{W_1} x_{i,1}^t$
  $R_1 \leftarrow R_{i0}^1 - temp$
  if $temp \geq R_{i0}^1$ **then**
    $x_{i,1}^m \leftarrow 0$
  else if $0 < C_{ava}^m \leq c_m$ **then**
    $x_{i,1}^m \leftarrow min(\lfloor C_{ava}^m/(\alpha_{ik}b_2 + \beta_{ik}b_3)\rfloor, R_1)$
  end if
  end if
end for
for $t = 1, \ldots, W_1$, **do**
  $C_{ava}^t \leftarrow C_{ava}^t - x_{i,1}^t(\alpha_{i1}b_2 + \beta_{i1}b_3)$
end for

the updated available capacity as input, aiming to assign non-first revisits while considering capacity constraints and expected service time windows. It identifies and records the specific day(s) for the k-th revisit and then assigns the next (k+1)-th revisit. The (k+1)-th revisit is prioritized within the patient's preferred period whenever possible. The assigned revisit count at this stage is stored in the completed assignment list, and the allocated and unallocated demands are dynamically updated. If the available capacity in the patient's preferred period is exhausted and there are remaining unallocated (k+1)-th revisit demands, the expected service time window for the (k+1)-th revisit is calculated. If there is available capacity in a certain period within the time window (excluding the patient's preferred day), the total number of patients that can be accommodated is determined, rounded down to an integer value. Take the smaller value between this and the unmet demand as the number of patients assigned for that period. The allocated (k+1)-th revisit count for that period is then recorded in the completed assignment list, and the demand status is updated in real time. If no available capacity exists within the entire time window (excluding the preferred day) and there are still unassigned (k+1)-th revisit demands, these demands are reassigned to the patient's preferred period. Repeat this process until all the non-first revisits are allocated. This heuristic is denoted as H2.

(3) Adaptive Allocation Heuristic for First Visits

This heuristic takes the updated available capacity after the revisit assignment results as input, aiming to assign first visits

Algorithm 2 Adaptive Allocation Heuristic for Non-First Revisits

Input: First revisit assignment results and the updated available capacity.
Output: Non-first revisits allocation plans.
for k = 1, …, $K-1$, **do**
  for t = 1, …, L, **do**
    if $x_{ik}^t > 0$ **then**
      $rd_k \leftarrow t$
      ComSch $\leftarrow \varnothing$
      $num \leftarrow 0$
      $pre \leftarrow \tau_{i,k+1}$
      if $0 < C_{ava}^{pre} \le c_{pre}$ **then**
        num $\leftarrow \min(\lfloor C_{ava}^{pre}/(\alpha_{i,k+1}b_2 + \beta_{i,k+1}b_3), x_{ik}^t)$
        Add num to ComSch
        $x_{i,k+1}^{pre} \leftarrow x_{i,k+1}^{pre} + num$
        $C_{ava}^{pre} \leftarrow C_{ava}^{pre} - num(\alpha_{i,k+1}b_2 + \beta_{i,k+1}b_3)$
      end if
      $a \leftarrow rd_k + \underline{v}_{i,k+1}$
      $b \leftarrow rd_k + \bar{v}_{i,k+1}$
      for $j = a, \ldots, b$, **do**
        $temp \leftarrow$sum (ComSch)
        $rs \leftarrow x_{ik}^t - temp$
        if $temp > x_{ik}^t$ and $j \neq pre$ **then**
          $num' \leftarrow 0$
          $x_{i,k+1}^j \leftarrow x_{i,k+1}^j + num'$
        end if
        if $temp < x_{ik}^t$ *and* $0 < C_{ava}^j \le c_j$ *and* $j \neq pre$ **then**
          num$' \leftarrow min(\lfloor C_{ava}^j/(\alpha_{i,k+1}b_2 + \beta_{i,k+1}b_3), rs)$
          Add num' to ComSch
          $x_{i,k+1}^j \leftarrow x_{i,k+1}^j + num'$
          $C_{ava}^j \leftarrow C_{ava}^j - num'(\alpha_{i,k+1}b_2 + \beta_{i,k+1}b_3)$
        end if
      end for
      $temp' \leftarrow$ sum (ComSch)
      if $temp' < x_{ik}^t$ **then**
        $rs' \leftarrow x_{ik}^t - temp'$
        $x_{i,k+1}^{pre} \leftarrow x_{i,k+1}^{pre} + rs'$
        $C_{ava}^{pre} \leftarrow C_{ava}^{pre} - rs'(\alpha_{i,k+1}b_2 + \beta_{i,k+1}b_3)$
      end if
    end if
  end for
end for

as much as possible under capacity constraints and expected service time windows. It continuously tracks the allocated and unallocated demand in real time. If capacity is available on a given day, it calculates the maximum number of patients that can be served based on the available capacity and rounds down the value. The smaller value between this number and the number of first visits that remain to be allocated is taken as the number of patients assigned to that day. Continue in this manner until all first visits have been assigned. Finally, update the available capacity for each period accordingly. This heuristic is hereafter referred to as H3.

Algorithm 3 Adaptive Allocation Heuristic for First Visits

Input: The updated available capacity after the revisit assignment results.
Output: First visits assignment plans.
for $m = 1, \ldots, W_2$, **do**
  $temp \leftarrow \sum_{t=1}^{W_2} u_t$
  $F_1 \leftarrow F_0 - temp$
  if $temp \ge F_0$ **then**
    $u_m \leftarrow 0$
  else if $0 < C_{ava}^m \le c_m$ **then**
    $u_m \leftarrow min(\lfloor C_{ava}^m/b_1 \rfloor, F_1)$
  end if
  end if
end for
for $t = 1, \ldots, W_2$, **do**
  $C_{ava}^t \leftarrow C_{ava}^t - u_t b_1)$
end for

(4) Solve Sub-problems

Taking the updated available capacity after assigning the realized demand of the current period as input, this heuristic aims to assign the stochastic first revisits, non-first revisits, and the first visits that arrive on day $n \in \mathcal{N}\backslash\{0\}$. For each scenario-based subproblem, conducting the same algorithm logic as H1, H2 and H3, respectively, except changing the start date to n+1. And the available capacities are dynamically updated accordingly. Finally, compute the expected performance across all scenarios as the solution to the original problem.

V. Numerical Studies

In this section, numerical studies are implemented to evaluate how well the proposed DB-ACARP and optimization models perform. In the following part, we first explain the experiment settings of the testing instances. Then, experiments are designed and conducted to verify the performance of DB-ACARP by comparing it with Gurobi. Next, the effectiveness of the suggested multi-stage stochastic optimization model is then verified by comparison with alternative model. Finally, we implement sensitive analysis and discuss the numerical results to provide some managerial insights.

A. Data Settings

Based on operational data extracted from an online and offline medical service provider in Shanghai, we create test instances. From the data analysis results, revisit patients can

generally be categorized into three types. The first category involves continuous online revisits for medication prescriptions, the second requires alternating offline and online revisits (one offline followed by one online), and the third consists of continuous offline revisits with examinations. This setting allowed us to examine regularities in capacity allocation across online and offline services and different revisit patient types. Our model can easily accommodate more than these three revisit types by expanding the type set parameter without altering the core algorithm, and the allocation results can approximate more diverse chronic revisit systems when most patients follow these three disease progression patterns. The service capacity and cost units mentioned below are denoted as "pu" and "mu" respectively. We generate our instances by setting the length of booking horizon L to be 35 days and fixing the planning horizon to be 5 days, $\mathcal{N} = \{0, \ldots, 4\}$. Since the size of a full multistage scenario tree grows exponentially, the model is practically intractable due to the curse of dimensionality. We use a rolling-horizon approximation of a multistage stochastic program, where the future uncertainty is represented by Monte-Carlo scenario sampling rather than an explicit scenario tree. This provides a computationally tractable approximation while preserving the non-anticipativity and multistage information structure. Specifically, we generate the demand of first visit patients f_n^s for all periods $n \in \mathcal{N}\backslash\{0\}$ and scenarios $s \in \mathcal{S}$ from a Poisson distribution with rate λ_f = 12.49. The first-time revisit demand r_{in}^{1s} is generated from Poisson distributions with rates λ_{ri} = 19.03, 25.8, 7.56 for i = 1, 2, 3. Subsequent revisit demand $r_{in}^{ks},\ k = 2, ..., K_i$, is set equal to r_{in}^{1s} across all periods and scenarios. All generated values are rounded to the nearest integers, and a minimum demand of 1 is enforced. For each period from period 1 to $N-1$, we generate S random demand scenarios, each occurring with equal probability $1/S$. Across these $N-1$ future periods, the scenarios form S scenario paths spanning from stage 0 to stage $N-1$.

