A Branch and Price Algorithm for Scheduling in Surgery Pre-admission Testing Clinics

Mohammad Al Syouf* Ankit Bansal† Saligrama Agnihothri‡

Abstract

A Surgery Pre-Admission Testing (PAT) clinic is a hospital unit designed to serve pre-operative patients by gathering critical patient information and performing procedure-specific tests to prepare them for surgery. Patients may require multiple tests, each conducted by a specialized nurse. A patient must be assigned to a room before starting any test and must stay there until all tests are completed, with only one test active at a time. This can lead to inefficiencies, such as patients waiting for nurses or nurses remaining idle while another test is ongoing for a patient. To mitigate this, a scheduling approach is required to synchronize the schedules of patients, rooms, and nurses. In this paper, we introduce a novel path-based formulation for the PAT scheduling problem (PATSP) to determine the schedules of patients, rooms, and nurses while incorporating these synchronization constraints. We propose a Branch and Price algorithm to solve this model and introduce various techniques to enhance its computational efficiency. Our computational experiments show that the proposed approach significantly outperforms an existing method in the literature. Additionally, we explore the impact of integrating synchronization constraints into the scheduling framework.

Keywords: OR in health services, Surgery pre-admission testing, Patient appointment scheduling, Resource allocation, Branch and Price algorithm

1 Introduction

Before surgery, patients must undergo testing and preparation to ensure that they can successfully complete the procedure. Traditionally, patients visit different hospital departments depending on their doctor's test orders (Cordier and Riane 2013). As a result, patients must schedule and attend multiple appointments at various test centers located throughout the hospital or even outside of the

*Department of Systems Science and Industrial Engineering, State University of New York-Binghamton, Binghamton, NY, United States

†Department of Systems Science and Industrial Engineering, State University of New York-Binghamton, Binghamton, NY, United States, abansal@binghamton.edu (*Corresponding author*)

‡School of Management, State University of New York-Binghamton, Binghamton, NY, United States

hospital. This fragmented approach increases the inconvenience for the patient and the time required to complete necessary tests. If tests are not completed on time and results are not available the day before surgery, the procedure may need to be postponed.

On the contrary, Surgery Pre-Admission Testing (PAT) clinics consolidate all required tests into a single location and appointment, reducing the number of canceled surgeries and pre-surgery waiting times. Patients visiting a PAT clinic may need multiple tests, each performed by a nurse with specialized training. After being taken to an exam room, patients remain there until all necessary tests are completed, which can be performed in any order. As nurses move between exam rooms to perform the required tests, the patient may have to wait for a nurse or a nurse may have to wait if the patient is occupied with another test. The duration of tests depend on the specific test and remains constant regardless of the patient.

This study is motivated by a case study conducted at a PAT clinic within a hospital system that conducts approximately 17,000 surgical procedures annually. Details of this case study are provided in Appendix A of Agnihothri et al. (2024). During a typical PAT visit, patients interact with two to five specialized nurses, depending on the surgeon's recommendations and the complexity of the upcoming procedure. Each visit begins with a consultation with a pharmacist, followed by a sequence of services that may include laboratory tests, X-rays, EKGs, and a medical history review with a registered nurse. Patients also receive preoperative instructions and, when required, consult with a nurse practitioner regarding anesthesia. The primary goal of the case study was to identify strategies to enhance clinic capacity and reduce patient wait times.

McCarty et al. (2012) present a case study of the PAT clinic at Massachusetts General Hospital (MGH), which conducts approximately 34,000 surgical procedures annually. The clinic is equipped with 12 exam rooms, and each patient is required to meet with a laboratory technician, an anesthesiologist, and a nurse—all within the same room—prior to completing their visit. The clinical staff consists of five lab technicians, eight anesthesiologists, and five nurses. The study examines the clinic's operations prior to the implementation of a process improvement initiative and provides a comprehensive view of the PAT experience from both patient and provider perspectives. It emphasizes the critical role of effective PAT operations, citing their substantial downstream influence on operating room efficiency and the broader perioperative care system at MGH.

This paper addresses the PAT scheduling problem (PATSP), which involves assigning patients to exam rooms, allocating nurses to individual patients, and determining the start time of each test for every patient, with the objective of minimizing the total number of rooms and nurses utilized. The

study focuses on addressing PATSP when i.) the required tests of each patient are known at the time of scheduling; ii.) each nurse is trained to perform only one type of test and is not cross-trained; iii.) a patient can be attended by only one nurse at a time and vice versa; and iv.) once a room is assigned to a patient, it remains in use until all the tests of the patient are completed.

Any optimal solution for PATSP must synchronize room and nurse schedules. Without this, a patient may wait for a nurse after being assigned a room, or a nurse may wait while the patient undergoes another test, increasing the number of rooms or nurses required to serve all patients. Agnihothri et al. (2024) study a similar problem that seeks to minimize both the makespan and the total idle time experienced by patients. They propose a Mixed Integer Linear Programming (MILP) formulation to model the problem; however, solving their model to optimality for realistic-size instances requires high computational effort. In this paper, we introduce a novel path-based formulation for PATSP. Unlike traditional path-based approaches in patient scheduling (Beaudry et al. 2010, Hanne et al. 2009, Schmid and Doerner 2014, Nemati et al. 2016), which model patient paths through nodes representing hospitals or intra-hospital locations such as rooms, our proposed model defines nodes as patients and constructs paths for rooms and nurses over these patient nodes. This approach has two key advantageous: i.) it results in a Dantzig-Wolfe reformulation of PATSP, enabling the use of the Branch and Price (B&P) algorithm (Barnhart et al. 1998) to efficiently solve realistic-size instances of the problem. ii.) unlike the MILP formulation in Agnihothri et al. (2024), this modeling approach eliminates the need for Big-M to represent sequencing constraints. Big-M constraints are known to weaken the linear programming relaxation of a MILP, which negatively impacts the efficiency of the Branch and Bound Algorithm.

B&P algorithms have been widely used to address path-based Dantzig-Wolfe formulations both with (Fink et al. 2019, Li et al. 2020, Jungwirth et al. 2022) and without (Costa et al. 2019) synchronization constraints. In this work, we develop a B&P framework to solve the path-based Dantzig-Wolfe formulation of PATSP, which includes synchronization constraints. Our proposed B&P algorithm differs from existing implementations by introducing several enhancements to improve computational performance. Specifically, we incorporate: (i) facet-defining valid inequalities for the convex hull of the pricing subproblem, (ii) a tailored upper-bounding heuristic to generate feasible solutions at each node of the B&P tree, (iii) a tailored branching rule, and (iv) a collection of lower bounds on the optimal objective value of PATSP.

We conduct extensive computational experiments to evaluate the effectiveness of the proposed approach. Using realistic problem instances from Agnihothri et al. (2024), we demonstrate that for

the objective function considered here, our modeling and algorithmic approach outperforms the MILP formulation presented in Agnihothri et al. (2024). We also examine how synchronizing room and nurse schedules affects the number of rooms required and the nurse staffing levels needed to serve a given set of patient. The remainder of the paper is organized as follows. Section 2 provides a brief review of the literature. The new formulation for `PATSP` is given in Section 3. Section 4 outlines the details of the `B&P` algorithm to solve the formulation in Section 3. Implementation details of the proposed algorithm are explained in Section 5 while Section 6 outlines the computational experiments. Section 7 concludes the paper.

2 Literature Review

Figure 1 provides an overview of the relevant literature streams discussed in this review and identifies the specific area where this paper is situated within those streams. We start with patient appointments scheduling (assigning a specific time to a patient to start receiving service) where the method used depends on the service setup. There are three different service configurations: i). *Single-station setup*: Here, a patient visits only one server (for example, an exam room or test station). Appointments are scheduled sequentially, and there is only one resource to manage. The goal is to reduce patient waiting time while keeping the server busy. Comprehensive reviews of single-station scheduling models are provided by Cayirli and Veral (2003), Gupta and Denton (2008), Gupta and Wang (2012), Ahmadi-Javid et al. (2017). ii). *Multi-station setup*: In a multi-station setup, such as a clinic with different specialized areas like imaging and lab tests, patients move from one station to another. The scheduling system must coordinate several resources at the same time. For example, if a patient needs both imaging and lab tests, the system should ensure that both stations are available when needed. Well-coordinated scheduling can reduce overall wait times, but delays in one station can affect others if not managed carefully. Models for multi-station scheduling are discussed in Wang et al. (2019) and Apergi et al. (2020) while Marynissen and Demeulemeester (2019) provide a review in hospital settings. iii). *Mobile server (bedside care) setup*: Note that in the previous two scenarios, the server stays in one place while the patient visits multiple servers to receive service. `PATSP` belongs to a third category called *mobile server setup* in which the patient stays in the same place and receives multiple services sequentially by mobile servers. This approach can improve patient comfort and reduce the delays associated with patient moving between servers. Zhang et al. (2019) studied an Integrated Practice Unit (IPU), where a co-located multidisciplinary team provides treatment for a

single patient visit. When service times are deterministic, they propose a MILP model which differs from the problem we studied here except that they also consider a mobile server model. Morrice et al. (2020) use simulation to study a scheduling problem in an IPU. A comprehensive review of papers in this category is provided by Agnihothri et al. (2024). The same paper proposes a MILP formulation for a similar `PATSP` problem with a different objective function. However, this formulation contains a vast number of variables and constraints, including Big-M constraints, which pose significant challenges for commercial MILP solvers when solving realistic-size instances. Therefore, this paper introduces a path-based formulation for `PATSP` that eliminates Big-M constraints and leads to a decomposition framework that can be leveraged to solve realistic-size instances.

Research on path-based approaches in healthcare can be broadly classified into two categories: studies that explore routing between healthcare facilities and those that focus on intra-hospital routing. The former primarily addresses patient transportation to hospitals (Agius et al. 2022, Nemati et al. 2016), pharmaceutical logistics (Xu et al. 2025, Kramer et al. 2019), and home healthcare services (Di Mascolo et al. 2021, Cissé et al. 2017). In contrast, the literature on routing within a healthcare facility is relatively sparse. Notable studies include Beaudry et al. (2010), Hanne et al. (2009), Schmid and Doerner (2014), which investigate patient movement within a hospital, and Gartner et al. (2018), Jungwirth et al. (2022), which examine the paths of physical therapists within a healthcare facility. This paper falls within the category of intra-hospital routing. However, unlike existing studies in this area—which typically model patient flows through nodes representing physical locations within a hospital (e.g., rooms)—the approach proposed here defines nodes as patients and constructs routing paths for rooms and nurses over these patient nodes.

Path-based approaches are also widely used to model complex synchronization constraints among the entities that flow through the paths (Soares et al. 2024, Drexl 2012). Drexl (2012) classify such synchronization constraints into five types: *Task, Operational, Movement, Load* and *Resource*. Jungwirth et al. (2022) propose a path-based approach for scheduling physical therapists across multiple hospital locations while considering *Resource* and *Operational* synchronization constraints. Their *Resource* synchronization constraints account for capacity limitations at each location, whereas their *Operational* synchronization constraints enforce precedence relationships between multiple treatments of a patient. From a modeling perspective, the approach introduced in this paper closely aligns with the one proposed in Hashemi Doulabi et al. (2020). They develop a path-based model for assigning Operating Rooms (ORs) to patients, where OR and surgeon paths through the patients are determined while ensuring that both the OR and the surgeon are available simultaneously for a surgery

to take place. In this paper, we develop a path-based model for determining room and nurse paths through the patients such that they satisfy three key *Operational* synchronization constraints: (i) a patient must be assigned a room before any test can be conducted, (ii) patients stay in their assigned room until all their tests are complete, and (iii) multiple tests for the same patient cannot overlap in time. We solve the proposed model using a `B&P` algorithm.

`B&P` algorithm (Barnhart et al. 1998) has been widely used to solve various computationally intractable problems (Lalonde et al. 2022, Costa et al. 2019, Oliveira and Pessoa 2020, Zhang et al. 2020, Deleplanque et al. 2020, Dell'Amico et al. 2008). In the healthcare domain, it has been applied to schedule medical providers (Legrain and Omer 2024, Beliën and Demeulemeester 2008, Akbarzadeh and Maenhout 2025, Jungwirth et al. 2022), allocate costly resources such as OR time (Bargetto et al. 2023, Hashemi Doulabi et al. 2016), and schedule patient appointments (Reihaneh et al. 2023). Additionally, it has also been widely implemented to solve path-based models with synchronization constraints (Fink et al. 2019, Li et al. 2020, Jungwirth et al. 2022). Thus, we present a `B&P` algorithm to solve the proposed path-based model for `PATSP` and introduce strategies to improve its computational efficiency. Our computational experiments show that the proposed `B&P` algorithm outperforms the Branch and Cut algorithm implemented to solve the MILP formulation of Agnihothri et al. (2024) for `PATSP` with a different objective function.

3 Problem Description and Path-Based Formulation

Let $P = \{1, 2, 3, \ldots, |P|\}$ be the index set of all patients and $L = \{1, 2, 3, \ldots, |L|\}$ be the index set of all tests. Define $L_p \subseteq L$ as the subset of tests to be performed on patient $p \in P$ and $P_l \subseteq P$ as the subset of patients that require test l. Let $\hat{s}_{pl}$ be the time periods required to perform test l on patient p and $\tilde{s}_p = \sum_{l \in L_p} \hat{s}_{pl}$ be total number of time periods required to perform all tests on patient p. Let T represent the duration of the planning horizon. $\tilde{T}_p, \hat{T}_{pl}$ denote the feasible time periods for starting tests on patient p and the feasible time periods for starting test l on patient p, respectively. Specifically, $\tilde{T}_p = \{1, 2, 3, \ldots, \tilde{T}_p^f\}$, $\tilde{T}_p^f = T - \tilde{s}_p + 1$ and $\hat{T}_{pl} = \{1, 2, 3, \ldots, \hat{T}_{pl}^f\}$, $\hat{T}_{pl}^f = T - \hat{s}_{pl} + 1$.