The first appointment of the revisit patient MWTTs is W_1= 3 and the first visit patient MWTT is W_2= 8. The allowance span of offline (online) revisit interval is set to be $[\underline{a}^o,\ \overline{a}^o] = [2,\ 4]$ ($[\underline{a}^e,\ \overline{a}^e] = [4,\ 6]$). The three types of revisits occur 6, 6, and 8 times, respectively. The preferred day is generated for each visit of each patient category using calibrated probabilities. According to the operational data, the mean service time of an offline first visit, offline revisit, and online revisit is 40, 30, and 20. We set the regular daily capacity c_t to 1440. Through discussions and interactions with hospital managers, we learned that the hospital prioritized overtime over idle time, assigning it twice the weight. Revisit patients were emphasized to ensure continuity of care, with the weight ratio of revisit to first-visit transfers set at 2:1.5. Patient preference was assigned a weight of 1, as the hospital prioritizes improving overall service accessibility before addressing individual patient preferences. Therefore, unit overtime cost, idle cost, rejection or diverting cost per revisit, per first visit, and delay or early cost of revisit per day is set to 2:1:2:1.5:1. We set the instances described above to be base settings. Each instance executed in the following experiments consists of a five-stage rolling-horizon re-optimization decision process. At beginning of each period, the allocated staffing, service capacity, and patient demand are updated. Once the realized information becomes available, the proposed stochastic model is re-solved to obtain the decision for that period. Then, the current stage decision is implemented, and the system moves to the next period and the process repeats. Since we operate in a dynamic, multi-stage decision-making setting, we focus more on short-term costs (TC_5) of first five decision stages, as these represent the costs incurred following the implementation of decisions. However, for a comprehensive analysis, we also evaluate the long-term costs (TC_{30}) over the first 30 periods and the overall objective function values across five decisions (TC_{obj}). In addition, to investigate how different parameters affect the performance of the proposed model, we generate new instance sets by varying patient demand, capacity level and pattern, MWTTs, and revisit interval.

B. Algorithm Performance

The proposed DB-ACARP algorithm performance analysis is presented in this subsection. We run some test instances to compare the performances of the proposed DB-ACARP and the multistage stochastic programming model directly resolved by Gurobi. The mathematical programming problem is implemented using Gurobi 11.0.1 with Python 3.11.5 on a PC equipped with a 13th Gen Intel(R) Core (TM) i7-13700H processor, featuring 14 physical cores and 20 logical processors, utilizing up to 20 threads. It is noted that the proposed multistage stochastic programming model is applied in an online allocation scenario. The performance metric of computation time is important in such settings, where the parameters are updated in each stage and the model must generate results within a short time. Therefore, we set a time limit of 1800 seconds for each model run, leading to a total time limit of 9000 seconds for the five-stage decision-making process. The MIPGap parameter in Gurobi is set to 0.05%.

We consider 9 low-demand, 5 medium-demand, and 5 high-demand instances, where the patient arrival rate is 0.25 times, 0.5 times, and 1.0 times the baseline level, chosen with consideration of model stability and computational efficiency. To evaluate the algorithm's performance under varying levels of resource constraints, and considering the dynamic nature of real-world outpatient systems where capacity is not fixed due to frequent staffing adjustments by managers, we set different capacity levels for each level of patient arrival rate. The number of scenarios varies across instances as well. The generated instances are denoted as "D[L/M/H]-S[b]-C[c]". Table II presents the scale of some of our test instances. For our largest test instance, the model size is extremely large, with the number of variables and constraints reaching tens of millions. The proposed model is an NP-hard problem. The solution space grows exponentially as the problem size increases. In Table III, the performance value, computation time (denoted as "Cot"), and performance improvement of the proposed DB-ACARP algorithm compared with Gurobi are reported in the "Gurobi" and "DB-ACARP" columns, respectively. We denote Imp-I = (TC(Gurobi) - TC(DB-ACARP))/TC(Gurobi) ×100%, I = 1, 2, 3 to represent the improvement in each performance

indicator. Here, TC(Gurobi) and TC(DB-ACARP) represent the values of each performance indicator obtained by Gurobi and DB-ACARP, respectively. We use "ABgap" to represent the average gap between the solutions obtained by Gurobi for five model runs within the time limit and the optimal solution.

TABLE II: Model size of some test instances

Instances	Size			
	Continuous variables	Integer variables	Total variables	Total constraints
DBL-S8-C1800	32776	62867	95643	899821
DBL-S15-C1800	60664	116375	177039	1665733
DBL-S30-C1800	120424	231035	351459	3306973
DBL-S100-C1800	399304	766115	1165419	10966093
DBL-S150-C1800	598504	1148315	1746819	16436893
DBL-S200-C1800	797704	1530515	2328219	21907693

The results indicate that for both short-term costs (TC_5) and long-term costs (TC_{30}), DB-ACARP consistently outperforms Gurobi, with average improvements of 48.6% and 19.4%, respectively. For all test cases, Gurobi fails to provide an exact solution within the time limit, with the lowest ABgap being 1.3%. In some instances, such as "DM-S8-C800", "DH-S5-C2200", and "DH-S8-C1800", Gurobi even fails to find a feasible solution due to an "out of memory" error. However, DB-ACARP provides better solutions in a very short time, even for large scale problems with 2000 scenarios, such as within 60 seconds, especially in terms of short-term and long-term costs. In terms of overall objective value, DB-ACARP performs worse than Gurobi. This is likely due to the optimization separates current-period decisions from future random demand. The algorithmic framework does not incorporate future-period demand information when optimizing the current decision. As a result, DB-ACARP performs better on indicators related only to the demand raised in current decision period, such as TC_5 and TC_{30}, but less effectively on the overall objective value indicator, TC_{obj}, which is influenced by both current and future period demand. The rule-based heuristics can be strengthened in several ways. Solution quality may be improved using variable-neighborhood search with simulation-based evaluation and local repair, or by adopting ensemble strategies that combine greedy, balanced, and priority-based rules. A hybrid exact–heuristic approach, which solves the current-period decision optimally using a MIP solver while applying heuristics for future periods, can also be effective. Moreover, the framework can incorporate value-function approximations for future periods through forward–backward iterations, such as cutting-plane or Benders-type methods. With enhanced scenario-sampling techniques (e.g., importance sampling, stratified sampling, or scenario aggregation), such procedures may yield convergence guarantees and improve policy quality.

To more precisely diagnose why the proposed algorithm underperforms in terms of the overall objective value and how this affects the solution allocations, we compare its results with Gurobi's optimal solution on instance DL-S2-C500 for a single run (i.e., allocation for first-period realized demand and four subsequent stages stochastic demand). Table IV summarizes the comparison of the cost components. We observe that, compared with the approximate solution of the proposed algorithm, the Gurobi optimal solution accepts more patients overall, rejects more first-visit patients, and ensures that all revisit patients are served. To achieve this, it uses more overtime capacity and leaves less idle capacity. In terms of costs, the approximate solution reduces overtime costs by 93.9% and preference-related costs by 47.3%, but incurs an additional 1955 units of idle cost. Since these savings cannot offset the higher idle and rejection costs, the overall objective value TC'_{obj} of the approximate solution is higher. Fig. 6 illustrates the allocation plans. The impact on allocation is that the proposed algorithm adopts conservative allocation strategies that allocates less total capacity in each period to avoid overtime, which is particularly important in healthcare systems with constrained capacity in practice. Here, TC'_{obj} denotes the objective of a single run considering overtime and idle costs without discounting future cost, which distinguishes it from TC_{obj} in Table III. Since we adopt a multi-stage decision framework and are more concerned about costs incurred after the execution of decisions (TC_5 and TC_{30}) in real-world applications, and given DB-ACARP's extremely fast solving speed for large scale problems, we can conclude that the proposed DB-ACARP algorithm has significant advantages and practical application value. Our findings also suggest that the proposed algorithm fits healthcare systems that value overtime reduction and patient preference satisfaction more than minimizing idle resources and patient rejections.

C. Performance of the Rolling-Horizon Multistage Stochastic Framework

1) Value of Modeling Subsequent Stochastic Demands

This study's explicit modeling of the stochastic nature of patients who will arrive on subsequent days is a significant contribution. An alternative approach is to make decisions at the current stage while ignoring the stochastic nature of future patient appointments. Through numerical experiments, we evaluate the benefits of explicitly incorporating this uncertainty into the decision-making process. In order to do this, we assume that the daily demand coming from future days is 0 (that is, Poisson with $\lambda_f = \lambda_{ri} = 0$) as comparisons. To investigate the impact of the number of look-ahead steps on system performance, we further conducted experiments for each instance with stochastic demands one and two steps into future. Note that we conduct the above experiments using Gurobi within a five-period rolling-horizon re-optimization framework. The time limit for each decision run is set to 200 seconds, a total of 1000 seconds for five runs. We also solve the proposed model with DB-ACARP. The results are summarized in Table V using the short-term cost measure, i.e., the total costs over five decision periods. Columns "$TC_{5\text{-}NSD}$", "$TC^1_{5\text{-}SD}$", and "$TC^2_{5\text{-}SD}$" show the model's performance without accounting for future demand, and with stochastic demands considered one and two steps ahead, respectively. The proposed model was solved using the proposed method and Gurobi, with the corresponding costs presented in columns "$TC^{PM}_{5\text{-}SD}$" and "$TC^4_{5\text{-}SD}$". $D_i = (TC_{5\text{-}NSD} - TC^i_{5\text{-}SD})/TC_{5\text{-}NSD} \times 100\%, i = 1, 2, 4$, represents the percentage reduction in cost achieved by considering

TABLE III: The performance comparison results of Gurobi and the DB-ACARP

No.	Instances	Gurobi					DB-ACARP				Imp-1	Imp-2	Imp-3
		TC_5	TC_{30}	TC_{obj}	ABgap	CoT(s)	TC_5	TC_{30}	TC_{obj}	CoT(s)			
1	DL-S2-C500	178	4878	20299	17.1	7260	120	4010	28957	0.1	32.6	17.8	-42.7
2	DL-S5-C600	575	7715	26690	6.1	4124	451	7141	46587	0.2	21.6	7.4	-74.5
3	DL-S5-C700	1180	9780	34321	4.4	9067	980	9450	55413	0.1	16.9	3.4	-61.5
4	DL-S2-C550	193	5433	23425	13.9	5499	91	4241	30552	0.1	52.8	21.9	-30.4
5	DL-S2-C450	335	4255	18615	24	9087	80	2860	25628	0.1	76.1	32.8	-37.7
6	DL-S2-C750	1321	10821	36509	1.3	9036	612	7692	38451	0.1	53.7	28.9	-5.3
7	DL-S3-C550	619	5709	23000	13.8	7291	413	5143	35398	0.1	33.3	9.9	-53.9
8	DL-S2-C520	381	5781	20365	13.1	4971	206	4176	28878	0.1	45.9	27.8	-41.8
9	DL-S3-C600	325	6145	28370	10.9	3704	257	5057	40112	0.1	20.9	17.7	-41.4
10	DM-S2-C800	618	6438	38024	35.6	9035	96.5	4846.5	43769	0.1	84.4	24.7	-15.1
11	DM-S2-C900	733	9153	31886	24.4	9028	139	5979	52489	0.1	81	34.7	-64.6
12	DM-S3-C850	932	10472	27781	18.9	9041	570	9210	60382	0.1	38.8	12.1	-117.3
13	DM-S3-C950	524	9894	33252	24.2	9040	215	7265	63702	0.1	59	26.6	-91.6
14	DM-S8-C800	-	-	-	-	-	93	4343	52743	0.2	-	-	-
15	DM-S1000-C800	-	-	-	-	-	90	4740	60041	27.1	-	-	-
16	DM-S2000-C800	-	-	-	-	-	90	4610	63214	54.2	-	-	-
17	DH-S2-C2000	1773	20673	63590	16.1	9029	659	17439	115705	0.1	62.8	15.6	-82
18	DH-S3-C2100	1471	25791	66017	18.6	9043	748	23468	141950	0.1	49.2	9	-115
19	DH-S3-C1900	2038	22998	63899	26.7	9042	422	14162	124792	0.1	79.3	38.4	-95.3
20	DH-S5-C2200	-	-	-	-	-	1523	25473	169240	0.1	-	-	-
21	DH-S8-C1800	-	-	-	-	-	434	15334	138277	0.2	-	-	-
22	DH-S1000-C1800	-	-	-	-	-	397	15057	150566	26.8	-	-	-
23	DH-S2000-C1800	-	-	-	-	-	163	9703	135567	54.6	-	-	-

Note: Instances are denoted as D[L/M/H]-S[b]-C[c], where D = demand level (L = Low, M = Medium, H = High), S = number of scenarios, and C = capacity level. For example, DL-S2-C500 represents an instance with low demand, 2 scenarios, and a capacity level of 500.

TABLE IV: Comparison of cost components and overall objective values between Gurobi and the DB-ACARP

Method	C_{To}	C_{Ta}	C_{Tf}	C_{Tr}	C_{Tfn}	C_{Trn}	C_{Rt}	C_{Rtn}	TC'_{obj}
Gurobi	1320	9465	0	0	25.1	0	4	32.9	10814.1
DB-ACARP	80	11420	0	0	9.1	31.9	0	17.3	11558.4

Note: C_{To} = overtime cost; C_{Ta} = idle cost; C_{Tf} = realized first-visit rejection cost; C_{Tr} = realized revisit rejection cost; C_{Tfn} = future stochastic first-visit transfer cost; C_{Trn} = future stochastic revisit transfer cost; C_{Rt} = realized patient preference cost; C_{Rtn} = future stochastic patient preference cost; TC'_{obj} = total objective value without discounting future overtime and idle costs.

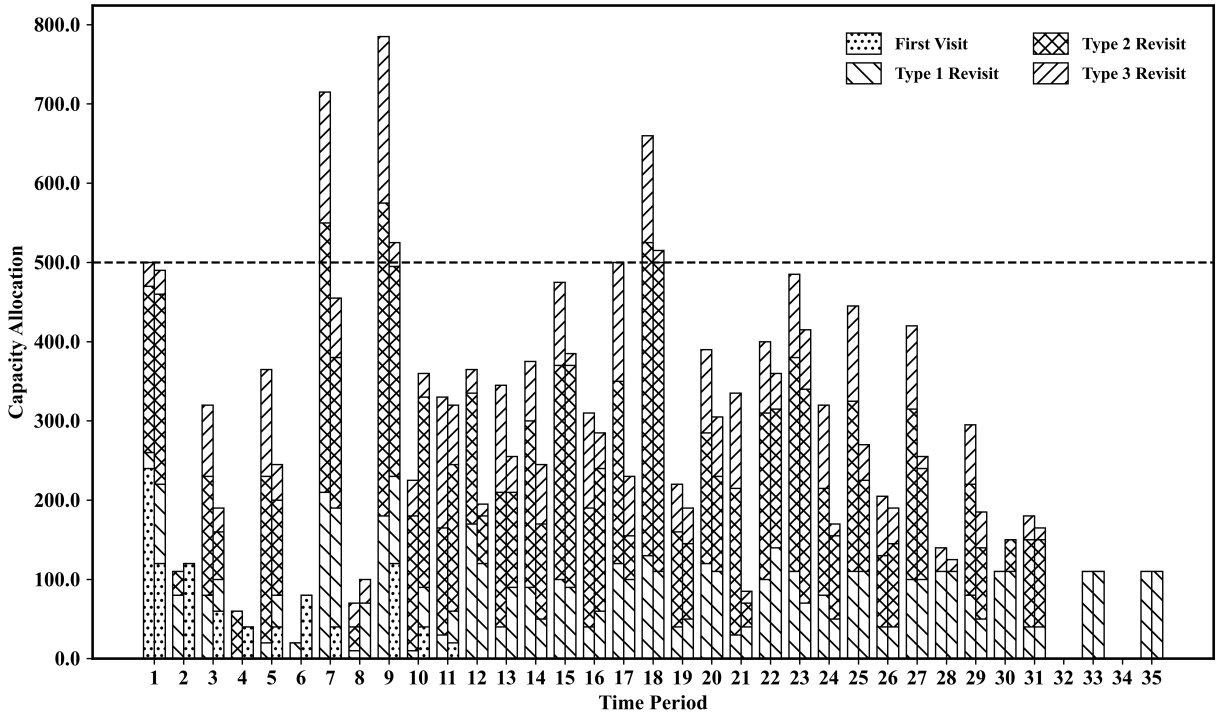


Fig. 6: Stacked bar comparison of capacity allocation for four patient types under Gurobi solution and DB-ACARP solution over 35 time periods. Each time period has two side-by-side bars (left for 'Gurobi solution' and right for 'DB-ACARP solution'). Hatch patterns differentiate patient types. The dashed line shows $y = 500$ reference.

TABLE V: Value of modeling subsequent stochastic demands

Instances	Demand	Scenarios	Capacity	$TC_{5\text{-}NSD}$	$TC^{1}_{5\text{-}SD}$	$TC^{2}_{5\text{-}SD}$	$TC^{4}_{5\text{-}SD}$	$TC^{PM}_{5\text{-}SD}$	$D_1\%$	$D_2\%$	$D_4\%$
1	H	3	1000	3237.5	1491	1879	3113	389.5	53.9	41.9	3.8
2	H	3	800	4755.5	3473	3912.5	3887	419	26.9	17.7	18.3
3	M	2	300	2384.5	2536	1842	1575	258	-6.3	22.7	33.9
4	M	2	250	3186.5	2598.5	1530	1733	271	18.4	51.9	45.6
5	L	2	250	1010.5	667	347	439.5	111	33.9	65.6	56.5
6	L	2	200	1269.5	1138	726	1163.5	129	10.3	42.8	8.3

one-step, two-step, and four-step future information, compared to the solution without such consideration.