Consider a room with the assignment of patients in set $\tilde{P} = \{p_1, p_2, p_3, \ldots, p_k\} \subseteq P$. Let t_{p_j} be the first time period in which the room is assigned to patient $p_j \in \tilde{P}$. Without loss of generality, we assume that $t_{p_1} < t_{p_2} < \cdots < t_{p_k}$. With this notation, the schedule of this room over the planning horizon T can be expressed as $\omega = \{(p_j, t_{p_j}) \mid p_j \in \tilde{P}\}$, where $t_{p_{j+1}} \geq t_{p_j} + \tilde{s}_{p_j} \ \forall p_j \in \tilde{P} \setminus p_k$. Additionally, when augmented with dummy nodes, $\{\mathtt{o}\} \cup \omega \cup \{\mathtt{o}'\}$ forms a path from $\mathtt{o}$ to $\mathtt{o}'$ in the Directed Acyclic

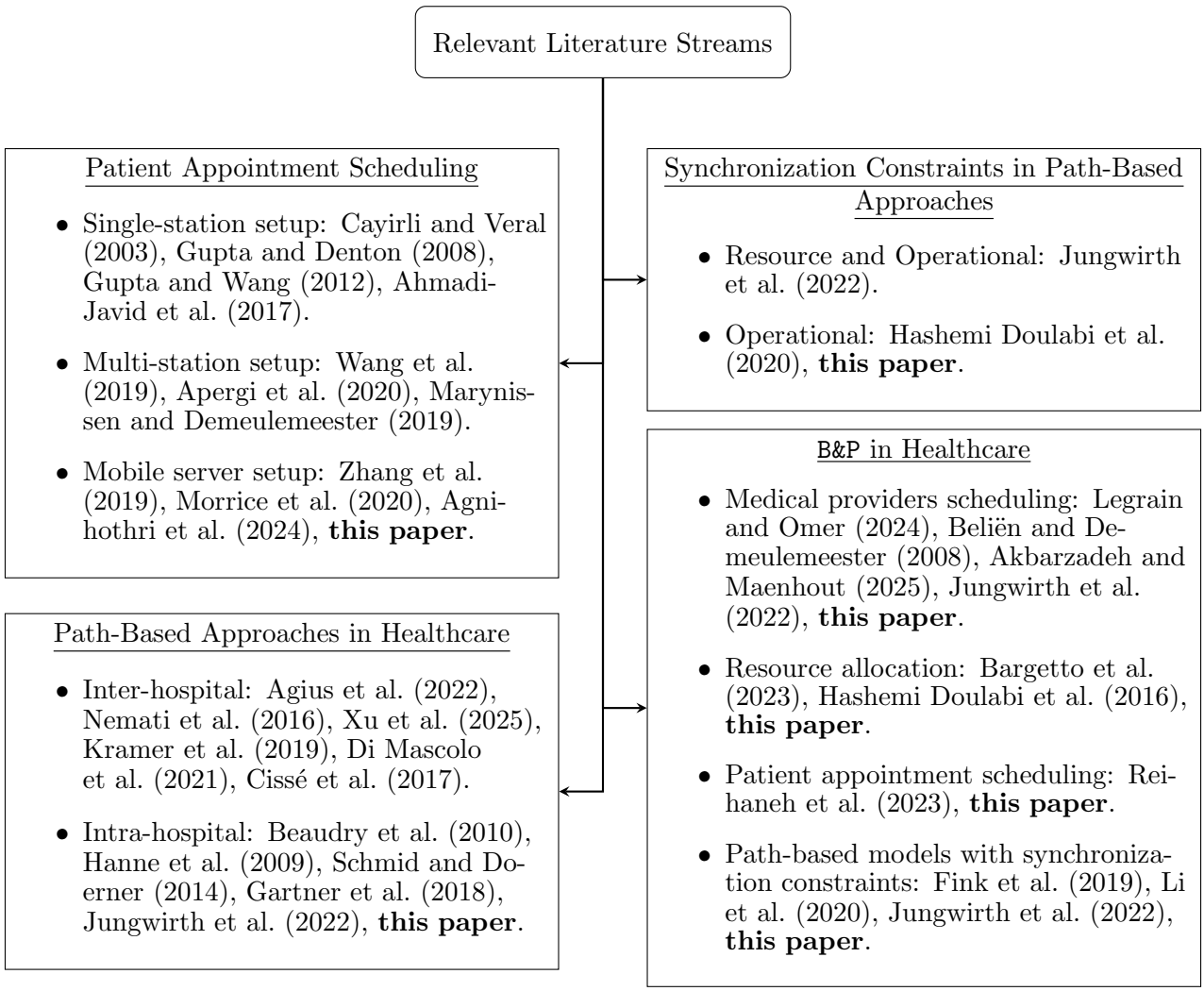


Figure 1: Literature Review Summary.

Graph (DAG), $G(\tilde{V}, \tilde{E})$. Here, $\tilde{V} =: \{i = (i_1, i_2) \mid i_1 \in P, i_2 \in \tilde{T}_{i_1}\} \cup \{\mathtt{o} = (0,0), \mathtt{o}' = (0, T+1)\}$, and $\tilde{E} =: \{(i,j) = ((i_1,i_2),(j_1,j_2)) \mid i \in \tilde{V} \setminus \mathtt{o}', j \in \tilde{V} \setminus \mathtt{o}, i_1 \neq j_1, j_2 > i_2 + \tilde{s}_{i_1} - 1\}$, with $\tilde{s}_0 = 0$. Figure 2 illustrates graph $G(\tilde{V}, \tilde{E})$ for $P = \{1,2,3\}$, $\tilde{s}_1 = 2$, $\tilde{s}_2 = 1$, $\tilde{s}_3 = 2$, and $T = 5$. Similarly, an assignment of a nurse performing test l can be modeled as a path from $\mathtt{o}$ to $\mathtt{o}'$ in the graph $G(\hat{V}_l, \hat{E}_l)$, where $\hat{V}_l = \{i = (i_1, i_2) \mid i_1 \in P_l, i_2 \in \hat{T}_{i_1}\} \cup \{\mathtt{o} = (0,0), \mathtt{o}' = (0, T+1)\}$, and $\hat{E}_l = \{(i,j) = ((i_1,i_2),(j_1,j_2)) \mid i \in \hat{V}_l \setminus \mathtt{o}', j \in \hat{V}_l \setminus \mathtt{o}, i_1 \neq j_1, j_2 > i_2 + \hat{s}_{i_1 l} - 1\}$, with $\hat{s}_{0l} = 0$. Since each test is conducted by a nurse dedicated to that specific test, we use the terms test schedules/paths and nurse schedules/paths interchangeably.

Let Ω be the index set of all feasible room paths in $G(\tilde{V}, \tilde{E})$, and for each $l \in L$, let Π_l be the index set of all feasible paths of test l in $G(\hat{V}_l, \hat{E}_l)$. Let $\tilde{\lambda}_z$ be a binary variable that takes a value of 1 if room path $z \in \Omega$ is selected and 0 otherwise while $\hat{\lambda}^l_z$ be a binary variable that takes the value 1 if path $z \in \Pi_l$ is selected and 0 otherwise. $\tilde{a}_{ptz}$ is a binary parameter that takes value 1 if room path z allocates

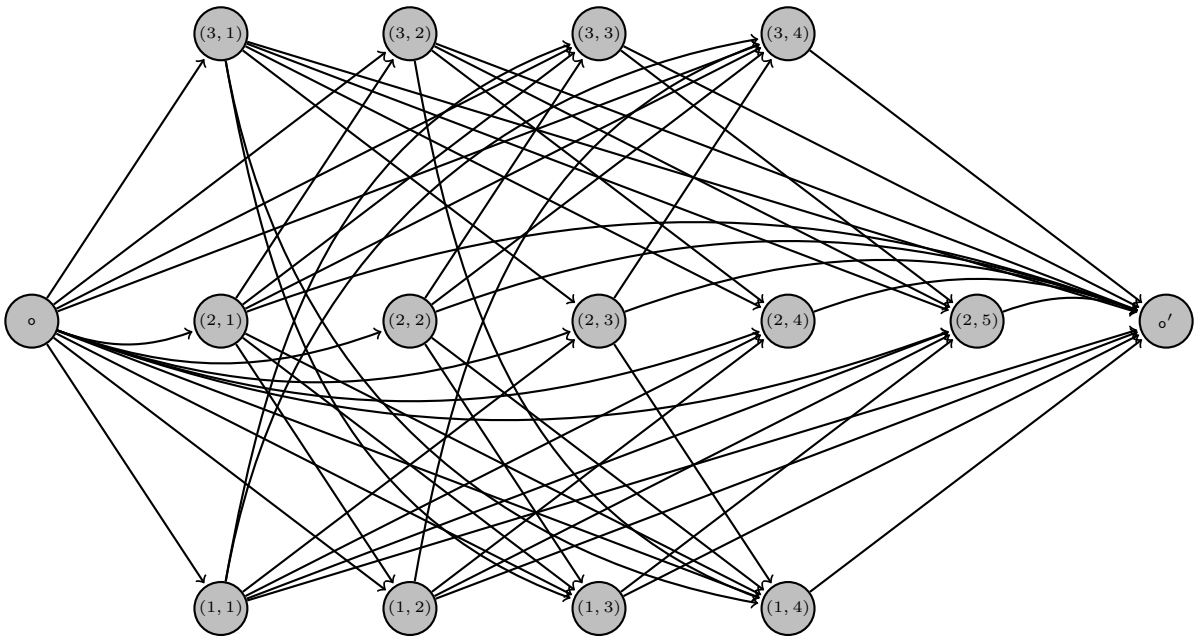


Figure 2: Example of Graph $G(\tilde{V}, \tilde{E})$.

patient p the room at the start of time period t and 0 otherwise while $\hat{a}^l_{ptz}$ is a binary parameter that takes the value 1 if path z of test l start that test on patient p at the beginning of time period t and 0 otherwise. $\tilde{b}_{pt_1t_2z}$ is a binary parameter that takes the value 1 if room path z assigns the room to patient p beginning at time period t_1 and subsequently, right after patient p, allocates the same room to another patient starting at time period t_2 and 0 otherwise. Lastly, $t^*_p = \min\{\hat{s}_{pl} | l \in L_p\}$ is the duration of the shortest test to be performed on patient p. Table 1 summarizes the aforementioned notations. Using these notations, the path-based Dantzig-Wolfe formulation for `PATSP` is given by:

$$(\mathcal{P}) \quad \min \sum_{z\in\Omega} \tilde{\lambda}_z + \sum_{l\in L} \sum_{z\in\Pi_l} \hat{\lambda}^l_z \tag{1a}$$

$$\text{s.t.} \sum_{t\in\tilde{T}_p} \sum_{z\in\Omega} \tilde{a}_{ptz}\tilde{\lambda}_z = 1 \qquad \forall p \in P \tag{1b}$$

$$\sum_{t\in\hat{T}_{pl}} \sum_{z\in\Pi_l} \hat{a}_{ptz}\hat{\lambda}^l_z = 1 \qquad \forall l \in L_p, \forall p \in P \tag{1c}$$

$$\sum_{l\in L_p} \sum_{\tau=\max(1,t-\hat{s}_{pl}+1)}^{\min(t,\hat{T}^f_{pl})} \sum_{z\in\Pi_l} \hat{a}^l_{p\tau z}\hat{\lambda}^l_z \le 1 \qquad \forall t = 1,\ldots,T-t^*_p+1, \forall p \in P \tag{1d}$$

$$\sum_{z\in\Pi_l} \hat{a}^l_{ptz}\hat{\lambda}^l_z \le \sum_{z\in\Omega} \sum_{\tau_1=1}^{\min(t,\tilde{T}^f_p)} \sum_{\tau_2=\max(\tau_1+\tilde{s}_p, t+\hat{s}_{pl})}^{T+1} \tilde{b}_{p\tau_1\tau_2 z}\tilde{\lambda}_z \quad \forall t \in \hat{T}_{pl}, \forall l \in L_p, \forall p \in P \tag{1e}$$

$$\tilde{\lambda}_z \in \{0,1\} \qquad \forall z \in \Omega \tag{1f}$$

Notation	Description
Sets and Parameters	
P	Index set of patients
L	Index set of tests
L_p	Index set of tests required by patient p
P_l	Index set of patients requiring test l
$\hat{s}_{pl}$	Number of time periods required to perform test l on patient p
$\tilde{s}_p$	Number of time periods required to perform all required tests on patient p
T	Number of time periods in the planning horizon
$\tilde{T}_p$	Set of time periods in which tests can start on patient p
$\hat{T}_{pl}$	Set of time periods in which test l can start in patient p
$\tilde{T}_p^f$	Last time period in which any test can start on patient p
$\hat{T}_{pl}^f$	Last time period in which test l on patient p can start
Ω	Index set of room paths
Π_l	Index set of test l paths
$\tilde{a}_{ptz}$	Binary parameter that is 1 if room path z allocates patient p to a room at the start of time period t and 0 otherwise
$\hat{a}_{ptz}^l$	Binary parameter that is 1 if path z of test l starts the test on patient p in time period t and 0 otherwise
$\tilde{b}_{pt_1t_2z}$	Binary parameter that is 1 if room path z allocates patient p to the room at the start of time period t_1 and to another patient right after patient p at the start of time period t_2 and 0 otherwise
t_p^*	Duration of the shortest test required for patient p
Variables	
$\tilde{\lambda}_z$	Binary variable that is 1 if room path z is selected and 0 otherwise.
$\hat{\lambda}_z^l$	Binary variable that is 1 if path z of test l is selected and 0 otherwise

Table 1: Summary of the notations for $\mathcal{P}$.

$$\hat{\lambda}_z^l \in \{0,1\} \qquad \forall z \in \Pi_l, \forall l \in L \tag{1g}$$

Constraints (1b) ensure that each patient is allocated a room, while constraints (1c) ensure that each test $l \in L_p$ is performed on patient p. Constraints (1d) enforce that all tests $l \in L_p$ for each patient $p \in P$ are performed sequentially. Constraints (1e) guarantee that every test $l \in L_p$ required by patient p is conducted after the patient is assigned a room and before that room is reassigned to another patient. Finally, the objective function (1a) seeks to minimize the total number of rooms and nurses needed to complete all required tests for the patients. As the number paths in Ω and $\Pi_l \ \forall l \in L$ are exponential in number, we propose a Branch & Price (`B&P`) Algorithm (Barnhart et al. 1998) to solve $\mathcal{P}$.

4 Branch and Price Algorithm for $\mathcal{P}$

This section presents the details of B&P for solving $\mathcal{P}$. Section 4.1 describes the Restricted Master Problem (RMP), while Section 4.2 focuses on the pricing subproblem. Section 4.3 outlines the branching rule while Section 4.4 discusses the proposed upper-bounding heuristic to generate feasible solutions of PATSP. Section 4.5 derives the lower bounds on the optimal objective function value of PATSP while Section 4.6 gives the symmetry breaking constraints for $\mathcal{P}$.

4.1 Restricted Master Problem (RMP)

B&P involves solving the linear programming relaxation of $\mathcal{P}$ using Column Generation (Bertsimas and Tsitsiklis 1997). Let $\bar{\Omega}^k \subset \Omega$ and $\bar{\Pi}_l^k \subset \Pi_l \ \forall l \in L$ be the subset of paths added till iteration k of Column Generation. Then, RMP in iteration k is given by:

$$(\texttt{RMP}^k) \qquad \min \sum_{z \in \bar{\Omega}^k} \tilde{\lambda}_z + \sum_{l \in L} \sum_{z \in \bar{\Pi}_l^k} \hat{\lambda}_z^l \tag{2a}$$

$$\text{s.t. } [\pi_p^1] \sum_{t \in \tilde{T}_p} \sum_{z \in \bar{\Omega}^k} \tilde{a}_{ptz} \tilde{\lambda}_z = 1 \qquad \forall p \in P \tag{2b}$$

$$[\pi_{pl}^2] \sum_{t \in \hat{T}_{pl}} \sum_{z \in \bar{\Pi}_l^k} \hat{a}_{ptz}^l \hat{\lambda}_z^l = 1 \qquad \forall l \in L_p, \forall p \in P \tag{2c}$$

$$[\pi_{pt}^3] \sum_{l \in L_p} \sum_{\tau = \max(1, t - \hat{s}_{pl} + 1)}^{\min(t, \hat{T}_{pl}^f)} \sum_{z \in \bar{\Pi}_l^k} \hat{a}_{p\tau z}^l \hat{\lambda}_z^l \leq 1 \qquad \forall t = 1, \ldots, T - t_p^* + 1, \forall p \in P \tag{2d}$$

$$[\pi_{plt}^4] \sum_{z \in \bar{\Pi}_l^k} \hat{a}_{ptz}^l \hat{\lambda}_z^l \leq \sum_{z \in \bar{\Omega}^k} \sum_{\tau_1 = 1}^{\min(t, \tilde{T}_p^f)} \sum_{\tau_2 = \max(\tau_1 + \tilde{s}_p, t + \hat{s}_{pl})}^{T+1} \tilde{b}_{p\tau_1\tau_2 z} \tilde{\lambda}_z \quad \forall t \in \hat{T}_{pl}, \forall l \in L_p, \forall p \in P \tag{2e}$$

$$0 \leq \tilde{\lambda}_z \leq 1 \qquad \forall z \in \bar{\Omega}^k \tag{2f}$$

$$0 \leq \hat{\lambda}_z^l \leq 1 \qquad \forall z \in \bar{\Pi}_l^k, \forall l \in L \tag{2g}$$

with dual variables stated in square brackets before each constraint set.