We find that assigning patients at the current stage while explicitly considering stochastic appointments one, two, and four steps ahead reduces operational costs by an average of 22.8%, 40.4%, and 27.8%, respectively, compared to optimizing independently. Considering two steps ahead yields the best average performance. Notably, in Instance 3, the one-step-ahead model performs worse than the model that ignores stochastic future demand. These indicate that incorporating more future steps does not necessarily improve results. This may be due to accumulated prediction errors. As the number of steps increases, uncertainty grows, potentially misleading the optimization. Additionally, optimizing over more steps increases problem complexity, which can compromise short-term stability. Healthcare practitioners must balance prediction accuracy, model complexity, and system robustness, choosing an appropriate number of look-ahead steps to achieve optimal system performance. Our method consistently performs best under this metric, owing to its adaptive response to demand and flexible capacity allocation decisions. During the experiment, we observed that under the given time constraints, TC_{5-NSD} consistently achieved the optimal solution due to its lower model complexity. In contrast, TC_{5-SD} exhibited a certain gap from the optimal solution. In such cases, the value of TC_{5-SD} was lower than that of TC_{5-NSD}. Then, with longer runtime, the proposed model is expected to perform better as it would yield more precise solutions. In capacity-limited systems with low daily capacity (e.g., Instances 2 and 4), considering stochastic future arrivals enables more effective use of limited resources than models ignoring them, which is crucial in practice.

2) Rolling-Horizon Multistage Model vs. Benchmarks

In this subsection, we compare the operational performance of the two-stage (TS-SAA), multistage (MS-SAA), and deterministic (DM) models on instance DL-S2-C500. Given that the proposed model with a large sample of scenarios is intractable, we adopt a small-sample repeated-sampling approach for pseudo out-of-sample evaluation. Case 1 samples two scenarios for each instance, and Case 2 samples four. Eight repetitions are performed for each case, using distinct random seeds across instances and periods. The multistage model is executed under a full five-period rolling-horizon stochastic optimization, while the two-stage model solves only once for the initial optimization. The deterministic model is solved separately for each scenario, and the corresponding average cost is computed. The running time for each optimization is limited to 360 seconds, yielding a total of 1800 seconds for MS-SAA. TS-SAA and DM achieve optimal solutions in the vast majority of instances. Results are summarized under the columns "TS-SAA," "MS-SAA," and "DM" in Table VI.

The MS-SAA models yield significantly lower total costs on average under both cases than TS-SAA and DM. In Case 1, MS-SAA reduces costs by 11.4% and 14.4% compared with TS-SAA and DM, respectively. With an increased number of sampled scenarios, the improvement is more pronounced. In Case 2, the reductions increase to 13.2% and 15.4%. These results demonstrate the benefits of multistage stochastic models within a rolling-horizon implementation framework, particularly when incorporating more lookahead demand information. This is because multistage models offer greater flexibility, allowing decisions to adapt patient arrivals over time compared with two-stage models. They capture temporal information and support dynamic adjustment strategies. The deterministic model exhibits the worst performance, highlighting the value of explicitly modeling uncertainty through stochastic optimization.

D. Sensitive Analyses

Several factors can affect integrated online and offline outpatient systems in the studied healthcare settings. Capacity volume, capacity pattern, period parameter, and cost coefficient, are important examples. In this section, experiments are conducted to analyze the effects of the above parameters on the solution and system performances.

1) Impacts of Capacity Volume and Capacity Pattern

Capacity volume and capacity pattern are key factors influencing system performance. To examine the impact of capacity volume, we designed two groups of experiments, each representing a system of a different scale. Each group contains eight instances, with varying capacity levels within each instance. These instances are labeled as "D[L/M/H]-C[c]". Then, to investigate the impact of capacity patterns, we conducted ten groups of experiments featuring different demand patterns. Each group consists of four instances, each configured with various capacity patterns. Groups 1–5 and Groups 6–10 represent two systems of different scales. Groups 1–5 are set with Demand Pattern 1–5 and Capacity Pattern 1–4, as illustrated in Fig. 7, while Groups 6–10 follow Demand Pattern 6–10 and Capacity Pattern 5–8, as shown in Fig. 8. The results of the three performance evaluation metrics (TC_5, TC_{30}, and TC_{obj}) are presented in Tables VII and VIII.

From Table VII, it can be observed that in both groups, TC_5 initially decreases as capacity increases. However, after reaching a certain capacity (C1800 for Group 1 and C800 for Group 2), TC_5 rises sharply. This suggests that there exists an optimal capacity range that minimizes short-term costs. Exceeding this range leads to rising short-term costs.

TABLE VI: Comparative results of rolling-horizon multistage stochastic model and benchmarks

Case	Approach	Ins 1	Ins 2	Ins 3	Ins 4	Ins 5	Ins 6	Ins 7	Ins 8	Average
1	TS-SAA	10847	11111	10691	10645	10067	10765	10772	10174	10634
	MS-SAA	9833	9722	9168	9206	9411	9229	9186	9623	9422
	DM	12077	11186	10899	10669	10856	10958	10871	10529	11005
2	TS-SAA	11150	10144	10912	10666	10839	10592	11085	10666	10757
	MS-SAA	9181	9169	9610	9424	9244	9419	9193	9437	9335
	DM	11268	10657	10985	10776	10950	10846	11181	11591	11032

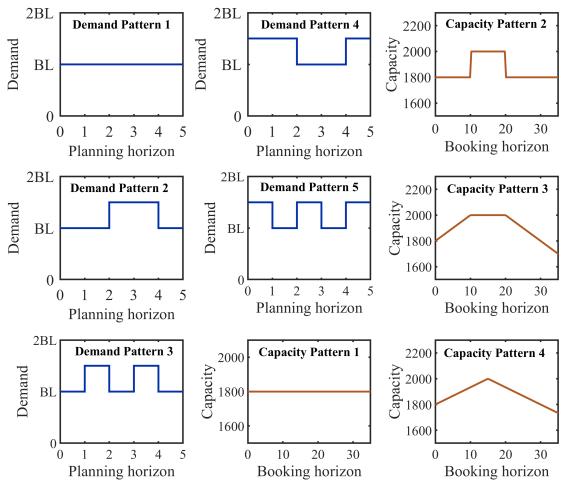


Fig. 7: Varied demand and capacity patterns used for simulation experiments (groups 1–5)

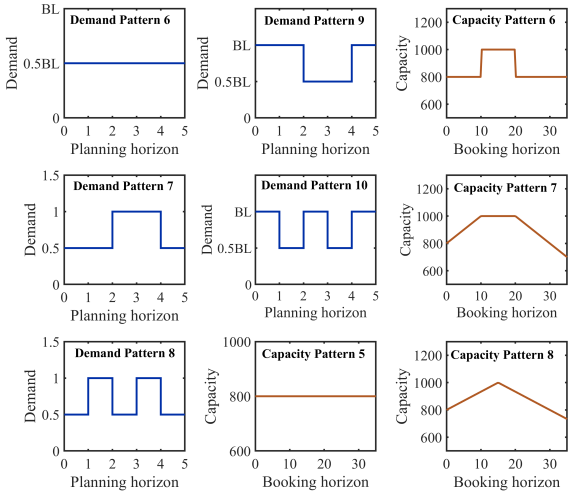


Fig. 8: Varied demand and capacity patterns used for simulation experiments (groups 6–10)

Both groups show a steady increase in TC_{30} as capacity grows. Group 1 exhibits a more gradual rise, while Group 2 experiences steeper increases after C1200, reflecting higher long-term operational challenges in larger systems. Overall objective values increase with capacity in both groups. Group 1's TC_{obj} rises more rapidly after C1600, while Group 2 maintains a more moderate growth, suggesting better cost efficiency at higher capacities. Here, we only provide managerial insights on optimizing capacity levels to enhance short-term efficiency, as evaluating TC_5 is more aligned with practical applications.

Insight 1. *Since TC_5 decreases initially with increased capacity in both groups, managers should identify and operate within the optimal capacity range to maximize short-term cost efficiency.*

From the result in Table VIII, we can observe the following trends for TC_5, TC_{30}, and TC_{obj} across all groups. For TC_5, in most groups, TC_5 generally exhibits an initial decline as capacity patterns vary, reaching the minimal cost when the capacity pattern first increase, remains constant, and then gradually decreases, such as in Capacity Pattern 3 and Capacity Pattern 7. However, the rate and extent of this decrease vary depending on the demand patterns and group settings. This suggests that optimizing capacity patterns can significantly impact short-term costs, although the relationship might not be strictly linear across all scenarios. For TC_{30}, TC_{30} demonstrates more variable behavior. In some groups (e.g., Groups 1–3), it decreases with adjusting capacity pattern, while in others (e.g., Groups 4-5), it fluctuates without a clear trend. This suggests that long-term cost efficiency is influenced by both capacity patterns and the nature of demand patterns, and may not always align with short-term cost trends. For TC_{obj}, TC_{obj} generally shows a slight increase or fluctuates as capacity pattern varies. This indicates that while short-term costs can be minimized through capacity optimization, the overall cost performance may be more sensitive to other factors.

For each group, given a demand pattern, there is an optimal capacity pattern that yields the best performance for different performance indicators. For the system represented by Groups 1–5 and Groups 6–10, there are optimal combinations of demand and capacity patterns that yield the best performance for different performance indicators. For instance, for the system represented by Groups 1–5, the lowest TC_5 (77) occurs in Group 1, Demand Pattern 1, Capacity Pattern 3; the lowest TC_{30} (9982) is in Group 3, Demand Pattern 3, Capacity Pattern 2; and the lowest TC_{obj} (105439) appears in Group 4, Demand Pattern 4, Capacity Pattern 1. Therefore, we can also obtain useful and interesting combined insights.

Insight 2.1. *For a given system capacity should not remain constant over time; instead, managers can frequently adjust staffing to align with fluctuating demand.*

Insight 2.2. *Managers can identify and implement the optimal capacity pattern for a given demand pattern, as well as the optimal combinations of demand and capacity patterns, to minimize various performance indicators.*

Insight 2.3. *While TC_5 and TC_{30} often follow similar trends of initial decrease with capacity adjustments, TC_{obj} does not always reflect these improvements, highlighting the complexity of balancing immediate and future performance.*

TABLE VII: Impact of capacity level on different system performance indicators

Group	Instances	TC_5	TC_{30}	TC_{obj}	Group	Instances	TC_5	TC_{30}	TC_{obj}
1	DH-C800	419	7379	49806	2	DM-C200	304	2243	12697
	DH-C1000	390	7789	59526		DM-C400	247	3217	23964
	DH-C1200	313	9152	73403		DM-C600	185	4134	37443
	DH-C1400	221	10251	90303		DM-C800	78	5067	56435
	DH-C1600	178	8918	108294		DM-C1000	263	7383	77315
	DH-C1800	105	12155	131226		DM-C1200	822	13352	100169
	DH-C2000	139	15439	153403		DM-C1400	1810	19340	123059
	DH-C2200	634	21414	176584		DM-C1600	2810	25340	146181

Note: Instances are denoted as D[L/M/H]- C[c], where D = demand level (L = Low, M = Medium, H = High) and C = capacity level. For example, DH-C800 represents an instance with high demand, and a capacity level of 800.

TABLE VIII: Sensitivity analysis with different demand and capacity patterns

Group	Demand Pattern	Capacity Pattern	TC_5	TC_{30}	TC_{obj}	Group	Demand Pattern	Capacity Pattern	TC_5	TC_{30}	TC_{obj}
1	(1)*	(1)	105	12155	131226	6	(6)*	(5)	109	4579	52343
		(2)	104	11694	135752			(6)*	86*	4856	57619
		(3)*	77*	13427	141321			(7)	90.5	4531	61729
		(4)	105.3	12722	138565			(8)*	120.8	4497*	59339
2	(2)	(1)	149	11749	126320	7	(7)	(5)	300	4930	49159
		(2)	137	10587	130136			(6)	283	4903	53552
		(3)	108	11478	135392			(7)	260.5	4961	56767
		(4)	135.3	11602	133173			(8)	297.3	4964	55035
3	(3)*	(1)	137.5	10438	122630	8	(8)	(5)	324.5	5035	47644
		(2)*	141.5	9982*	127123			(6)	301.5	5242	52564
		(3)	126	10596	131929			(7)	250.5	14951	55316
		(4)	131.3	10508	129655			(8)	266.3	5283	53897
4	(4)*	(1)*	406.5	12387	105439*	9	(9)*	(5)*	370.5	6051	43718*
		(2)	400.5	12821	110557			(6)	351	6801	49298
		(3)	370.5	12921	113851			(7)	302	7142	51341
		(4)	392.3	12939	111791			(8)	359.8	7037	50246
5	(5)	(1)	332.5	13793	112077	10	(10)	(5)	352.5	5883	45737
		(2)	335.5	13156	116086			(6)	328	6848	51650
		(3)	265.5	13516	120655			(7)	317.5	6798	54327
		(4)	309.8	14017	118937			(8)	355.8	6663	52553

Note: The symbol "*" denotes the optimal solutions.

This suggests that managers should consider both immediate gains and future stability when optimizing capacity patterns.

2) Impacts of period parameters

First, we consider the impact of the MWTTs by varying W_1 from 2 to 4. Table IX summarizes the solution performance across different cost components for various values of W_1. With increasing W_1, we observe that the patient preferred revisit day violation cost C_{Rt} and total short-term cost of the first five periods (decision periods) TC_5 have similar trends with the lowest value at W_1=3, both decreasing first and then tends to increase. Contrast to C_{Rt} and TC_5, idle cost C_a show an upward trend. Intuitively, these tendencies are due to the fact that more capacities can be used for service when the time window is wider with larger W_1 values. The results indicate that initially, C_{Rt} plays a dominant role in reducing the TC_5, leading to a decrease in TC_5. Subsequently, the combined increasing effects of C_a and C_{Rt} contribute to a rise in the TC_5.

Insight 3.1. *The sensitivity analysis of the W_1 provides a guiding tool for hospital practitioners in selecting the right W_1 for their settings. By adjusting W_1, it is possible to effectively balance cost, efficiency, and patient demand, thereby optimizing the overall system performance.*

Next, we analyze the impact of the allowable span of the revisit intervals by varying the span from 1 to 3. Table X summarizes cost components under different combinations of $(\underline{a}^o,\ \overline{a}^o)$; $(\underline{a}^e,\ \overline{a}^e)$ (labeled as "Com"). C_a and C_{Rt} show significant variation, suggesting they have the largest impact on TC_5. The rejection or transfer cost C_{Tf}/C_{Tr} and overtime cost C_o remain unchanged, suggesting these factors might not influence TC_5 in these scenarios. Com 1 has the highest TC_5, driven by a notably high C_a. Com 4 has the lowest TC_5, indicating a potentially more cost-efficient setup. Table X illustrates that the TC_5 follows the same trends as C_a and C_{Rt}: it first decreases and then increases as the allowable spans for offline and online revisits expand, as seen by comparing Coms 1, 4, and 6, or Coms 2, 4, and 5. Comparing Coms 1 and 2, Coms 3 and 4, or Coms 5 and 6, where each pair has the same allowable span but a longer revisit interval in the latter, reveals mixed results. In some cases, the latter scenario yields better system performance, while in others, the former performs better.

TABLE IX: Sensitivity analysis with different W_1

W_1	C_o	C_a	C_{Tf}	C_{Tr}	C_{Rt}	TC_5
2	0	50	0	4	351	405
3	0	110	0	4	125	239
4	0	220	0	4	204	428

Insight 3.2. *This suggests that as a patient's condition improves, practitioners can optimize system performance by adjusting either the allowable span of the revisit interval or the revisit interval itself.*

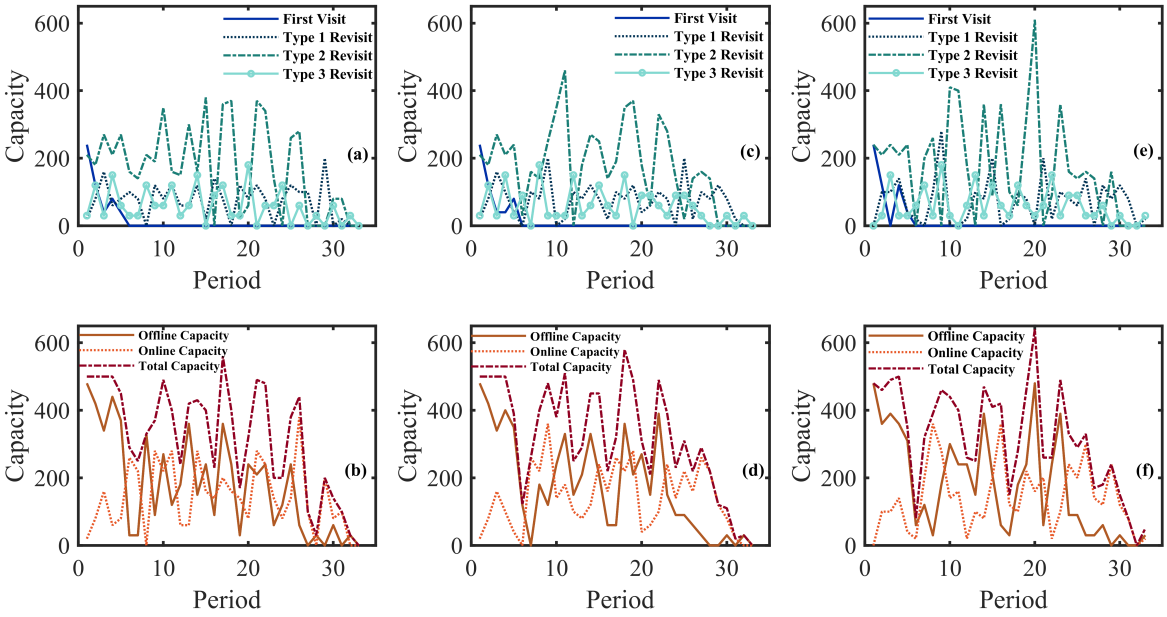


Fig. 9: Capacity allocation with different W_1. (a)(b): W_1=2. (c)(d): W_1=3. (e)(f): W_1=4