4.2 Pricing Subproblems

Additional columns for $\texttt{RMP}^k$ are determined by solving the pricing subproblem which decomposes into $|L|+1$ subproblems, one for the rooms and one each for $|L|$ tests. The notations used in the

formulations of the pricing subproblems are summarized in Table 2. The reduced cost of a room path variable $\tilde{\lambda}_z$ is given by:

$$\tilde{r}_z = 1 - \sum_{p\in P}\sum_{t\in\tilde{T}_p} \pi^1_p \tilde{a}_{ptz} + \sum_{p\in P}\sum_{\tau_1\in\tilde{T}_p}\sum_{\tau_2=\tau_1+\tilde{s}_p}^{T+1}\sum_{l\in L_p}\sum_{\tau_3=\tau_1}^{\tau_2-\hat{s}_{pl}} \pi^4_{pl\tau_3}\tilde{b}_{p\tau_1\tau_2 z}, \tag{3}$$

while the reduced cost of test path variable $\hat{\lambda}^l_z$ is stated as follows:

$$\hat{r}^l_z = 1 - \sum_{p\in P_l}\sum_{t\in\hat{T}_l}\left(\pi^2_{pl} + \sum_{\tau=max(t^*_p,t)}^{min(\hat{s}_{pl}+t-1,T-t^*_p+1)} \pi^3_{pt} + +\pi^4_{plt}\right)\hat{a}_{ptz}. \tag{4}$$

For each $(i,j)\in\tilde{E}$ of $G(\tilde{V},\tilde{E})$, let

$$\tilde{h}_{ij} = \begin{cases} -\pi^1_{j_1} & \text{if } i = \mathtt{o}, \\ \sum_{l\in L_{i_1}}\sum_{t=i_2}^{j_2-\hat{s}_{i_1 l}} \pi^4_{i_1 lt} & \text{if } j = \mathtt{o}', \\ -\pi^1_{j_1} + \sum_{l\in L_{i_1}}\sum_{t=i_2}^{j_2-\hat{s}_{i_1 l}} \pi^4_{i_1 lt} & \text{otherwise.} \end{cases} \tag{5}$$

The room pricing subproblem can be formulated as identifying the shortest path between nodes $\mathtt{o}$ and $\mathtt{o}'$ in $G(\tilde{V},\tilde{E})$, ensuring that nodes corresponding to each patient $p\in P$ are visited at most once. The length of each arc $(i,j)\in\tilde{E}$ is determined by $\tilde{h}_{ij}$. Let y_{ij} is a binary variable that takes the value 1 if arc $(i,j)\in\tilde{E}$ is in the shortest path and 0, otherwise; $\delta^+(i) = \{j\in\tilde{V}|(i,j)\in\tilde{E}\}$ and $\delta^-(i) = \{j\in\tilde{V}|(j,i)\in\tilde{E}\}$. Using these notations, the room pricing subproblem is given by the following Binary Integer Program (BIP):

$$(\widetilde{\mathtt{SP}}) \qquad \tilde{\eta} = \min \sum_{(i,j)\in\tilde{E}} \tilde{h}_{ij} y_{ij} \tag{6a}$$

$$\text{s.t.} \sum_{j\in\delta^+(\mathtt{o})} y_{\mathtt{o}j} = 1 \tag{6b}$$

$$-\sum_{j\in\delta^-(\mathtt{o}')} y_{j\mathtt{o}'} = -1 \tag{6c}$$

$$\sum_{j\in\delta^+(i)} y_{ij} - \sum_{j\in\delta^-(i)} y_{ji} = 0 \qquad \forall i\in\tilde{V}\setminus\{\mathtt{o},\mathtt{o}'\} \tag{6d}$$

$$\sum_{j:j_1=p}\sum_{i\in\delta^-(j)} y_{ij} \le 1 \qquad \forall p\in P \tag{6e}$$

$$y_{ij}\in\{0,1\} \qquad \forall (i,j)\in\tilde{E} \tag{6f}$$

Notation	Description
Sets and Parameters	
$\tilde{V}$	Set of nodes for the room subproblem
$\hat{V}_l$	Set of nodes for the test l subproblem
$\tilde{E}$	Set of arcs for the room subproblem
$\hat{E}_l$	Set of arcs for the test l subproblem
$\tilde{h}_{ij}$	Cost of arc (i,j) in room subproblem
$\hat{h}^l_{ij}$	Cost of arc (i,j) in test l subproblem
$\delta^-(i)$	Set of nodes that can directly reach node i
$\delta^+(i)$	Set of nodes that can be directly reached from node i
$\mathtt{o}$, $\mathtt{o}'$	Dummy nodes representing the start and the end of a path, respectively
Variables	
y_{ij}	Binary variable that is 1 if arc (i,j) is selected and 0 otherwise

Table 2: Summary of the notations for $\widetilde{\mathtt{SP}}$ and $\widehat{\mathtt{SP}}^l$.

Constraints (6b) and (6c) ensure that the shortest path starts at $\mathtt{o}$ and ends at $\mathtt{o}'$, respectively. Constraints (6d) give the flow balance constraints at all the nodes in $\tilde{V} \setminus \{\mathtt{o}, \mathtt{o}'\}$ while Constraints (6e) ensure that the shortest path visits at most one node associated with each patient $p \in P$. Objective function (6a) minimizes the length of the path from $\mathtt{o}$ to $\mathtt{o}'$ where length of an arc $(i,j) \in \tilde{E}$ is given by $\tilde{h}_{ij}$. If $1 + \tilde{\eta} < 0$, then optimal solution of $\widetilde{\mathtt{SP}}$ yields a room path with negative reduced cost that is added to $\mathtt{RMP}^k$ to construct $\mathtt{RMP}^{k+1}$.

Similarly, the pricing subproblem for test l can be formulated as identifying shortest path from nodes $\mathtt{o}$ and $\mathtt{o}'$ in $G(\hat{V}_l, \hat{E}_l)$ such that nodes pertaining to each patient $p \in P_l$ is included at-most once in the selected path. Let the length of each arc $(i,j) \in \hat{E}_l$ is given by

$$\hat{h}^l_{ij} = \begin{cases} -\pi^2_{j_1 l} - \pi^4_{j_1 l j_2} - \sum_{t=j_2}^{min(\hat{s}_{pl}+j_2-1, T-t^*_{j_1}+1)} \pi^3_{j_1 t} & \text{if } j \neq \mathtt{o}', \\ 0 & \text{if } j = \mathtt{o}'. \end{cases} \tag{7}$$

Analogous to $\widetilde{\mathtt{SP}}$, the pricing subproblem for test l is given by the following BIP:

$$(\widehat{\mathtt{SP}}^l) \qquad \hat{\eta}^l = \min \sum_{(i,j)\in\hat{E}_l} \hat{h}^l_{ij} y_{ij} \tag{8a}$$

$$\text{s.t.} \sum_{j\in\delta^+(\mathtt{o})} y_{\mathtt{o}j} = 1 \tag{8b}$$

$$- \sum_{j\in\delta^-(\mathtt{o}')} y_{j\mathtt{o}'} = -1 \tag{8c}$$

$$\sum_{j\in\delta^+(i)} y_{ij} - \sum_{j\in\delta^-(i)} y_{ji} = 0 \qquad \forall i \in \hat{V}_l \setminus \{\mathtt{o}, \mathtt{o}'\} \tag{8d}$$

$$\sum_{j:j_1=p} \sum_{i \in \delta^-(j)} y_{ij} \leq 1 \qquad \forall p \in P_l \tag{8e}$$

$$y_{ij} \in \{0,1\} \qquad \forall (i,j) \in \hat{E}_l \tag{8f}$$

The description of $\widehat{\texttt{SP}}^l$ is similar to that of $\widetilde{\texttt{SP}}$. If $\hat{\eta}^l + 1 < 0$, the optimal solution of $\widehat{\texttt{SP}}^l$ gives a path of test l with negative reduced cost that is added to $\texttt{RMP}^k$ to construct $\texttt{RMP}^{k+1}$.

4.2.1 Valid Inequalities for the Pricing Subproblem

Pricing subproblems $\widetilde{\texttt{SP}}$ and $\widehat{\texttt{SP}}^l$ can be interpreted as an extension of the Elementary Shortest Path Problem (ESPP) (Taccari 2016). While the ESPP requires that each node in the graph to be visited at most once, $\widetilde{\texttt{SP}}$ and $\widehat{\texttt{SP}}^l$ generalize this by imposing the condition that a subset of nodes is restricted to being visited at most once. Additionally, unlike the pricing subproblems in the Vehicle Routing Problem (Feillet et al. 2004), $\widetilde{\texttt{SP}}$ and $\widehat{\texttt{SP}}^l$ do not include any resource constraints. Therefore, we follow the approach of Taccari (2016) and solve $\widetilde{\texttt{SP}}$ and $\widehat{\texttt{SP}}^l$ using the Branch and Cut algorithm as implemented by a commercial MILP solver. Furthermore, to enhance the computational efficiency of the Branch and Cut algorithm, we introduce a family of Valid Inequalities (VIs) for $\widetilde{\texttt{SP}}$. Analogous VIs can be easily derived for $\widehat{\texttt{SP}}^l$, but they are omitted here to avoid redundancy.

Let $\delta_p^+(v) = \{j \in \tilde{V} | (v,j) \in \tilde{E}, j_1 = p\}$, $\delta_{\bar{p}}^+(v) = \{j \in \tilde{V} | (v,j) \in \tilde{E}, j_1 \neq p\}$, $\delta_p^-(v) = \{i \in \tilde{V} | (i,v) \in \tilde{E}, i_1 = p\}$ and $\delta_{\bar{p}}^-(v) = \{i \in \tilde{V} | (i,v) \in \tilde{E}, i_1 \neq p\}$. $\delta_p^+(v)$ denote the set of nodes in $\tilde{V}$ that are directly reachable from v and are associated with patient p while $\delta_{\bar{p}}^+(v)$ denote the set of nodes in $\tilde{V}$ that are directly reachable from v but are not associated with patient p. Similarly, $\delta_p^-(v)$ refers to the set of nodes in $\tilde{V}$ that can directly reach v and are associated with patient p while $\delta_{\bar{p}}^-(v)$ denote the set of nodes in $\tilde{V}$ that can directly reach v but are not associated with patient p. With these notations, the following proposition gives two families of VIs for $\widetilde{\texttt{SP}}$.

Proposition 1. *The following inequalities are valid for* $\widetilde{\texttt{SP}}$*:*

$$\sum_{i \in \delta_p^-(\nu)} y_{i\nu} \leq \sum_{i \in \delta_{\bar{p}}^+(\nu)} y_{\nu i} \quad \forall \nu = (\nu_1, \nu_2) \in \tilde{V} \setminus \{\texttt{o}, \texttt{o}'\},\ p \in P \setminus \nu_1 \tag{9}$$

$$\sum_{i \in \delta_{\bar{p}}^-(\nu)} y_{i\nu} \geq \sum_{i \in \delta_p^+(v)} y_{\nu i} \quad \forall \nu = (\nu_1, \nu_2) \in \tilde{V} \setminus \{\texttt{o}, \texttt{o}'\},\ p \in P \setminus \nu_1 \tag{10}$$

Proof. (9): Let $\boldsymbol{y}^*$ be a feasible solution of $\widetilde{\texttt{SP}}$. Assume that there exists a node $\nu = (\nu_1, \nu_2)$ and

patient $p \neq \nu_1$ such that

$$\sum_{i \in \delta_p^-(\nu)} y^*_{i\nu} > \sum_{i \in \delta_{\bar{p}}^+(\nu)} y^*_{\nu i} \geq 0. \tag{11}$$

Due to constraints in (6), we know that $\sum_{i \in \delta_p^-(\nu)} y^*_{i\nu} \leq 1$ and $\sum_{i \in \delta_{\bar{p}}^+(\nu)} y^*_{\nu i} \leq 1$. This implies, due to (11), $\sum_{i \in \delta_p^-(\nu)} y^*_{i\nu} = 1$, $\sum_{i \in \delta_{\bar{p}}^+(\nu)} y^*_{\nu i} = 0$ and $\exists j \in \delta_p^-(\nu)$ such that $y^*_{j\nu} = 1$. However, as $y^*_{j\nu} = 1$, in order to satisfy (6d) and (6e), $\exists q \in \delta_{\bar{p}}^+(\nu)$ such that $y^*_{\nu q} = 1$, which is a contradiction.

(10): For each node $\nu \in (\nu_1, \nu_2)$ and some patient $p \neq \nu_1$, (6d) is equivalent to

$$\sum_{i \in \delta_p^-(\nu)} y_{i\nu} + \sum_{i \in \delta_{\bar{p}}^-(\nu)} y_{i\nu} = \sum_{i \in \delta_p^+(\nu)} y_{\nu i} + \sum_{i \in \delta_{\bar{p}}^+(\nu)} y_{\nu i}$$

$$\Rightarrow \sum_{i \in \delta_p^-(\nu)} y_{i\nu} - \sum_{i \in \delta_p^+(\nu)} y_{\nu i} = - \sum_{i \in \delta_{\bar{p}}^-(\nu)} y_{i\nu} + \sum_{i \in \delta_{\bar{p}}^+(\nu)} y_{\nu i} \geq 0$$

where the last inequality is due to (9). □

VIs (9) ensures that if there is a flow of one unit into node ν from a node pertaining to patient $p \neq \nu_1$, then there is one unit of flow from ν to a node not pertaining to patient p. Similarly, VIs (10) ensure that if there is a flow of one unit from node ν to a node pertaining to patient $p \neq \nu_1$, then there is one unit of flow into ν from a node not pertaining to patient p. Both of these VIs are aligned with the condition enforced by constraints (6e).

Let $\boldsymbol{Y}$ be the set of feasible solutions of $\widetilde{\texttt{SP}}$. We demonstrate the strength of VIs (9) and (10) by showing that they define facets of $conv(\boldsymbol{Y})$. Let $\mathbf{1}_J$ be a vector of length $|\tilde{E}|$ in which all entries are 0 except for those pertaining to $J \subseteq \tilde{E}$, which are equal to 1. In the following proposition, we determine $dim(conv(\boldsymbol{Y}))$.

Proposition 2. *$dim(conv(\boldsymbol{Y})) = |\tilde{E}| - |\tilde{V}| + 1$ where $|\tilde{E}| = \sum_{v \in \tilde{V}} |\delta^+(v)|$.*

Proof. $\widetilde{\texttt{SP}}$ consists of $|\tilde{E}| = \sum_{v \in \tilde{V}} |\delta^+(v)|$ variables. Consider the following points:

1. $\boldsymbol{y}^1_J = \mathbf{1}_J,\ \forall J \in \tilde{Q}_1,\ \tilde{Q}_1 = \bigcup_{v \in \tilde{V} \setminus \{\texttt{o}, \texttt{o}'\}} \{(\texttt{o}, v), (v, \texttt{o}')\}$.

2. $\boldsymbol{y}^2_J = \mathbf{1}_J,\ \forall J \in \tilde{Q}_2,\ \tilde{Q}_2 = \bigcup_{\substack{v \in \tilde{V} \setminus \{\texttt{o}, \texttt{o}'\} \\ \bar{v} \in \delta^+(v) \setminus \{\texttt{o}'\}}} \{(\texttt{o}, v), (v, \bar{v}), (\bar{v}, \texttt{o}')\}$.

It can be easily verified that points $[\boldsymbol{y}^1_J]_{J \in \tilde{Q}_1}, [\boldsymbol{y}^2_J]_{J \in \tilde{Q}_2} \in \boldsymbol{Y}$. As $|\tilde{Q}_1| = |\tilde{V}| - 2$ and $|\tilde{Q}_2| = |\tilde{E}| - 2\left(|\tilde{V}| - 2\right)$, the total number of such points is $|\tilde{E}| - |\tilde{V}| + 2$. Furthermore, these points are linearly independent since each one corresponds uniquely to the flow of one unit through each of the $|\tilde{E}| - |\tilde{V}| + 2$ arcs in $\tilde{E} \setminus \cup_{v \in \delta^+(\texttt{o})} \{(\texttt{o}, v)\}$. As linearly independent points are also affinely independent, $\dim(conv(\boldsymbol{Y})) \geq$

$|\tilde{E}|-|\tilde{V}|+1$. Moreover, as $|\tilde{V}|-1$ equality constraints in $\widetilde{\texttt{SP}}$ are linearly independent (Wolsey 2020), $dim(conv(\boldsymbol{Y})) \leq |\tilde{E}|-|\tilde{V}|+1$. Thus, $dim(conv(\boldsymbol{Y})) = |\tilde{E}|-|\tilde{V}|+1$. □

Next, in Propositions 3 and 4, we show that VIs (9) and (10) define facets of $conv(\boldsymbol{Y})$.