TABLE X: Sensitivity analysis results on revisit interval

Com	$(\underline{a}^o,\ \overline{a}^o)$; $(\underline{a}^e,\ \overline{a}^e)$	C_o	C_a	C_{Tf}	C_{Tr}	C_{Rt}	TC_5
1	(2, 3);(4, 5)	0	720	0	4	236	960
2	(3, 4);(5, 6)	0	80	0	4	169	253
3	(1, 3);(3, 5)	0	370	0	4	182	556
4	(2, 4);(4, 6)	0	110	0	4	125	239
5	(1, 4);(3, 6)	0	190	0	4	216	410
6	(2, 5);(4, 7)	0	290	0	4	418	712

To analyze the solutions, we consider the capacity allocation for each patient type, as well as the online, offline, and total capacities under different values of W_1, as illustrated in Fig. 9. The figure reveals two key observations. First, there is significant variation across periods: both patient-type-specific capacities and online/offline allocations fluctuate significantly over time. To address this, we suggest a dynamic adjustment strategy based on the capacity allocation scheme. Second, the effect of W_1. Comparing figures (a)-(b), (c)-(d), and (e)-(f) highlights varying W_1 directly and significantly influences both patient type distribution and the split between online and offline capacities. Figures (a), (c), and (e) illustrate the capacity allocation for different types of patients (first visits, Type 1, Type 2, and Type 3 revisits). It can be observed that the dominant trend is Type 2 revisits, which consistently demand the most capacity across different periods, while first visits and Type 3 revisits display more sporadic allocation patterns. As shown in Figures (b), (d), and (f), in some cases, the allocation of online and offline capacity tends to be balanced, while in others, it may lean towards offline or online services. The volatility of total capacity varies with different values of W_1.

Insight 3.3. *This suggests that dynamically adjusting total capacity allocation across periods, optimally distributing capacity between online and offline services, and optimizing capacity allocation for different patient types can help minimize overall system operational costs.*

3) Impacts of cost parameters

In this subsection, we conduct sensitivity analysis on unit overtime cost, idle cost, revisit rejection or diverting cost, and patient preference violation cost, with each unit cost varying between 0.2 and 3. As shown in Fig. 10, we find that overtime cost and first visit/revisit rejection or diverting cost are largely insensitive to the changes in any of the unit costs, as they consistently remain zero, while revisit rejection or diverting cost stays very low (ranging from 0.4 to 6) with almost no variation. Idle cost dominates the overall trend, as the trend of total short-term cost always aligns with that of idle cost, which tends to decrease as the unit patient preference violation cost increases. We further observe that when the unit overtime and idle costs are 2.5, the unit revisit rejection or diverting cost equals 1.6 or 2.5, and the unit patient preference violation cost takes values of 0.6 or 1.4, two to four cost components exhibit sharp changes and inconsistent trends around these points. Moreover, the trends of total long-term cost and total short-term cost are sometimes aligned but not always. For instance, as the unit idle cost increases, the total long-term cost rises almost monotonically in a linear trend, whereas the total short-term cost shows an overall increasing trend but with localized decreases. Notably, when the unit revisit rejection or diverting cost is set to 1.6, and under many values of patient preference violation cost, the total long-term cost and total short-term cost move in completely opposite directions. This may be due to the fact that small changes in unit cost coefficients exert limited influence in the short term but can accumulate and be amplified in the long term, leading to divergent trends. These findings reveal that the proposed model involves discrete decisions and exhibits "critical points" or "threshold effects". The impact of unit cost coefficients is not always transmitted linearly, but may instead trigger capacity allocation changes or

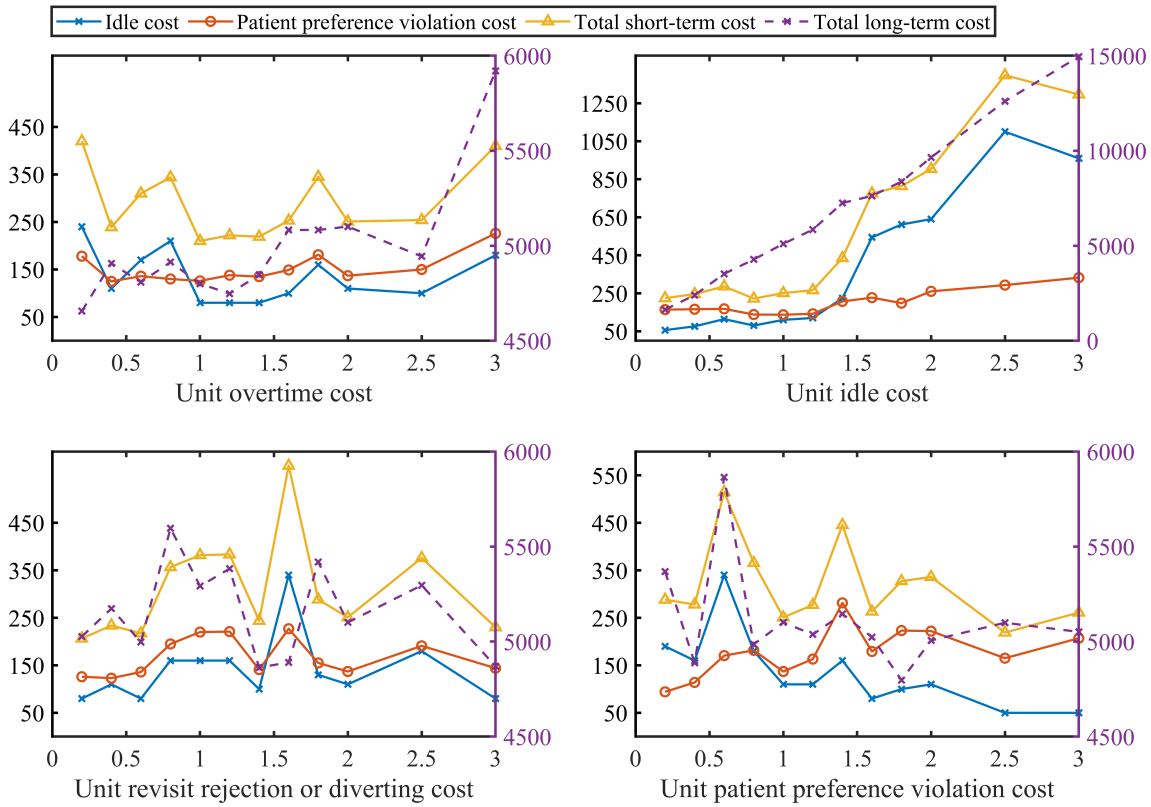


Fig. 10: Changes in idle cost, patient preference cost and total short-term cost (left axis) and total long-term cost (right axis) across unit costs. A dual y-axis is adopted to display both cost components within a single figure, as their magnitudes differ by two orders of magnitude.

strategy switching in certain intervals or points, causing non-monotonicity. The short-term optimum does not necessarily align with the long-term optimum. Therefore, policy-making should consider both immediate and long-term effects. In particular, at the intervals or points where trend inconsistency or "jumps" occur, the system is highly sensitive to parameter changes, and decisions should be made with caution.

VI. Conclusions

This paper investigates a first visit and multitype multiple revisits capacity allocation problem considering stochastic demands in online and offline outpatient clinics setting. Due to the stochastic demands for first visits and multitype multiple revisits and the multiple revisits transfers between online and offline services, it is challenging for hospital practitioners to dynamically match patients with service capacities. We develop a dynamic multistage SMIP model to minimize the total expected costs for violating patient preferred revisit day, rejecting or transferring patients, capacity overtime, and capacity idle time by considering the access way and time interval of revisits. These revisit interval constraints make it intractable directly. Therefore, we reformulate the model and propose the DB-ACARP algorithm to solve the problem. To demonstrate the performance of our proposed algorithm, we consider a commercial solver as benchmark. Numerical results indicate that the running time and solution quality of the DB-ACARP algorithm is improved a lot compared with the commercial solver both in short-term and long-term costs. In addition, numerical studies are conducted to analyze how certain important factors affect the decision model. We have also presented some managerial insights to provide guidelines to practitioners in employing the decision model and solution method.