Proposition 3. VI (9) *for some* $\nu = (\nu_1, \nu_2) \in \tilde{V} \setminus \{\mathtt{o}, \mathtt{o}'\}$ *and* $p \in P \setminus \nu_1$ *is facet-defining for* $conv(\boldsymbol{Y})$ *if* $\delta_p^+(\nu) \neq \emptyset$ *and* $\delta_p^-(\nu) \neq \emptyset$.

Proof. Let $\mathcal{F}^1_{\nu,p} = \{\boldsymbol{y} \in conv(\boldsymbol{Y}) \mid \sum_{i\in\delta_p^-(\nu)} y_{i\nu} = \sum_{i\in\delta_{\bar{p}}^+(\nu)} y_{\nu i}\}$ for some $\nu = (\nu_1, \nu_2) \in \tilde{V} \setminus \{\mathtt{o}, \mathtt{o}'\}$ and $p \in P \setminus \nu_1$ such that $\delta_p^+(\nu) \neq \emptyset$ and $\delta_p^-(\nu) \neq \emptyset$. Consider the following points:

1. $\boldsymbol{y}^1_J = \mathbf{1}_J, \forall J \in \hat{Q}_1,\ \hat{Q}_1 = \bigcup_{v\in\tilde{V}\setminus\{\mathtt{o},\mathtt{o}',\nu\}}\{(\mathtt{o}, v), (v, \mathtt{o}')\}$; $|\hat{Q}_1| = |\tilde{V}|-3$.
2. $\boldsymbol{y}^2_J = \mathbf{1}_J, \forall J \in \hat{Q}_2,\ \hat{Q}_2 = \bigcup_{\substack{v\in\tilde{V}\setminus\{\mathtt{o},\mathtt{o}',\nu\} \\ \bar{v}\in\delta^+(v)\setminus\{\mathtt{o}',\nu\}}} \{(\mathtt{o}, v), (v, \bar{v}), (\bar{v}, \mathtt{o}')\}$; $|\hat{Q}_2| = |\tilde{E}|-|\delta^+(\nu)|-|\delta^-(\nu)|-2\left(|\tilde{V}|-3\right)$.
3. $\boldsymbol{y}^3_J = \mathbf{1}_J, \forall J \in \hat{Q}_3,\ \hat{Q}_3 = \bigcup_{v\in\delta_{\bar{p}}^-(\nu)\setminus\{\mathtt{o}\}}\{(\mathtt{o}, v), (v, \nu), (\nu, \bar{v}), (\bar{v}, \mathtt{o}')\}$ for some $\bar{v} \in \delta_p^+(\nu)$; $|\hat{Q}_3| = |\delta_{\bar{p}}^-(\nu)|-1$.
4. $\boldsymbol{y}^4_J = \mathbf{1}_J, \forall J \in \hat{Q}_4,\ \hat{Q}_4 = \bigcup_{v\in\delta_p^+(\nu)}\{(\mathtt{o}, \nu), (\nu, v), (v, \mathtt{o}')\}$; $|\hat{Q}_4| = |\delta_p^+(\nu)|$.
5. $\boldsymbol{y}^5_J = \mathbf{1}_J, \forall J \in \hat{Q}_5,\ \hat{Q}_5 = \bigcup_{v\in\delta_p^-(\nu)}\{(\mathtt{o}, v)(v, \nu)(\nu, \mathtt{o}')\}$; $|\hat{Q}_5| = |\delta_p^-(\nu)|$.
6. $\boldsymbol{y}^6_J = \mathbf{1}_J, \forall J \in \hat{Q}_6,\ \hat{Q}_6 = \bigcup_{v\in\delta_{\bar{p}}^+(\nu)\setminus\{\mathtt{o}'\}}\{(\mathtt{o}, \bar{v}), (\bar{v}, \nu), (\nu, v), (v, \mathtt{o}')\}$ for some $\bar{v} \in \delta_p^-(\nu)$; $|\hat{Q}_6| = |\delta_{\bar{p}}^+(\nu)|-1$.

It can be easily verified that all these points belong to $\mathcal{F}^1_{\nu,p}$. As $|\delta^+(v)| = |\delta_{\bar{p}}^+(\nu)|+|\delta_p^+(\nu)|$ and $|\delta^-(\nu)| = |\delta_{\bar{p}}^-(\nu)|+|\delta_p^-(\nu)|$, the total number of these points $\sum_{i=1}^6|\hat{Q}_i| = |\tilde{E}|-|\tilde{V}|+1 = dim(conv(\boldsymbol{Y}))$. Additionally, these points are linearly independent because each one uniquely represents the flow of one unit through each of the $|\tilde{E}|-|\tilde{V}|+1$ arcs in $\tilde{E} \setminus \left(\left(\cup_{v\in\delta^+(\mathtt{o})}\{(\mathtt{o}, v)\}\right) \cup (\nu, \mathtt{o}')\right)$. This implies that $\dim(\mathcal{F}^1_{\nu,p}) \geq \dim(conv(\boldsymbol{Y})) - 1$.

Consider the point $\boldsymbol{y}_{J_1} = \mathbf{1}_{J_1}$, $J_1 = \{(\mathtt{o}, v), (v, \nu), (\nu, \mathtt{o}')\}$, for some $v \in \delta_p^-(\nu)$ and the point $\boldsymbol{y}_{J_2} = \mathbf{1}_{J_2}$, $J_2 = \{(\mathtt{o}, \nu), (\nu, \mathtt{o}')\}$. We observe that $\boldsymbol{y}_{J_1}, \boldsymbol{y}_{J_2} \in \boldsymbol{Y}$, $\boldsymbol{y}_{J_1} \in \mathcal{F}^1_{\nu,p}$ and $\boldsymbol{y}_{J_2} \notin \mathcal{F}^1_{\nu,p}$. Hence, $\mathcal{F}^1_{\nu,p}$ is a proper face of $conv(\boldsymbol{Y})$, and therefore, $\dim(\mathcal{F}^1_{\nu,p}) \leq \dim(conv(\boldsymbol{Y})) - 1$. This implies $\dim(\mathcal{F}^1_{\nu,p}) = \dim(conv(\boldsymbol{Y})) - 1$. □

Proposition 4. VI (10) *for some* $\nu = (\nu_1, \nu_2) \in \tilde{V} \setminus \{\mathtt{o}, \mathtt{o}'\}$ *and* $p \in P \setminus \nu_1$ *is facet-defining for* $conv(\boldsymbol{Y})$ *if* $\delta_p^+(\nu) \neq \emptyset$ *and* $\delta_p^-(\nu) \neq \emptyset$.

Proof. Let $\mathcal{F}^2_{\nu,p} = \{\boldsymbol{y} \in conv(\boldsymbol{Y}) \mid \sum_{i\in\delta_p^+(\nu)} y_{\nu i} = \sum_{i\in\delta_{\bar{p}}^-(\nu)} y_{i\nu}\}$ for some $\nu = (\nu_1, \nu_2) \in \tilde{V} \setminus \{\mathtt{o}, \mathtt{o}'\}$ and $p \in P \setminus \nu_1$ such that $\delta_p^+(\nu) \neq \emptyset$ and $\delta_p^-(\nu) \neq \emptyset$. Consider the following points:

1. $\boldsymbol{y}^1_J = \mathbf{1}_J,\ \forall J \in \hat{Q}_1, \hat{Q}_1 = \bigcup_{v \in \tilde{V} \setminus \{\mathtt{o}, \mathtt{o}', \nu\}} \{(\mathtt{o}, v), (v, \mathtt{o}')\};\ |\hat{Q}_1| = |\tilde{V}| - 3.$

2. $\boldsymbol{y}^2_J = \mathbf{1}_J,\ \forall J \in \hat{Q}_2,\ \hat{Q}_2 = \bigcup_{\substack{v \in \tilde{V} \setminus \{\mathtt{o}, \mathtt{o}', \nu\} \\ \bar{v} \in \delta^+(v) \setminus \{\mathtt{o}', \nu\}}} \{(\mathtt{o}, v), (v, \bar{v}), (\bar{v}, \mathtt{o}')\};\ |\hat{Q}_2| = |\tilde{E}| - |\delta^+(\nu)| - |\delta^-(\nu)| - 2\left(|\tilde{V}| - 3\right).$

3. $\boldsymbol{y}^3_J = \mathbf{1}_J,\ \forall J \in \hat{Q}_3, \hat{Q}_3 = \bigcup_{v \in \delta^+_{\bar{p}}(\nu) \setminus \{\mathtt{o}'\}} \{(\mathtt{o}, \bar{v}), (\bar{v}, \nu)(\nu, v), (v, \mathtt{o}')\}$ for some $\bar{v} \in \delta^-_p(\nu);\ |\hat{Q}_3| = |\delta^+_{\bar{p}}(\nu)| - 1.$

4. $\boldsymbol{y}^4_J = \mathbf{1}_J,\ \forall J \in \hat{Q}_4,\ \hat{Q}_4 = \bigcup_{v \in \delta^-_p(\nu)} \{(\mathtt{o}, v), (v, \nu), (\nu, \mathtt{o}')\};\ |\hat{Q}_4| = |\delta^-_p(\nu)|.$

5. $\boldsymbol{y}^5_J = \mathbf{1}_J,\ \forall J \in \hat{Q}_5, \hat{Q}_5 = \bigcup_{v \in \delta^+_p(\nu)} \{(\mathtt{o}, \nu)(\nu, v)(v, \mathtt{o}')\};\ |\hat{Q}_5| = |\delta^+_p(\nu)|.$

6. $\boldsymbol{y}^6_J = \mathbf{1}_J,\ \forall J \in \hat{Q}_6, \hat{Q}_6 = \bigcup_{v \in \delta^-_{\bar{p}}(\nu) \setminus \{\mathtt{o}\}} \{(\mathtt{o}, v), (v, \nu), (\nu, \bar{v}), (\bar{v}, \mathtt{o}')\}$ for some $\bar{v} \in \delta^+_p(\nu);\ |\hat{Q}_6| = |\delta^-_{\bar{p}}(\nu)| - 1.$

It is straightforward to confirm that all these points lie within $\mathcal{F}^2_{\nu,p}$. The total number of these points is $\sum_{i=1}^{6} |\hat{Q}_i| = |\tilde{E}| - |\tilde{V}| + 1 = \dim(conv(\boldsymbol{Y}))$. Also, these points are linearly independent because each one uniquely represents the flow of one unit through each of the $|\tilde{E}| - |\tilde{V}| + 1$ arcs in $\tilde{E} \setminus \left(\left(\cup_{v \in \delta^+(\mathtt{o})} \{(\mathtt{o}, v)\}\right) \cup (\nu, \mathtt{o}')\right)$, implying that $\dim(\mathcal{F}^2_{\nu,p}) \geq \dim(conv(\boldsymbol{Y})) - 1$. Moreover, consider the point $\boldsymbol{y}_{J_1} = \mathbf{1}_{J_1}$, where $J_1 = \{(\mathtt{o}, \nu), (\nu, v), (v, \mathtt{o}')\}$ for some $v \in \delta^+_p(\nu)$, and the point $\boldsymbol{y}_{J_2} = \mathbf{1}_{J_2}$, where $J_2 = \{(\mathtt{o}, \nu), (\nu, \mathtt{o}')\}$. Note that $\boldsymbol{y}_{J_1}, \boldsymbol{y}_{J_2} \in \boldsymbol{Y}$, $\boldsymbol{y}_{J_1} \in \mathcal{F}^2_{\nu,p}$ but $\boldsymbol{y}_{J_2} \notin \mathcal{F}^2_{\nu,p}$. Therefore, $\mathcal{F}^2_{\nu,p}$ is a proper face of $conv(\boldsymbol{Y})$, and $\dim(\mathcal{F}^2_{\nu,p}) \leq \dim(conv(\boldsymbol{Y})) - 1$. This implies that $\dim(\mathcal{F}^2_{\nu,p}) = \dim(conv(\boldsymbol{Y})) - 1$. □

4.3 Branching Rule

In our implementation of the B&P algorithm, we use two branching rules, `Branching Rule 1` and `Branching Rule 2`. A single application of `Branching Rule 1` eliminates multiple nodes from $G(\tilde{V}, \tilde{E})$ or $G(\hat{V}_l, \hat{E}_l)\ \forall l \in L$, while a single application of `Branching Rule 2` removes several arcs from the same graphs. For a fractional solution of the RMP, `Branching Rule 1` is prioritized and if `Branching Rule 1` is insufficient to eliminate the fractional solution, we then apply `Branching Rule 2`. We demonstrate that the combined implementation of `Branching Rule 1` and `Branching Rule 2` will always exclude a fractional solution. In the following subsections, we explain two branching rules.

4.3.1 `Branching Rule 1`

`Branching Rule 1` is motivated by the Special Ordered Sets of Type 1 (SOS1) branching rule stated in Fischer and Pfetsch (2018). Let $\kappa_{pt} = \sum_{z \in \bar{\Omega}} \tilde{a}_{ptz} \tilde{\lambda}_z,\ \forall p \in P,\ t \in \tilde{T}_p$, and $\zeta^l_{pt} = \sum_{z \in \bar{\Pi}_l} \hat{a}^l_{ptz} \hat{\lambda}^l_z,\ \forall l \in L,\ p \in P_l,\ t \in \hat{T}_{pl}$. κ_{pt} is a binary variable that takes the value 1 if patient p is allocated a room at

the start of period t and 0 otherwise. Similarly, ζ^l_{pt} is a binary variable that takes the value 1 if test l starts on patient p at the start of period t and 0 otherwise. From (1b) and (1c) we infer that the following equality constraints are satisfied by all feasible solutions of $\mathcal{P}$:

$$\sum_{t\in\tilde{T}_p} \kappa_{pt} = 1 \qquad \forall p \in P \tag{12a}$$

$$\sum_{t\in\hat{T}_{pl}} \zeta^l_{pt} = 1 \qquad \forall p \in P_l, \forall l \in L \tag{12b}$$

Let $\tilde{\boldsymbol{\lambda}}^*$ and $\hat{\boldsymbol{\lambda}}^{l^*} \forall l \in L$ constitute a fractional solution of $\texttt{RMP}^k$. Define $\kappa^*_{pt} = \sum_{z\in\bar{\Omega}^k} \tilde{a}_{ptz}\tilde{\lambda}^*_z,\ \forall p \in P,\ t \in \tilde{T}_p,\ \zeta^{l^*}_{pt} = \sum_{z\in\bar{\Pi}^k_l} \hat{a}^l_{ptz}\hat{\lambda}^{l^*}_z,\ \forall l \in L, \forall p \in P_l, \forall t \in \hat{T}_{pl},\ \tilde{U}^*_p = \{t \in \tilde{T}_p | \kappa^*_{pt} > 0\}\ \forall p \in P$ and $\hat{U}^{l^*}_p = \{t \in \hat{T}_{pl} | \zeta^{l^*}_{pt} > 0\}\ \forall l \in L, \forall p \in P_l$. Let $\tilde{p} \in P$ be such that $|\tilde{U}^*_{\tilde{p}}| \geq |\tilde{U}^*_p|\ \forall p \in P$, and $\hat{l} \in L$, $\hat{p} \in P$ be such that $|\hat{U}^{\hat{l}^*}_{\hat{p}}| \geq |\hat{U}^{l^*}_p|\ \forall l \in L, \forall p \in P_l$. If $|\tilde{U}^*_{\tilde{p}}| \geq 2$, let $\{\tilde{t}_1, \tilde{t}_2, \ldots, \tilde{t}_{\tilde{T}^f_{\tilde{p}}}\}$ be such that $\kappa^*_{\tilde{p}\tilde{t}_1} \geq \kappa^*_{\tilde{p}\tilde{t}_2} \geq \ldots, \geq \kappa^*_{\tilde{p}\tilde{t}_{\tilde{T}^f_{\tilde{p}}}}$ and define $\mathcal{S}_1 = \{(\tilde{p}, \tilde{t}_j) | j \mod 2 = 0\}$ and $\mathcal{S}_2 = \{(\tilde{p}, \tilde{t}_j) | j \mod 2 = 1\}$. On the left branch we set the flow though the node set $\mathcal{S}_1$ equal to zero while on the right branch we set the flow though the node set $\mathcal{S}_2$ equal to zero. Alternatively, if $|\tilde{U}^*_{\tilde{p}}| = 1$ and $|\hat{U}^{\hat{l}^*}_{\hat{p}}| \geq 2$ then analogous steps are applied to obtain node sets $\mathcal{S}_1$ and $\mathcal{S}_2$ for patient $\hat{p} \in P_{\hat{l}}$. (1b) and (1c) ensure that this branching scheme is valid and doesn't exclude any integer feasible solution.

$|\tilde{U}^*_{\tilde{p}}| \geq 2$ or $|\hat{U}^{\hat{l}^*}_{\hat{p}}| \geq 2$ ensure that $\mathcal{S}_1, \mathcal{S}_2 \neq \emptyset$. As the total flow through these node sets in the fractional solution is non-zero, this branching scheme exclude the fractional solution $\tilde{\boldsymbol{\lambda}}^*$ and $\hat{\boldsymbol{\lambda}}^{l^*} \forall l \in L$. Furthermore, this branching scheme can be easily incorporated in `B&P` by excluding the nodes in $\mathcal{S}_1$, $\mathcal{S}_2$ from the pricing subproblem of the respective children nodes. Lastly, if $|\tilde{U}^*_{\tilde{p}}| = |\hat{U}^{\hat{l}^*}_{\hat{p}}| = 1$, `Branching Rule 1` will not exclude the fractional solution as, in this case, $\mathcal{S}_1 = \emptyset$. Therefore, we implement `Branching Rule 2` as described in the next subsection.

4.3.2 `Branching Rule 2`

We implement `Branching Rule 2` when $|\tilde{U}^*_{\tilde{p}}| = |\hat{U}^{\hat{l}^*}_{\hat{p}}| = 1$. As $|\tilde{U}^*_{\tilde{p}}| = 1$, for each $p \in P$, there exists a $\tilde{t}_p \in \tilde{T}_p$ such that $\kappa^*_{p\tilde{t}_p} = \sum_{z\in\bar{\Omega}^k} \tilde{a}_{p\tilde{t}_p z}\tilde{\lambda}^*_z = 1$. Let $i_p = (p, \tilde{t}_p)$ be the corresponding node in $\tilde{V}$ and $\theta^*_{ji_p} = \sum_{z\in\bar{\Omega}^k} \tilde{d}_{jpz}\tilde{\lambda}^*_z\ \forall j \in \delta^-(i_p)$ where $\tilde{d}_{jpz}$ is a binary parameter that takes the value 1 if path z contains the arc (j, i_p) and 0 otherwise. $\theta^*_{ji_p}$ is equal to 1 if the room is assigned to patient p at time period $\tilde{t}_p$ and is assigned to a patient associated with node $j = (j_1, j_2)$ at time period j_2, immediately

preceding patient p. From (2b) and (6e), we get:

$$\sum_{j \in \delta^-(i_p)} \theta^*_{ji_p} = 1 \quad \forall p \in P. \tag{13}$$

Let $\tilde{X}^*_p = \{j \in \delta^-(i_p) | \theta^*_{ji_p} > 0\}$ and $\tilde{p} \in P$ be such that $|\tilde{X}^*_{\tilde{p}}| \geq |\tilde{X}^*_p| \ \forall p \in P$. The following proposition shows that if $|\tilde{U}^*_{\tilde{p}}| = 1$ and $\tilde{\boldsymbol{\lambda}}^*$ is fractional, then $|\tilde{X}^*_{\tilde{p}}| \geq 2$. Analogous proposition also holds for each $l \in L$ with fractional $\hat{\boldsymbol{\lambda}}^{l^*}$ and $|\hat{U}^{\hat{l}^*}_{\hat{p}}| = 1$.

Proposition 5. *If* $|\tilde{U}^*_{\tilde{p}}| = 1$ *and* $\tilde{\boldsymbol{\lambda}}^*$ *is fractional, then* $|\tilde{X}^*_{\tilde{p}}| \geq 2$.

Proof. Let $\tilde{t}_0 = 0$ and assume that $|\tilde{X}^*_{\tilde{p}}| \leq 1$. From this assumption and (13), we infer that $|\tilde{X}^*_p| = 1 \ \forall p \in P$. This implies that for each $p \in P$, $\exists (p_2, \tilde{t}_{p_2}) \in \delta^-(i_p)$ such that the flow through the arc $\{(p_2, \tilde{t}_{p_2}), (p, \tilde{t}_p)\}$ is equal to 1. This, along with $|\tilde{U}^*_{\tilde{p}}| = 1$, shows that for each $p \in P$, $\tilde{\boldsymbol{\lambda}}^*$ leads to the total flow of 1 into nodes i_p and this flow is coming from exactly one node in $\delta^-(i_p)$. This can only occur if $\tilde{\boldsymbol{\lambda}}^*$ is integral. □

Let $\{j_1, j_2, \ldots, j_{|\delta^-(\tilde{p})|}\}$ be such that $\theta^*_{j_1 i_{\tilde{p}}} \geq \theta^*_{j_2 i_{\tilde{p}}} \geq \ldots, \geq \theta^*_{j_{|\delta^-(\tilde{p})|} i_{\tilde{p}}}$ and define $\mathcal{A}_1 = \{(j_q, i_{\tilde{p}}) | q \mod 2 = 0\}$ and $\mathcal{A}_2 = \{(j_q, i_{\tilde{p}}) | q \mod 2 = 1\}$. On the left branch, we restrict the flow through the arc set $\mathcal{A}_1$ to zero while on the right branch, we restrict the flow through the arc set $\mathcal{A}_2$ equal to zero. As $|\tilde{U}^*_{\tilde{p}}| = 1$ and $\tilde{\boldsymbol{\lambda}}^*$ is fractional, from Proposition 5 get $|\tilde{X}^*_{\tilde{p}}| \geq 2$. This implies that $\mathcal{A}_1, \mathcal{A}_2 \neq \emptyset$. As the total flow through these arc sets in the fractional solution is non-zero, this branching scheme excludes the fractional solution $\tilde{\boldsymbol{\lambda}}^*$ and $\hat{\boldsymbol{\lambda}}^{l^*} \forall l \in L$. Additionally, (2b) ensure that this branching scheme is valid and does not exclude any integer feasible solutions. This branching rule can be easily implemented within the `B&P` algorithm by removing the arcs in $\mathcal{A}_1, \mathcal{A}_2$ from the pricing subproblems of the respective children nodes. Lastly, if $\tilde{\boldsymbol{\lambda}}^*$ is integral, then there exists a $\hat{l}$ such that $\hat{\boldsymbol{\lambda}}^{\hat{l}^*}$ is fractional and the arc sets $\mathcal{A}_1, \mathcal{A}_2$ can be constructed in a similar manner for some $\hat{p} \in P_{\hat{l}}$. Thus, `Branching Rule 2` always succeeds in excluding the fractional solution when $|\tilde{U}^*_{\tilde{p}}| = |\hat{U}^{\hat{l}^*}_{\hat{p}}| = 1$. This ensures that the combined implementation of `Branching Rule 1` and `Branching Rule 2` makes the proposed `B&P` algorithm an exact approach for solving `PATSP`.

4.4 Upper-Bounding Heuristic

In this section, we present a heuristic to generate feasible solutions for `PATSP` based on the fractional solution of $\texttt{RMP}^k$. These feasible solutions provide upper bounds on the optimal objective function values of `PATSP`, which can potentially facilitate early termination of `B&P`. Algorithm 1 provides the

pseudocode of the proposed upper-bounding heuristic. Let $\tilde{\boldsymbol{\lambda}}^*$ and $\hat{\boldsymbol{\lambda}}^{l^*}$ $\forall l \in L$ constitute a fractional solution of $\texttt{RMP}^k$, $\tilde{P}_z$ be the set of patients visited by room path z, and $\hat{P}_z^l$ be the set of patients visited by path z of test l. Let $\bar{\Omega}^h = \{z \in \bar{\Omega}^k \mid \tilde{\lambda}_z^* = 1, |\tilde{P}_z| \geq 2\}$ and $\bar{\Pi}_l^h = \emptyset$ $\forall l \in L$. In Step 1, identify $\tilde{z} = \arg\max\{\tilde{\lambda}_z^* \mid z \in \bar{\Omega}^k \setminus \bar{\Omega}^h, |\tilde{P}_z| \geq 2, \tilde{P}_z \cap \bigcup_{z_1 \in \bar{\Omega}^h} \tilde{P}_{z_1} = \emptyset\}$. If such a $\tilde{z}$ is found, update $\bar{\Omega}^h = \bar{\Omega}^h \cup \tilde{z}$. After this update, if $\bigcup_{z \in \bar{\Omega}^h} \tilde{P}_z \neq P$, return to Step 1. However, if no such $\tilde{z}$ exists or if $\bigcup_{z \in \bar{\Omega}^h} \tilde{P}_z = P$, Step 1 is not repeated. If $\bigcup_{z \in \bar{\Omega}^h} \tilde{P}_z = P$, it indicates that all required room paths for a feasible solution have been identified, and the heuristic proceeds to Step 3. Otherwise, the heuristic moves to Step 2.

In Step 2, the problem $\widetilde{\texttt{SP}}$ is solved for the patients $P \setminus \bigcup_{z \in \bar{\Omega}^h} \tilde{P}_z$, with arc costs $\tilde{h}_{ij} = \tilde{c}_{ij}$ $\forall (i,j) \in \tilde{E}$, where

$$\tilde{c}_{ij} = \begin{cases} j_2 - (i_2 + \tilde{s}_{p_{i_1}}) & \text{if } i_1 \neq 0, \\ j_2 - 1 & \text{if } i_1 = 0. \end{cases} \tag{14}$$

$\tilde{c}_{ij}$ represents the idle time of a room when assigned to patient i_1 starting at period i_2 and subsequently assigned to patient j_1 starting at period j_2. The optimal solution of $\widetilde{\texttt{SP}}$ yields a room path covering a subset of patients in $P \setminus \bigcup_{z \in \bar{\Omega}^h} \tilde{P}_z$. The set $\bar{\Omega}^h$ is updated to include this path. If $\bigcup_{z \in \bar{\Omega}^h} \tilde{P}_z = P$, the heuristic moves to Step 3, else Step 2 is repeated.

In Step 3, MILP (15) is solved to determine the start time of each test of each patient. In (15), ζ_{pt}^l is binary variable that takes the value 1 if test l is started for patient p in time period t while η is the maximum number of patients with same test active in a time period.

$$\min \ \eta \tag{15a}$$

$$\text{s.t.} \sum_{t \in \hat{T}_{pl}} \zeta_{pt}^l = 1 \qquad \forall l \in L_p, \forall p \in P \tag{15b}$$

$$\sum_{l \in L_p} \sum_{\tau = \max(1, t - \hat{s}_{pl} + 1)}^{\min(t, \hat{T}_{pl}^f)} \zeta_{pt}^l \leq 1 \qquad \forall t = 1, \ldots, T - t_p^* + 1, \forall p \in P \tag{15c}$$

$$\zeta_{pt}^l \leq \sum_{z \in \bar{\Omega}^h} \sum_{\tau_1 = 1}^{\min(t, \tilde{T}_p^f)} \sum_{\tau_2 = \max(\tau_1 + \tilde{s}_p, t + \hat{s}_{pl})}^{T+1} \tilde{b}_{p\tau_1\tau_2 z} \quad \forall t \in \hat{T}_{pl}, \forall l \in L_p, \forall p \in P \tag{15d}$$

$$\eta \geq \sum_{p \in P_l} \sum_{\tau = \max(1, t - \hat{s}_{pl} + 1)}^{\min(t, \hat{T}_{pl}^f)} \zeta_{p\tau}^l \qquad \forall t \in \hat{T}_{pl}, \forall l \in L \tag{15e}$$

$$\zeta_{pt}^l \in \{0, 1\} \qquad \forall l \in L, \forall p \in P_l, \forall t \in \hat{T}_{pl} \tag{15f}$$

Similar to (1c) and (1d), Constraints (15b) and (15c) ensure that each test $l \in L_p$ is performed

for patient p, and all tests $l \in L_p$ for each patient $p \in P$ are conducted sequentially, respectively. Additionally, analogous to (1e), Constraint (15d) guarantees that every test $l \in L_p$ required by patient p is performed after the patient is assigned a room and before that room is reassigned to another patient. The objective function (15a), in conjunction with constraint (15e), minimizes the maximum number of patients undergoing the same test within a single time period.

Let $\zeta^{l^*}_{pt},\ \forall l \in L, \forall p \in P_l, \forall t \in \hat{T}_{pl}$ comprise the optimal solution of (15) and $\hat{V}^h_l := \{(p,t) \mid \zeta^{l^*}_{pt} = 1,\ p \in P_l,\ t \in \hat{T}_{pl}\}\ \forall l \in L$. For each $l \in L$, in Step 4, $\widehat{\texttt{SP}}^l$ is solved over the node set $\{(p,t) \in \hat{V}^h_l \mid p \notin \bigcup_{z \in \bar{\Pi}^h_l} \hat{P}^l_z\}$ and $\hat{h}^l_{ij} = \hat{c}^l_{ij}$ where

$$\hat{c}^l_{ij} = \begin{cases} j_2 - (i_2 + \hat{s}_{i_1 l}) & \text{if } i_1 \neq 0, \\ j_2 - 1 & \text{if } i_1 = 0. \end{cases} \tag{16}$$

$\hat{c}^l_{ij}$ denotes the idle time for a test l nurse when assigned to patient i_1 beginning at period i_2 and successively reassigned to patient j_1 starting at period j_2. The optimal solution of $\widehat{\texttt{SP}}^l$ generates a test l path that covers a subset of patients in $P_l \setminus \bigcup_{z \in \bar{\Pi}^h_l} \hat{P}^l_z$. The set $\bar{\Pi}^h_l$ is then updated to include this newly identified path. Step 4 is repeated till $\bigcup_{z \in \bar{\Pi}^h_l} \hat{P}^l_z = P_l\ \forall l \in L$. After Step 4, the heuristic concludes with a feasible solution represented by the paths $\bar{\Omega}^h$ and $\bar{\Pi}^h_l\ \forall l \in L$.

This heuristic aims to generate high quality feasible solution for PATSP. To achieve this, Step 1 focuses on considering only room paths where $|\tilde{P}_z| \geq 2$, and in Steps 2 and 4, it employs the objective coefficients $\tilde{c}_{ij}$ and $\hat{c}^l_{ij}$ to ensure that each path in $\bar{\Omega}^h$ and $\bar{\Pi}^h_l\ \forall l \in L$ accommodates as many patients as possible. Additionally, (15) minimizes the maximum number of patients undergoing the same test within a single time period. This helps in distributing test start times evenly across the day, which may reduce the number of nurse paths needed to cover all patients.

4.5 Lower Bounds

In this section, we establish lower bounds on the number of room paths and test paths necessary to serve all patients.

Proposition 6. *The number of rooms required to serve all the patients in P must be at-least*

$$\max\left\{\left\lceil \frac{\sum_{p \in P} \tilde{s}_p}{T} \right\rceil, \left\lceil \frac{|P|}{|\dot{P}^r|} \right\rceil\right\},$$

where $\dot{P}^r = \arg\max\left\{|\dot{P}| \Big| \sum_{p \in \dot{P}} \tilde{s}_p \leq T, \dot{P} \subseteq P\right\}$.