Several future research directions could extend our model for integrated online and offline healthcare operation management. The first direction is to incorporate patient behavior, such as no shows and cancellations that could significantly affect capacity allocation decisions and resource utilization. The second direction is to provide the optimal decisions on the appointment day and time of online and offline revisits and first visits by studying the joint capacity allocation and appointment scheduling problem. It could consider modeling the revisit mode (online or offline) as decision variables. Additional constraints would ensure only eligible patient types are assigned to online visits. The third direction is to develop a robust or distributional robust optimization model to model uncertain parameters as close to reality as possible.

References

[1] T. S. Bergmo, P. E. Kummervold, D. Gammon, and L. B. Dahl, "Electronic patient–provider communication: Will it offset office visits and telephone consultations in primary care?" *Int. J. Med. Inform.*, vol. 74, no. 9, pp. 705–710, Sep. 2005.

[2] P. Whitten, L. Buis, and B. Love, "Physician-patient e-visit programs: Implementation and appropriateness," *Dis. Manag. Health Outcomes*, vol. 15, pp. 207-214, Dec. 2007.

[3] D. Huang, R. Sangthong, E. McNeil, V. Chongsuvivatwong, W. Zheng, and X. Yang, "Effects of a phone call intervention to promote adherence to antiretroviral therapy and quality of life of HIV/AIDS patients in Baoshan, China: A randomized controlled trial," *AIDS Res. Treat.*, vol. 1, pp. 1–9, 2013.

[4] H. Bavafa, L. M. Hitt, and C. Terwiesch, "The impact of e-visits on visit frequencies and patient health: Evidence from primary care," *Manage. Sci.*, vol. 64, no. 12, pp. 5461–5480, Dec. 2018.

[5] H. Bavafa, S. Savin, and C. Terwiesch, "Managing office revisit intervals and patient panel sizes in primary care," *arXiv preprint arXiv*, 2363685, 2013.

[6] S. Rath, K. Rajaram, and A. Mahajan, "Integrated anesthesiologist and room scheduling for surgeries: methodology and application," *Oper. Res.*, vol.65, no. 6, pp. 1460-1478, Nov.–Dec. 2017.

[7] J. Patrick, M.L. Puterman, and M. Queyranne, "Dynamic multipriority patient scheduling for a diagnostic resource," *Oper. Res.*, vol. 56, no. 6, pp. 1507-1525, Nov.–Dec. 2008.

[8] N. Liu, V.A.Truong, and B.R.Anderson, "Integrated scheduling and capacity planning with considerations for patients' length-of-stays," *Prod. Oper. Manag.*, vol. 28, no. 7, pp. 1735-1756, Sep. 2019.

[9] V.A. Truong, "Optimal advance scheduling," *Manage. Sci.*, vol. 61, no. 7, pp. 1584-1597, Jul. 2015.

[10] L. Zhou, N. Geng, Z. Jiang, and X. Wang, "Dynamic multi-type patient advance scheduling for a diagnostic facility considering heterogeneous waiting time targets and equity," *IISE Trans.*, vol. 54, no. 6, pp. 521-536, Jun. 2022.

[11] N. Geng, X. Xie, and Z. Jiang, "Implementation strategies of a contract-based MRI examination reservation process for stroke patients," *Eur. J. Oper. Res.*, vol. 231, no. 2, pp. 371-380, Nov. 2013.

[12] G. Dobson, S. Hasija, and E.J. Pinker, "Reserving capacity for urgent patients in primary care," *Prod. Oper. Manag.*, vol. 20, no. 3, pp. 456-473, May–Jun. 2011.

[13] K. Sun, M. Sun, D. Agrawal, R. Dravenstott, F. Rosinia, and A. Roy, "Equitable anesthesiologist scheduling under demand uncertainty using multiobjective programming," *Prod. Oper. Manag.*, vol. 32, no. 11, pp. 3699-3716, Nov. 2023.

[14] D. Khorasanian, J. Patrick, and A. Sauré, "Dynamic home care routing and scheduling with uncertain number of visits per referral," *Transp. Sci.*, vol. 58, no. 4, pp. 841-859, Oct. 2024.

[15] J. Dai, N. Geng, and X. Xie, "Dynamic advance scheduling of outpatient appointments in a moving booking window," *Eur. J. Oper. Res.*, vol. 292, no. 2, pp. 622-632, Feb. 2021.

[16] D. Astaraky and J. Patrick, "A simulation based approximate dynamic programming approach to multi-class, multi-resource surgical scheduling," *Eur. J. Oper. Res.*, vol. 245, no. 1, pp. 309-319, Jan. 2015.

[17] A. Bansal, B. Berg, and Y.-L. Huang, "A distributionally robust optimization approach for coordinating clinical and surgical appointments," *IISE Trans.*, vol. 53, no. 12, pp. 1311-1323, Dec. 2021.

[18] T. A. Silva and M. C. de Souza, "Surgical scheduling under uncertainty by approximate dynamic programming," *Omega-Int. J. Manage. Sci.*, vol. 95, p. 102066, Dec. 2020.

[19] M. Biggs and G. Perakis, "Dynamic routing with tree based value function approximations," 2020. [Online]. Available: https://ssrn.com/abstract=3680162

[20] A. Kumar, A.M. Costa, M. Fackrell, and P.G. Taylor, "A sequential stochastic mixed integer programming model for tactical master surgery scheduling," *Eur. J. Oper. Res.*, vol. 270, no. 2, pp. 734-746, May 2018.

[21] K.S. Shehadeh and R. Padman, "A distributionally robust optimization approach for stochastic elective surgery scheduling with limited intensive care unit capacity," *Eur. J. Oper. Res.*, vol. 290, no. 3, pp. 901-913, Feb. 2021.

[22] J. Zhang, M. Dridi, and A. El Moudni, "Column-generation-based heuristic approaches to stochastic surgery scheduling with downstream capacity constraints," *Int. J. Prod. Econ.*, vol. 229, pp. 107764, Jan. 2020.

[23] A. Heching, J. N. Hooker, and R. Kimura, "A logic-Based benders approach to home healthcare delivery," *Transp. Sci.*, vol. 53, no. 2, pp. 510-522, Apr. 2019.

[24] R. Bargetto, T. Garaix, and X. Xie, "A branch-and-price-and-cut algorithm for operating room scheduling under human resource constraints," *Comput. Oper. Res.*, p. 106136, 2023.

[25] S. Kifah and S. Abdullah, "An adaptive non-linear great deluge algorithm for the patient-admission problem," *INFORMS J. Comput.*, vol. 295, pp. 573-585, Jul. 2015.

[26] L. Zhou, N. Geng, Z. Jiang, and X. Wang, "Public hospital inpatient room allocation and patient scheduling considering equity," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 3, pp. 1124-1139, Jul. 2019.

[27] Y. Lu, Z. Jiang, N. Geng, S. Jiang, and X. Xie, "Appointment window scheduling with wait-dependent abandonment for elective inpatient admission," *Int. J. Prod. Res.*, vol. 60, no. 19, pp. 5977-5993, Oct. 2022.

[28] N. Geng and X. Xie, "Managing advance admission requests for obstetric care," *INFORMS J. Comput.*, vol. 34, no. 2, pp. 1224-1239, Apr. 2022.

[29] M. Salemi Parizi and A. Ghate, Multi-class, "multi-resource advance scheduling with no-shows, cancellations and overbooking," *Comput. Oper. Res.*, vol. 67, pp. 90-101, Mar. 2016.

[30] H. Mahmoudzadeh, A. Mirahmadi Shalamzari, and H. Abouee-Mehrizi, "Robust multi-class multi-period patient scheduling with wait time targets," *OPER. RES. HEALTH CARE*, vol. 25, p. 100254, Jun. 2020.

[31] A. Sauré, M.A. Begen, and J. Patrick, "Dynamic multi-priority, multi-class patient scheduling with stochastic service times," *Eur. J. Oper. Res.*, vol. 280, no. 1, pp. 254-265, Jan. 2020.

[32] T. B. T. Nguyen, A. I. Sivakumar, and S. C. Graves, "A network flow approach for tactical resource planning in outpatient clinics," *Health Care Manag. Sci.*, vol. 18, no. 2, pp. 124-136, May 2015.

[33] T. B. T. Nguyen, A. I. Sivakumar, and S.C. Graves, "Capacity planning with demand uncertainty for outpatient clinics," *Eur. J. Oper. Res.*, vol. 267, no. 1, pp. 338-348, Jan. 2018.

[34] X. Yu and A. Bayram, "Managing capacity for virtual and office appointments in chronic care," *Health Care Manag. Sci.*, vol. 24, no. 4, pp. 742-767, Oct. 2021.

[35] A. Sauré, J. Patrick, S. Tyldesley, and M.L. Puterman, "Dynamic multi-appointment patient scheduling for radiation therapy," *Eur. J. Oper. Res.*, vol. 223, no. 2, pp. 573-584, Jul. 2012.