Algorithm 1: Upper-Bounding Heuristic

Input: Fractional solution of $\texttt{RMP}^k$ ($\tilde{\boldsymbol{\lambda}}^*$ and $\hat{\boldsymbol{\lambda}}^{l^*} \forall l \in L$) .
Output: $\bar{\Omega}^h$ and $\bar{\Pi}_l^h \; \forall l \in L$.
Initialize $\bar{\Omega}^h = \{z \in \bar{\Omega}^k \mid \tilde{\lambda}_z^* = 1, |\tilde{P}_z| \geq 2\}$ and $\bar{\Pi}_l^h = \emptyset, \forall l \in L$;
`// Step 1`
while $\bigcup_{z \in \bar{\Omega}^h} \tilde{P}_z \neq P$ **do**
  Find $\tilde{z} = \arg\max\{\tilde{\lambda}_z^* \mid z \in \bar{\Omega}^k \setminus \bar{\Omega}^h, |\tilde{P}_z| \geq 2, \tilde{P}_z \cap \bigcup_{z_1 \in \bar{\Omega}^h} \tilde{P}_{z_1} = \emptyset\}$;
  if $\tilde{z}$ *exists* **then**
    Update $\bar{\Omega}^h := \bar{\Omega}^h \cup \{\tilde{z}\}$;
  else
    break;
`// Step 2`
while $\bigcup_{z \in \bar{\Omega}^h} \tilde{P}_z \neq P$ **do**
  Obtain room path $\tilde{z}$ by solving $\widetilde{\texttt{SP}}$ for the patient set $P \setminus \bigcup_{z \in \bar{\Omega}^h} \tilde{P}_z$ and arc cost $\tilde{h}_{ij} = \tilde{c}_{ij}, \forall (i,j) \in \tilde{E}$;
  Update $\bar{\Omega}^h := \bar{\Omega}^h \cup \{\tilde{z}\}$;
`// Step 3`
Solve (15) and extract $\hat{V}_l^h \; \forall l \in L$;
`// Step 4`
for $l \in L$ **do**
  while $\bigcup_{z \in \bar{\Pi}_l^h} \hat{P}_z^l \neq P_l$ **do**
    Obtain test path $\hat{z}$ by solving $\widehat{\texttt{SP}}$ for the patient set $P_l \setminus \bigcup_{z \in \bar{\Pi}_l^h} \hat{P}_z^l$, the node set $\{(p,t) \in \hat{V}_l^h \mid p \notin \bigcup_{z \in \bar{\Pi}_l^h} \hat{P}_z^l\}$, and arc cost $\hat{h}_{ij}^l = \hat{c}_{ij}^l, \forall (i,j) \in \hat{E}_l$;
    Update $\bar{\Pi}_l^h := \bar{\Pi}_l^h \cup \{\hat{z}\}$;
return $\bar{\Omega}^h$ and $\bar{\Pi}_l^h \; \forall l \in L$;

Proof. We show that both terms in the max operator are lower bounds on the number of rooms required to serve all patients on P.

i. $\left\lceil \frac{\sum_{p\in P} \tilde{s}_p}{T} \right\rceil$: $\sum_{p\in P} \tilde{s}_p$ denotes the total number of time periods needed to serve all patients in P, while T represents the maximum operational time of a room. Therefore, the expression $\left\lceil \frac{\sum_{p\in P} \tilde{s}_p}{T} \right\rceil$ calculates the minimum number of rooms required to serve all patients in P if each room is utilized to its full capacity. Thus, it provides a valid lower bound on the number of rooms necessary to accommodate all patients in P.

ii. $\left\lceil \frac{|P|}{|\dot{P}^r|} \right\rceil$: $|\dot{P}^r|$ represents the maximum number of patients that can be scheduled in a single room. To determine $\dot{P}^r$, we sort patients in non-decreasing order of their total test times, $\tilde{s}_p$; let the sorted order of patients be $p_1, p_2, \ldots, p_{|P|}$. Then, $\dot{P}^r = \{p_j \mid j = 1, 2, \ldots, \bar{k}\}$, where $\bar{k} = \max\{k \mid \sum_{j=1}^{k} \tilde{s}_{p_j} \leq T\}$. Since a maximum of $|\dot{P}^r|$ patients can fit in one room, at least $\left\lceil \frac{|P|}{|\dot{P}^r|} \right\rceil$ rooms are required to accommodate all patients in P.

□

The following corollary gives analogous lower bounds on the number of test l nurses required to serve all patients in $P_l \ \forall l \in L$.

Corollary 1. *The number of nurses required to perform test l on all patients in P_l must be at least*

$$\max\left\{\left\lceil \frac{\sum_{p\in P_l} \hat{s}_{pl}}{T} \right\rceil, \left\lceil \frac{|P|}{|\dot{P}^l|} \right\rceil\right\},$$

where $\dot{P}^l = \arg\max\left\{|\dot{P}| \,\middle|\, \sum_{p\in\dot{P}} \hat{s}_{pl} \leq T, \dot{P} \subseteq P_l\right\}$.

Let $\widetilde{LB}_r = \max\left\{\left\lceil \frac{\sum_{p\in P} \tilde{s}_p}{T} \right\rceil, \left\lceil \frac{|P|}{|\dot{P}^r|} \right\rceil\right\}$ be the room lower bound as described in Proposition 6, and $\widehat{LB}_l = \max\left\{\left\lceil \frac{\sum_{p\in P_l} \hat{s}_{pl}}{T} \right\rceil, \left\lceil \frac{|P|}{|\dot{P}^l|} \right\rceil\right\}, \ \forall l \in L$ be the nurse lower bounds as stated in Corollary 1. We incorporate these lower bounds by adding the following constraints to $\texttt{RMP}^k$:

$$[\pi^5] \quad \sum_{z\in\bar{\Omega}^k} \tilde{\lambda}_z \geq \widetilde{LB}_r \tag{17a}$$

$$[\pi_l^6] \quad \sum_{z\in\bar{\Pi}_l} \hat{\lambda}_z \geq \widehat{LB}_l \qquad \forall l \in L \tag{17b}$$

with dual variables stated in square brackets before each constraint. Following the addition of these constraints, a new room column is added to $\texttt{RMP}^k$ if $\tilde{\eta} < \pi^5 - 1$ while a new test l column is added to $\texttt{RMP}^k$ if $\hat{\eta}^l < \pi_l^6 - 1$.

4.6 Symmetry Breaking Constraints

In PAT clinic, patients can be divided into groups such that all patients within a group require the same set of tests. Let $K = \{1, 2, \ldots, |K|\}$ represent the index set of patient groups, P_k represent the index set of patients in group $k \in K$. Without the loss of generality, assume that $P_k = \{|P_{k-1}|+1, |P_{k-1}|+2, \ldots, |P_{k-1}|+|P_k|\}\ \forall k \in K, |P_0| = 0$. Let p_k^F be the index of first patient in group k, and p_k^L be the index of last patient in group k.

In any feasible solution of PATSP, swapping the schedules of any two patients within the same group results in a new solution with an identical objective function value. This indicates presence of symmetry in the formulation $\mathcal{P}$. In order address this symmetry, we add the following set of constraints to $\texttt{RMP}^k$:

$$[\pi_{pt}^7] \quad \sum_{\tau=1}^{t} \sum_{z \in \bar{\Omega}^k} \tilde{a}_{p\tau z} \tilde{\lambda}_z \geq \sum_{\tau=1}^{t} \sum_{z \in \bar{\Omega}^k} \tilde{a}_{(p+1)\tau z} \tilde{\lambda}_z \quad \forall p \in P_k \setminus p_k^L, \forall k \in K, \forall t \in \tilde{T}_p, \tag{18}$$

with the dual variables stated in square brackets before the constraints set. These constraints ensure the patient with higher index in a group are not allocated a room before the patient with lower index. Addition of these constraints to $\texttt{RMP}^k$ results in a change in the objective function coefficient of y_{ij} in $\widetilde{\texttt{SP}}$ from $\tilde{h}_{ij}$ to $\tilde{h}_{ij} + \Delta\tilde{h}_{ij}$ where

$$\Delta\tilde{h}_{ij} = \begin{cases} 0 & \text{if } j_1 = \texttt{o}', \\ -\sum_{t=j_2}^{\tilde{T}_{j_1}^f} \pi_{j_1 t}^7 & \text{if } j_1 \in \bigcup_{k \in K} \{p_k^F\}, \\ \sum_{t=j_2}^{\tilde{T}_{j_1}^f} \pi_{(j_1-1)t}^7 & \text{if } j_1 \in \bigcup_{k \in K} \{p_k^L\}, \\ \sum_{t=j_2}^{\tilde{T}_{j_1}^f} (\pi_{(j_1-1)t}^7 - \pi_{j_1 t}^7) & \text{otherwise.} \end{cases} \tag{19}$$

5 Implementation

Let $\bar{\Omega}^0$ and $\bar{\Pi}_l^0\ \forall l \in L$ represent the initial set of columns used to construct $\texttt{RMP}^0$ at the root node of B&P. The method employed to generate $\bar{\Omega}^0$ and $\bar{\Pi}_l^0\ \forall l \in L$ is outlined in Algorithm 2. These initial sets of columns are designed to ensure that each patient is assigned a dedicated room, each test required by a patient is assigned a dedicated nurse, and every patient is allocated a room in period 1.

Let $\mathcal{N}$ represent the set of nodes left to be processed within B&P and LB^*, UB^* denote the best known lower bound and upper bound, respectively. *Step 1* of B&P is an initialization step where in *Step 1.1* the root node problem n_0 is constructed which include $\texttt{RMP}^0$ along with the lower bounding constraints (17) and the symmetry breaking constraints (18) while in *Step 1.2*, $\mathcal{N}$ is set equal to $\{n_0\}$.

Algorithm 2: Generating Initial Columns

Input: Instance data
Output: $\bar{\Omega}^0$ and $\bar{\Pi}_l^0$, $\forall l \in L$
Initialize $\bar{\Omega}^0 = \emptyset$ and $\bar{\Pi}_l^0 = \emptyset$, $\forall l \in L$;
for $p \in P$ **do**
  Construct $\tilde{z} = \{(0,0),(p,1),(0,T+1)\}$;
  Update $\bar{\Omega}^0 := \bar{\Omega}^0 \cup \{\tilde{z}\}$;
  Initialize $\bar{L}_p = \emptyset$;
  for $l \in L_p$ **do**
    Construct $\hat{z} = \{(0,0),(p,\sum_{\hat{l} \in \bar{L}p} \hat{s}_{p\hat{l}} + 1),(0,T+1)\}$;
    Update $\bar{\Pi}_l^0 := \bar{\Pi}_l^0 \cup \{\hat{z}\}$;
    Update $\bar{L}_p := \bar{L}_p \cup \{l\}$;
return $\bar{\Omega}^0$ and $\bar{\Pi}_l^0$, $\forall l \in L$;

In *Step 2*, if $\mathcal{N} = \emptyset$ or if $\frac{UB^* - LB^*}{UB^*} \leq \epsilon$, then the algorithm terminates else it proceeds to *Step 3*. *Step 3* is the node selection step where in *Step 3.1*, a node $n_k \in \mathcal{N}$ is chosen using the *breadth-first search* strategy (Morrison et al. 2016). In *Step 3.2*, $\mathcal{N}$ is updated to $\mathcal{N} \setminus n_k$ while in *Step 3.3*, Phase-1 of the Two-Phase Simplex Method (Bertsimas and Tsitsiklis 1997) is implemented to investigate the feasibility of the Linear Program (LP) at the selected node. If the LP is infeasible, algorithm goes back to *Step 2*, else it proceeds to *Step 4*.

Step 4 involves executing the Column Generation algorithm to solve the LP at node n_k, yielding the optimal LP solution with an objective function value LB^{n_k}. Additionally, the upper-bounding heuristic described in Algorithm 1 is executed after each Column Generation iteration to search for a better-quality feasible solution to `PATSP` with a lower UB^*. If such a solution is found, UB^* is updated. If the LP solution obtained from Column Generation in *Step 4* is integral, UB^* is updated if $LB^{n_k} < UB^*$, and the algorithm returns to *Step 2*. Similarly, if $LB^{n_k} > UB^*$, the algorithm also returns to *Step 2*. However, if the LP solution is not integral and $LB^{n_k} \leq UB^*$, the algorithm proceeds to *Step 5*, the branching step. In *Step 5.1*, the branching rules outlined in Section 4.3 is applied to generate the children nodes n_{k_1} and n_{k_2} from n_k. In *Step 5.2*, $\overline{LB}^{n_{k_1}}$ and $\overline{LB}^{n_{k_2}}$ are set to LB^{n_k}, and n_{k_1} and n_{k_2} are added to $\mathcal{N}$. In *Step 5.3*, LB^* is updated to $\min\{\overline{LB}^n | n \in \mathcal{N}\}$, and the algorithm returns to *Step 2*.

6 Computational Experiments

The computational experiments are based on data from Agnihothri et al. (2024), which involves problem instances with four tests ($|L| = 4$) and four patient classes, each requiring a distinct combination of tests. Table 3 outlines the test combinations for each patient class and the percentage of the total

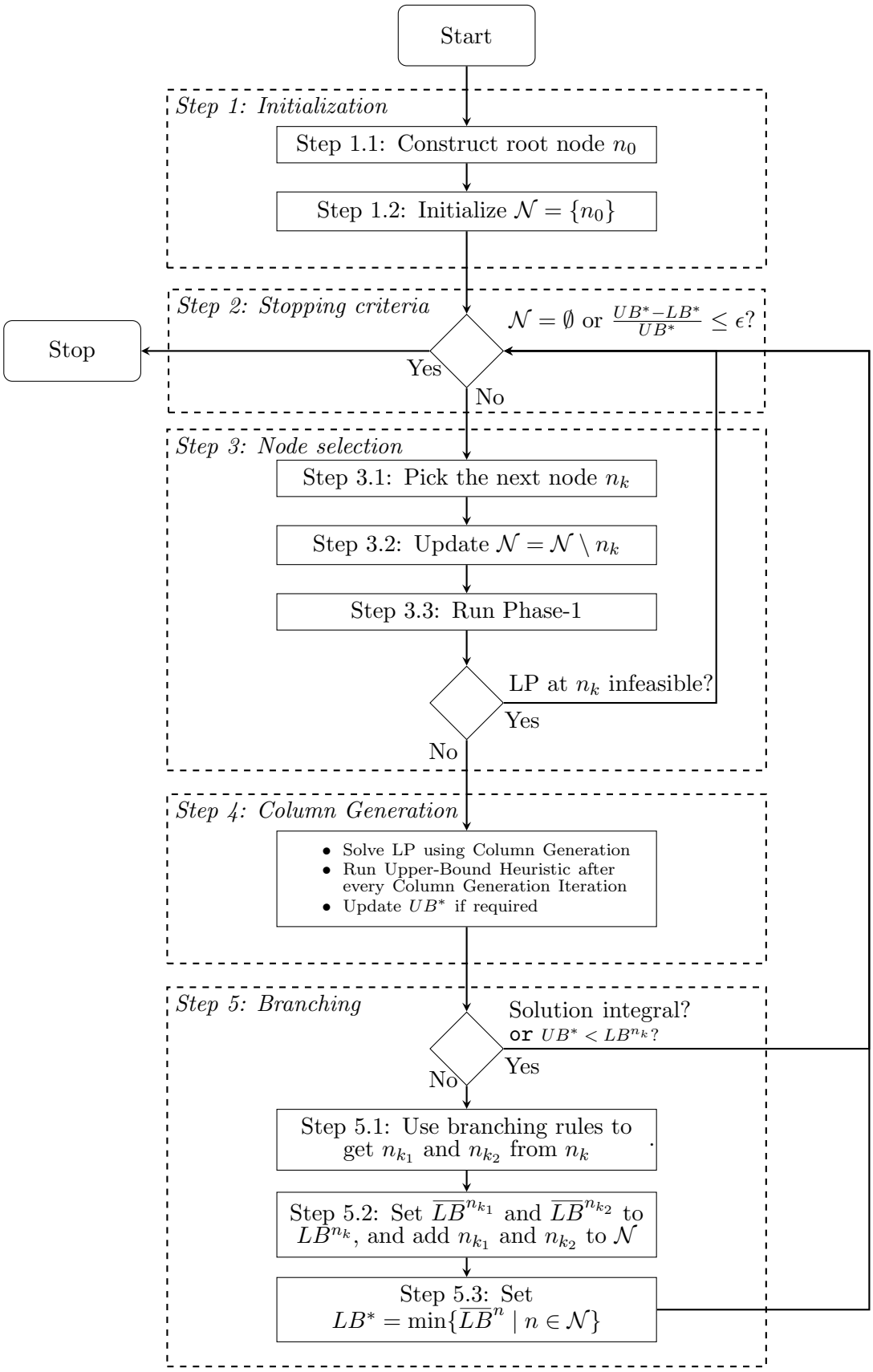


Figure 3: B&P algorithm for PATSP.

patient population represented by each class. Furthermore, Table 4 provides the mean (μ), variance (σ^2), and distributions of test durations.

Patient class	L_p	Proportion (%)
C_1	{1,2}	31%
C_2	{1,2,3}	14%
C_3	{1,2,4}	16%
C_4	{1,2,3,4}	39%

Table 3: Data for patient classes.

Test	μ	σ^2	Distribution
1	27.79	105.27	Gamma (6.73, 4.14)
2	8.83	26.01	Lognormal (2.02, 0.56)
3	5.98	9.67	Lognormal (1.82, 0.45)
4	7.00	6.30	Lognormal (1.61, 0.56)

Table 4: Test duration distributions.

Our preliminary experiments indicate that instances with higher total service times are more challenging to solve. To generate such instances, we introduce the parameter ϕ and create random problem instances based on the test duration distributions stated in Table 4 with mean equal to $\phi\mu$. We consider three values of $|P|= 10, 15, 20$, two values of $T = 16, 32$ and two values of $\phi = 1, 2$. Dividing the planning horizon into 15-minute intervals, these values of T correspond to planning horizon durations of 240 minutes and 480 minutes, respectively. We conduct a full factorial design over the values of $|P|$, T and ϕ, generating 5 random instances for each combination. This results in a total of $3 \times 2 \times 2 \times 5 = 60$ problem instances, which we solve using an Intel Xeon Gold 6132 CPU with 64 GB RAM, Python, and GUROBI 11.0.

6.1 Comparison with the MILP Formulation of PATSP

Table 5 and 6 compare the computational performance of B&P with VIs (9)-(10) ($\text{B\&P}_{\text{VI}}$) and B&P without VIs ($\text{B\&P}_{\text{No\_VI}}$) with the MILP formulation of PATSP, $\mathcal{M}$. This MILP formulation, proposed by Agnihothri et al. (2024), is detailed in Appendix A. In Table 5, Column 1 specifies the Case, Columns 2-4 present the average (maximum) optimality gap (%-Opt-Gap = $\frac{UB^k - LB^k}{UB^k} \times 100$) for each of the three algorithms across the five random instances, and Columns 5-7 show the percentage of instances for each case solved to a 1% optimality gap (%-Inst-Solved). Here, UB^k and LB^k represent the best upper and lower bounds of PATSP obtained by algorithm k. Similarly, Table 6 reports the average (maximum) computational time for each case across all three algorithms. All algorithms are executed with a time limit of 3600 seconds.

Case	%-Opt-Gap			%-Inst-Solved		
$T-\|P\|-\phi$	$\mathcal{M}$	B&P$_{\text{No_VI}}$	B&P$_{\text{VI}}$	$\mathcal{M}$	B&P$_{\text{No_VI}}$	B&P$_{\text{VI}}$
$16-10-1$	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)	100	100	100
$16-15-1$	4.86 (14.29)	3.67 (10.00)	3.43 (10.00)	60	60	60
$16-20-1$	17.47 (26.32)	3.76 (6.67)	3.90 (7.69)	0	40	40
$32-10-1$	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)	100	100	100
$32-15-1$	5.08 (14.29)	0.00 (0.00)	0.00 (0.00)	60	100	100
$32-20-1$	16.42 (27.27)	8.22 (20.00)	10.04 (20.00)	20	40	20
$16-10-2$	2.97 (7.69)	2.76 (7.14)	3.10 (8.33)	60	60	60
$16-15-2$	10.85 (15.00)	8.37 (11.11)	5.95 (11.11)	20	0	40
$16-20-2$	15.06 (19.23)	12.10 (25.81)	12.29 (25.81)	0	20	0
$32-10-2$	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)	100	100	100
$32-15-2$	13.01 (18.18)	9.37 (16.67)	9.37 (16.67)	0	20	20
$32-20-2$	20.93 (26.67)	7.00 (13.33)	12.35 (18.75)	0	40	20

Table 5: Comparison B&P$_{\text{No_VI}}$ and B&P$_{\text{VI}}$ with MILP $\mathcal{M}$ over the optimality gap and percentage of instances solved to a 1% gap.

Case	Computational Time (seconds)		
$T-\|P\|-\phi$	$\mathcal{M}$	B&P$_{\text{No_VI}}$	B&P$_{\text{VI}}$
$16-10-1$	69 (208)	465 (2186)	45 (116)
$16-15-1$	2199 (3600)	1583 (3882)	1569 (3667)
$16-20-1$	3604 (3607)	2494 (3923)	2369 (3916)
$32-10-1$	86 (143)	48 (97)	61 (135)
$32-15-1$	1759 (3600)	159 (344)	414 (1116)
$32-20-1$	2931 (3602)	3175 (4418)	3610 (3994)
$16-10-2$	1459 (3600)	1733 (3846)	1753 (3767)
$16-15-2$	3008 (3605)	3713 (3802)	2553 (3876)
$16-20-2$	3604 (3609)	3060 (4109)	3853 (4086)
$32-10-2$	273 (1283)	137 (448)	78 (179)
$32-15-2$	3601 (3602)	3031 (3753)	3064 (3928)
$32-20-2$	3605 (3608)	3086 (3995)	3687 (4202)

Table 6: Comparison B&P$_{\text{No_VI}}$ and B&P$_{\text{VI}}$ with MILP $\mathcal{M}$ over computational time.

As can be seen in Tables 5 and 6, for all cases with $|P|= 15, 20$, both B&P$_{\text{No_VI}}$ and B&P$_{\text{VI}}$ outperform $\mathcal{M}$ in terms of %-Opt-Gap. Furthermore, these solutions are determined by B&P$_{\text{No_VI}}$ and B&P$_{\text{VI}}$ in similar or shorter computational times. Additionally, for 7 out of 12 cases ($16-20-1, 32-15-1, 32-20-1, 16-15-2, 16-20-2, 32-15-2, 32-20-2$) either B&P$_{\text{No_VI}}$ or B&P$_{\text{VI}}$ solve more instances to a 1% optimality gap than $\mathcal{M}$. For example, in the Case $32-20-2$, B&P$_{\text{No_VI}}$ achieves an average optimality gap of 7% compared to 20.93% for $\mathcal{M}$. Additionally, B&P$_{\text{No_VI}}$ solves 40% of the instances within a 1% optimality gap, whereas $\mathcal{M}$ does not solve any instances within this threshold. The average computational time for B&P$_{\text{No_VI}}$ in this case is 3086 seconds, while $\mathcal{M}$ requires an average of 3605 seconds. Cases with $|P|= 10$ are relatively easy to solve, and all three algorithms exhibit similar computational performance on these cases. Lastly, as can be observed in Tables 5 and 6, cases

with $\phi = 2$ are more difficult to solve as compared to those with $\phi = 1$.

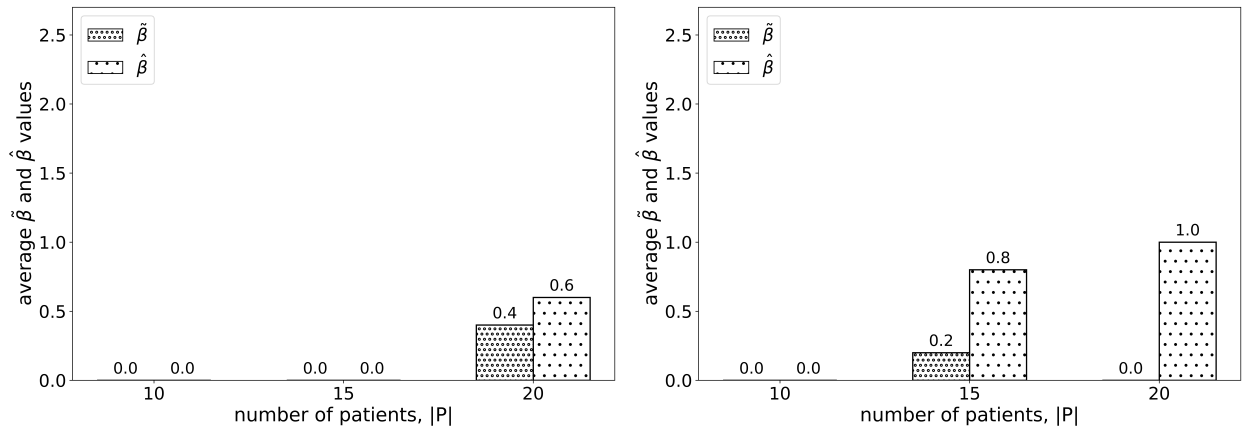
When comparing $\texttt{B\&P}_{\texttt{VI}}$ and $\texttt{B\&P}_{\texttt{No\_VI}}$, we find that the valid inequalities (9)-(10) are particularly effective for instances with $T = 16$. For example, in the Case of $16 - 15 - 2$, $\texttt{B\&P}_{\texttt{VI}}$ achieves the lowest optimality gap and solves more instances within a 1% optimality gap compared to the other two algorithms, while also requiring significantly less computational time. Similarly, for the Case $16 - 10 - 1$, $\texttt{B\&P}_{\texttt{VI}}$ takes an average of 45 seconds per instance, while $\texttt{B\&P}_{\texttt{No\_VI}}$ requires an average of 465 seconds to solve each instance of this case.

6.2 Sensitivity Analysis

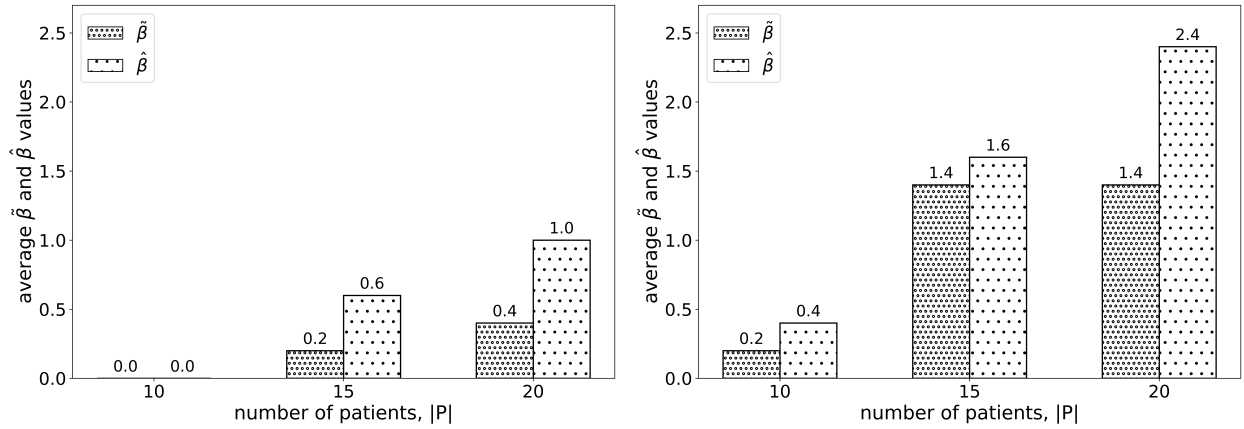
In this section, we examine how the best solution determined by B&P varies with changes in the parameters $|P|$, T, and ϕ. Specifically, we assess the extent to which the number of rooms and nurses used in the best B&P solution exceeds the corresponding lower bounds established in Section 4.5, which assume no synchronization constraints between room and nurse schedules. Any excess over these lower bounds reflects the additional resources required to satisfy synchronization constraints, thereby providing insight into the staffing implications of synchronization constraints between room and nurse schedules to serve a given set of patients.

Let $\widetilde{UB}_r$ denote the number of rooms, and $\widehat{UB}_l$ the number of nurses assigned to test l, $\forall l \in L$, in the best solution determined by $\texttt{B\&P}_{\texttt{No\_VI}}$. We define $\tilde{\beta} = \widetilde{UB}_r - \widetilde{LB}_r$ and $\hat{\beta} = \sum_{l \in L} \left(\widehat{UB}_l - \widehat{LB}_l \right)$ where $\tilde{\beta}$ and $\hat{\beta}$ capture the additional number of rooms and nurses, respectively, used in the best $\texttt{B\&P}_{\texttt{No\_VI}}$ solution compared to the lower bounds established in Section 4.5. Figures 4a and 4b show how the average values of $\tilde{\beta}$ and $\hat{\beta}$ across five random instances vary as $|P|$ increases from 10 to 20, for the case where $T = 32$, $\phi = 1$ and $T = 32$, $\phi = 2$, respectively. Figures 4c and 4d plots the same for $T = 16$, $\phi = 1$ and $T = 16$, $\phi = 2$, respectively. These figures reveal an increasing trend in both $\tilde{\beta}$ and $\hat{\beta}$ as $|P|$ increases, with the effect being more prominent when $T = 16$. This pattern can be explained by the fact that adding more patients introduces additional synchronization constraints between room and nurse schedules. As a result, there are more instances where either a patient is waiting in the room for a nurse, or a nurse is waiting for an ongoing test on a patient to be finished. Such delays increase idle time for both resources, reducing the number of patients each room and nurse can serve within the fixed planning horizon T.

Likewise, for $T = 16$, we see a clear upward trend in both $\tilde{\beta}$ and $\hat{\beta}$ as ϕ increases from 1 to 2. This pattern is also driven by synchronization constraints which may result in a patient waiting for a nurse who is still attending another patient, or a nurse waiting for a patient whose earlier test has not yet



(a) $T = 32$, $\phi = 1$ (b) $T = 32$, $\phi = 2$



(c) $T = 16$, $\phi = 1$ (d) $T = 16$, $\phi = 2$

Figure 4: Average $\tilde{\beta}$ and $\hat{\beta}$ values for $\texttt{B\&P}_{\texttt{No\_VI}}$.

concluded. When test durations are longer, wait times become more pronounced. These higher wait times lead to greater idle time for both rooms and nurses, ultimately lowering the number of patients they can accommodate within the fixed planning horizon T.

Finally, as the planning horizon T is reduced from 32 to 16, we observe an upward trend in both $\tilde{\beta}$ and $\hat{\beta}$. This can again be attributed to the synchronization constraints between room and nurse schedules, which result in idle periods for both resources. These idle times reduce the total time each room and nurse spends actively serving patients. When $T = 32$, the longer availability of rooms and nurses helps mitigate the impact of reduced patient throughput. However, when $T = 16$, the shorter time horizon amplifies the effect, leading to a more pronounced need for additional resources.

6.3 Evaluating the Importance of Synchronization

In this section, we assess the significance of synchronization between room and nurse schedules. To do so, we consider the following three heuristic scheduling approaches inspired by those outlined in White et al. (2011). These heuristics do not fully synchronize room schedules with nurse schedules.

Without the loss of generality, we assume rooms are indexed as $\{1, 2, 3, \dots\}$.

1. Shortest Time First (`STF`): Sort the patients in the non-decreasing order based on their $\tilde{s}_p$ values. Assign the unallocated patient with the smallest index in the sorted list to the room with the lowest index that can accommodate their total test time. The start time period for this patient is set to one period after the end time period of the last patient assigned to that room. Then, determine the test schedules by solving a modified version of MILP (20), where the room schedule variables are fixed based on this `STF` room scheduling paradigm.