[36] S. Yu, V.G. Kulkarni, and V. Deshpande, "Appointment scheduling for a health care facility with series patients," *Prod. Oper. Manag.*, vol. 29, no. 2, pp. 388-409, Feb. 2020.

[37] A. Sauré, J. Patrick, and M.L. Puterman, "Simulation-based approximate policy iteration with generalized logistic functions," *INFORMS J. Comput.*, vol. 27, no. 3, pp. 579-595, Aug. 2015.

[38] H.-J. Schütz and R. Kolisch, "Capacity allocation for demand of different customer-product-combinations with cancellations, no-shows, and overbooking when there is a sequential delivery of service," *Ann. Oper. Res.*, vol. 206, pp. 401-423, 2013.

[39] Y. Gocgun and M.L. Puterman, "Dynamic scheduling with due dates and time windows: an application to chemotherapy patient appointment booking," *Health Care Manag. Sci.*, vol. 17, no. 1, pp. 60-76, Feb. 2013.

[40] H. Zhang, J. Zhao, H. Leung, and W. Wang, "Multi-stage dynamic optimization method for long-term planning of the concentrate ingredient in copper industry," *Inf. Sci.*, vol. 605, pp. 333-350, Mar. 2022.

[41] W. Liu and J. Zhu, "A multistage decision-making method with quantum-guided expert state transition based on normal cloud models," *Inf. Sci.*, vol. 615, pp. 700-730, May 2022.

[42] P. Guo, Z. Chen, Y. Yang and R. Miao, "A multistage simulation-optimization-integrated methodology framework for user-oriented electric vehicle carsharing reallocation under dynamic price subsidy," *Energy*, vol. 290, p. 130207, Jan. 2024.

[43] Z. Zhang, X. Xie, and N. Geng, "Dynamic surgery assignment of multiple operating rooms with planned surgeon arrival times," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 3, pp. 680-691, Jul. 2013.

[44] L.F. Escudero, M.A. Garín, and A. Unzueta, "Cluster Lagrangean decomposition in multistage stochastic optimization," *Comput. Oper. Res.*, vol. 67, pp. 48–62, Mar. 2016.

[45] Y. Liu, R. Sioshansi, and A.J. Conejo, "Multistage stochastic investment planning with multiscale representation of uncertainties and decisions," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 781–791, Jan. 2018.

[46] S. Gul, B. T. Denton, and J. W. Fowler, "A progressive hedging approach for surgery planning under uncertainty," *INFORMS J. Comput.*, vol. 27, no. 4, pp. 755–772, Nov. 2015.

[47] T. G. Crainic, F. Xiaorui, M. Gendreau, W. Rei, and S. W. Wallace, "Progressive hedging-based metaheuristics for stochastic network design," *Networks*, vol. 58, no. 2, pp. 114–124, Mar. 2011.

[48] J. P. Watson and D. L. Woodruff, "Progressive hedging innovations for a class of stochastic mixed-integer resource allocation problems," *Comput. Manag. Sci.*, vol. 8, no. 4, pp. 355–370, Dec. 2011.

[49] J. Nossack, "Therapy scheduling and therapy planning at hospitals," *Omega-Int. J. Manage. Sci.*, vol. 109, p. 102594, Feb. 2022.

[50] B. Çelik, S. Gul, and M. Çelik, "A stochastic programming approach to surgery scheduling under parallel processing principle," *Omega-Int. J. Manage. Sci.*, vol. 115, p. 102799, Jan. 2023.

[51] K. Shao, W. Fan, S. Lan, M. Kong, and S. Yang, "A column generation-based heuristic for brachytherapy patient scheduling with multiple treatment sessions considering radioactive source decay and time constraints," *Omega-Int. J. Manage. Sci.*, p. 102853, Jan. 2023.

[52] M. Zhao, Y. Wang, X. Zhang, and C. Xu, "Online doctor-patient dynamic stable matching model based on regret theory under incomplete information," *Socio-econ. Plan. Sci.*, vol. 87, p. 101615, Jan. 2023.

[53] Y. Ding, D. Gupta, and X. Tang, "Early reservation for follow-up appointments in a slotted-service queue," *Oper. Res.*, vol. 71, no. 3, pp. 917-938, May–Jun. 2023.

[54] X. Shen, S. C. Du, Y. N. Sun, P. Z. Sun, R. Law, and E. Q. Wu, "Advance scheduling for chronic care under online or offline revisit uncertainty," *IEEE Trans. Autom. Sci. Eng.*, vol. 21, no. 4, pp. 5297-5310, Oct. 2024.

[55] X. Shen, N. Li, and X. Xie, "Multiserver time window allowance schedules for virtual visits with uncertain time-dependent no-shows and service times," *Adv. Eng. Inform.*, vol. 59, p. 102252, Jan. 2024.

[56] C. W. Yancy, M. Jessup, B. Bozkurt, J. Butler, D. E. Casey Jr, M. M. Colvin, ... , and C. Westlake, "2017 ACC/AHA/HFSA Focused Update of the 2013 ACCF/AHA Guideline for the Management of Heart Failure: A Report of the American College of Cardiology/American Heart Association Task Force on Clinical Practice Guidelines and the Heart Failure Society of America," *J. Am. Coll. Cardiol.*, vol. 70, no. 6, pp. 776–803, Aug. 2017.

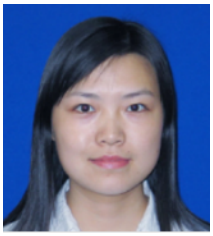
[57] Chinese Diabetes Society, "Guidelines for the prevention and treatment of type 2 diabetes in China (2020 edition)," *Chin. J. Diabetes Mellitus*, vol. 13, no. 4, pp. 315–409, Aug. 2021.

[58] National Health Service (NHS), "Outpatient Appointment Policy," NHS England, UK, 2019. [Online]. Available: https://www.england.nhs.uk.

[59] J. Gao, J. Zhu, J. Wang, K. Li, L. Zhen, and E. Demeulemeester, "The two-visit team orienteering problem considering time-interval-dependent profits and service consistency," Computers & Operations Research, vol. 188, Art. no. 107370, Apr. 2026.

[60] J. Li, J. Zhu, G. Peng, J. Wang, L. Zhen, and E. Demeulemeester, "Branch-price-and-cut algorithms for the team orienteering problem with interval-varying profits," European Journal of Operational Research, vol. 319, no. 3, pp. 793–807, Dec. 2024.

Xiaoxiao Shen (Member, IEEE) received the B.S. degree from the College of Information Management, Nanjing Agricultural University, Nanjing, China in 2019. In 2025, she received dual Ph.D. degrees from the Department of Industrial Engineering and Management at Shanghai Jiao Tong University, Shanghai, China, and the Department of Mechanical Engineering at Politecnico di Milano, Milan, Italy. She is currently a lecturer with the Department of Industrial Engineering, School of Mechanical and Electronic Engineering, Wuhan University of Technology. Her research interests include analysis, modeling, and decision-making optimization of operations management problems in manufacturing and service systems.



Jun Lv received the Ph.D. degree in business administration from Shanghai University of finance and economics, Shanghai, China, in 2008. She is currently an Associate Professor with faculty of economics and management, East China Normal University. Her current research interests focus on sustainable operation, production scheduling and operations research.

Shi-Chang Du (Member, IEEE) received the B.S. and M.S.E. degrees in mechanical engineering from the Hefei University of Technology, Hefei, China, in 2000 and 2003, respectively, and the Ph.D. degree in industrial engineering and management from Shanghai Jiao Tong University, Shanghai, China, in 2008. He was with the University of Michigan, Ann Arbor, MI, USA, from 2006 to 2007, as a Visiting Scholar. He is currently a Professor with the Department of Industrial Engineering and Management, Shanghai Jiao Tong University. He has authored/coauthored 80 articles in journals. His current research interests focus on quality control and smart manufacturing methods. He serves for several journals an Area Editor including Computers and Industrial Engineering, Flexible Services and Manufacturing Journal, Journal of Intelligent Manufacturing, and Computers in Industry, etc.

Andrea Matta (Member, IEEE) is Full Professor of Manufacturing and Production Systems at Department of Mechanical Engineering of Politecnico di Milano. He graduated in Industrial Engineering at Politecnico di Milano where he develops his teaching and research activities since 1998. He was a Distinguished Professor at the School of Mechanical Engineering, Shanghai Jiao Tong University (2014–2016), and a Guest Professor from 2017 to 2019. He has been visiting professor at Ecole Centrale Paris (France), University of California at Berkeley (USA), and Tongji University (China). His research area includes analysis, design and management of manufacturing and health care systems. He has published 170+ scientific papers on international and national journals/conference proceedings. He is the Editor in Chief of Flexible Services and Manufacturing Journal since 2017, and the Chair of the IEEE RAS Technical Committee on Sustainable Production Automation. He was awarded with the Shanghai One Thousand Talent and Eastern Scholar in 2013.