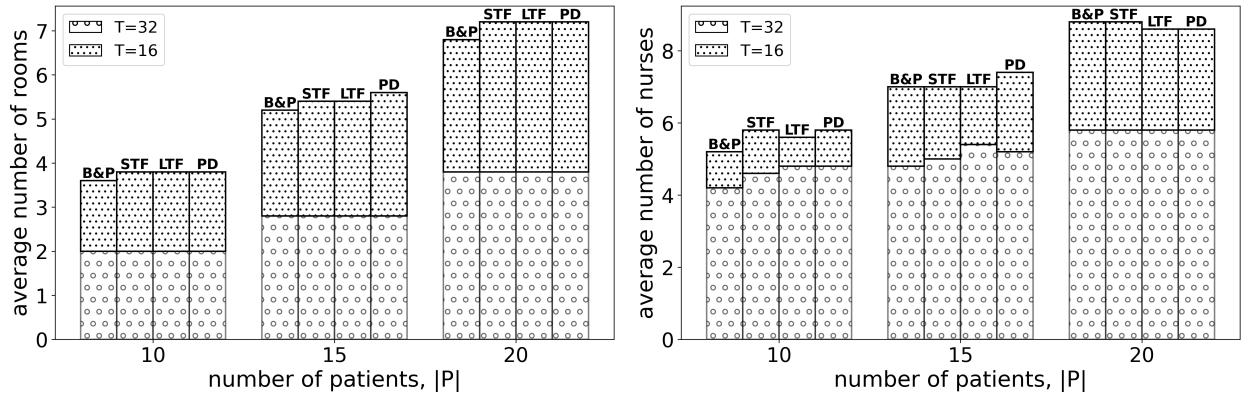
2. Longest Time First (`LTF`): Sort the patients in the non-increasing order based on their $\tilde{s}_p$ values. Assign the unallocated patient with the smallest index in the sorted list to the room with the lowest index that can accommodate their total test time. The start time period for this patient is set to one period after the end time period of the last patient allocated to that room. Then, determine the test schedules by solving a modified version of MILP (20), where the room schedule variables are fixed based on this `LTF` room scheduling paradigm.

3. Plateau Dome (`PD`): Sort the patients such that those with lower $\tilde{s}_p$ values are positioned at the start and end of the sorted list, while those with higher $\tilde{s}_p$ are positioned in the middle. Let $\tilde{P} = \{p_1, p_2, \dots, p_{|P|} \mid \tilde{s}_{p_1} \leq \tilde{s}_{p_2} \leq \dots \leq \tilde{s}_{p_{|P|}}\}$ be the index set of patients sorted in non-decreasing order of their total test time. Let

$$P_{\texttt{PD}} = \left\{ p_1, p_2, \dots, p_{\left\lfloor \frac{|P|}{4} \right\rfloor}, p_{\left\lfloor \frac{|P|}{2} \right\rfloor}, p_{\left\lfloor \frac{|P|}{2} \right\rfloor + 1}, \dots, p_{|P|}, p_{\left\lfloor \frac{|P|}{4} \right\rfloor + 1}, p_{\left\lfloor \frac{|P|}{4} \right\rfloor + 2}, \dots, p_{\left\lfloor \frac{|P|}{2} \right\rfloor - 1} \right\}.$$

   Assign the unallocated patient with the smallest index in $P_{\texttt{PD}}$ to the room with the lowest index that can accommodate their total test time. The start time period for this patient is set to one period after the end time period of the last patient assigned to that room. Then, determine the test schedules by solving a modified version of MILP (20), where the room schedule variables are fixed based on this `PD` room scheduling paradigm.

MILPs in all three of these heuristics are solved for a maximum of 3600s. Let N_r^k and $N_n^{l,k}$ represent the number of rooms and test l nurses utilized in the best solution obtained by algorithm k. The total idle time for rooms and nurses in this solution is given by $\dot{C}^k = \tilde{C}^k + \hat{C}^k$, where $\tilde{C}^k = N_r^k T - \sum_{p \in P} \tilde{s}_p$ represents the total idle time for rooms, and $\hat{C}^k = \sum_{l \in L}(N_n^{l,k} T - \sum_{p \in P_l} \hat{s}_{pl})$ represents the total idle time for nurses. We define $\tilde{\rho}^k = \frac{\dot{C}^k - \dot{C}^{\texttt{B\&P}_{\texttt{No\_VI}}}}{\dot{C}^k} \times 100, \;\; k = \texttt{STF}, \texttt{LTF}, \texttt{PD}$, which represents the percentage reduction in total idle time achieved by $\texttt{B\&P}_{\texttt{No\_VI}}$ compared to heuristic k.



(a) (b)

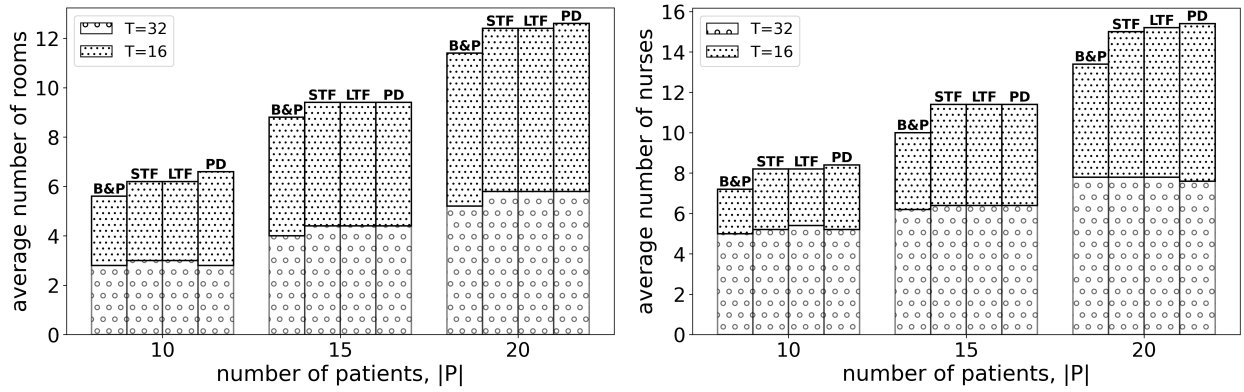
Figure 5: Average number of required rooms and nurses for STF, LTF, PD, and $\texttt{B\&P}_{\texttt{No\_VI}}$ for $\phi = 1$.

For each combination of $|P|$, T and ϕ, Table 7 reports the average (maximum) $\tilde{\rho}^k$ across five random instances for $k = \texttt{STF}, \texttt{LTF}, \texttt{PD}$. Similarly, when $\phi = 1$, Figures 5a and 5b plot the average number of rooms and nurses utilized, respectively, for each combination of $|P|$ and T. Figures 6 plot the same metrics for $\phi = 2$.

$T - \lvert P\rvert$	Heuristic	ϕ	
		1	2
$16 - 10$	STF	17.85 (40.00)	27.63 (52.17)
	LTF	14.85 (25.00)	27.63 (52.17)
	PD	17.85 (40.00)	28.19 (56.34)
$16 - 15$	STF	-4.78 (19.05)	21.27 (47.06)
	LTF	-4.78 (19.05)	21.27 (47.06)
	PD	14.97 (37.21)	21.27 (47.06)
$16 - 20$	STF	5.93 (29.63)	23.79 (62.34)
	LTF	3.48 (17.39)	24.59 (57.97)
	PD	4.10 (20.51)	26.12 (62.34)
$32 - 10$	STF	9.44 (25.00)	10.00 (25.40)
	LTF	14.37 (25.00)	14.33 (25.40)
	PD	14.37 (25.00)	5.08 (25.40)
$32 - 15$	STF	6.04 (30.19)	14.26 (24.24)
	LTF	15.79 (30.19)	15.00 (27.59)
	PD	9.28 (27.12)	15.00 (27.59)
$32 - 20$	STF	0.00 (0.00)	10.56 (41.56)
	LTF	-1.26 (19.51)	10.56 (41.56)
	PD	0.00 (0.00)	7.75 (41.56))

Table 7: Average (maximum) $\tilde{\rho}^k$ for $k = \texttt{STF}, \texttt{LTF}, \texttt{PD}$.

As shown in Table 7, in most cases, $\texttt{B\&P}_{\texttt{No\_VI}}$ yields solutions with significantly lower total idle-time compared to all three heuristics. Additionally, this reduction in idle time due to $\texttt{B\&P}_{\texttt{No\_VI}}$ is even more pronounced in instances where $T = 16$ or $\phi = 2$, i.e., when, on average, fewer surgeries can be scheduled per room or per nurse. This reduction in idle time occurs because, unlike the three heuristics, formulation $\mathcal{P}$ for PATSP explicitly incorporates synchronization between room and nurse



(a) (b)

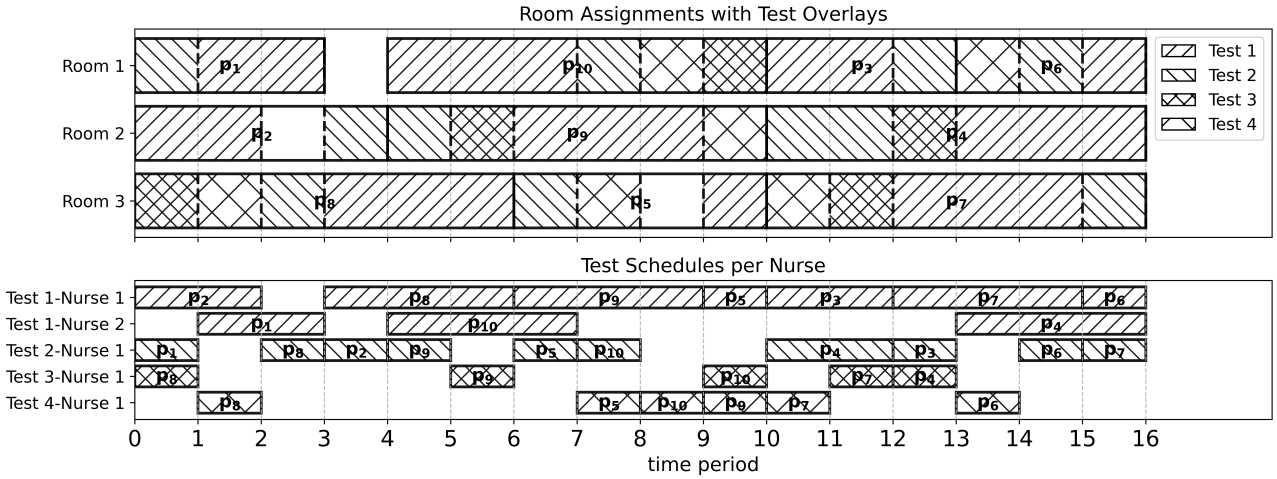
Figure 6: Average number of required rooms and nurses for STF, LTF, PD, and $\texttt{B\&P}_{\texttt{No\_VI}}$ for $\phi = 2$.

schedules. As a result, $\texttt{B\&P}_{\texttt{No\_VI}}$ produces solutions that require fewer rooms and nurses for the same workload compared to the three heuristics, as illustrated in Figures 5, 6. For instance, consider the cases where $|P|= 15$ and $\phi = 2$. As shown in Figure 6, $\texttt{B\&P}_{\texttt{No\_VI}}$ achieves a solution with a lower average number of rooms and nurses utilized compared to all three heuristics, for both $T = 16$ and $T = 32$. This leads to an average total idle-time reduction of approximately 21% for $T = 16$ and around 15% for $T = 32$ (see Table 7).

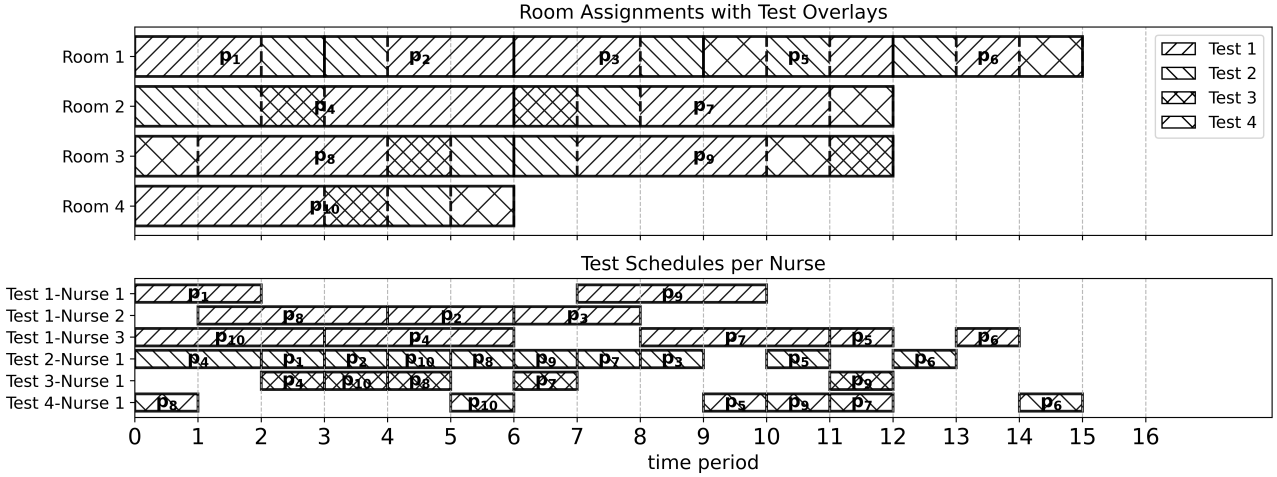
To further highlight the importance of synchronization between room and nurse schedules, we analyze the schedules generated for an instance with $|P|= 10$, $T = 16$, and $\phi = 1$. Figures 7a and 7b display the schedules produced by $\texttt{B\&P}_{\texttt{No\_VI}}$ and STF, respectively, while the schedules for PD and LTF are provided in Appendix B. All three heuristic methods required one additional room compared to $\texttt{B\&P}_{\texttt{No\_VI}}$. Moreover, STF also used one more Test 1 nurse than $\texttt{B\&P}_{\texttt{No\_VI}}$ due to the absence of synchronization between room and nurse schedules. As shown in Figure 7b, under STF, the patients assigned across the 4 rooms during the first 6 time periods collectively need 13 time periods of Test 1 service. This implies that at least $\lceil \frac{13}{6} \rceil = 3$ Test 1 nurses are needed to meet the demand within that window. In contrast, because $\texttt{B\&P}_{\texttt{No\_VI}}$ ensures synchronized scheduling between rooms and nurses, the patients across the same 4 rooms over the first 6 periods require only 9 time periods for Test 1. Thus, only 2 Test 1 nurses are sufficient to serve all patients during that time window.

7 Conclusion

In this paper, we propose a path-based formulation to optimize the scheduling of rooms, patients, and nurses in a PAT clinic, aiming to minimize the total number of rooms and nurses required to serve all patients. We develop a B&P algorithm to solve the path-based model and introduce several techniques



(a) B&P$_{\texttt{No_VI}}$



(b) STF

Figure 7: Room and nurse schedules generated using B&P$_{\texttt{No_VI}}$ and STF for an instance with $|P|= 10$, $T = 16$, and $\phi = 1$; p_k represents patient k.

to enhance its computational efficiency. Our numerical experiments highlight the computational advantages of our modeling and solution approach compared to the MILP formulation presented by Agnihothri et al. (2024) for a similar problem with a different objective function.

Additionally, we demonstrate the significance of incorporating synchronization between room and nurse schedules to achieve efficient resource utilization. The approaches which either completely ignore synchronization constraints—such as in the case of lower bounds on room and nurse requirements discussed in Section 4.5—or only partially account for them, as in the heuristics `STF`, `LTF`, and `PD`, tend to under- or over-estimate the number of rooms and nursing staff needed to serve a given patient set. In contrast, the decision support framework proposed for `PATSP` in this study fully integrates synchronization constraints, allowing decision-makers to determine the appropriate number of rooms and nurse staffing levels, thereby supporting efficient allocation and utilization of resources.

This study has certain limitations. It assumes that test durations are deterministic, whereas in practice they may be stochastic. Additionally, it assumes each nurse performs only one type of test, although there may be instances where a nurse can conduct multiple types of tests. Future research will aim to overcome these limitations by extending the current modeling and solution framework to accommodate: (i) stochastic test durations, and (ii) nurses capable of performing multiple test types. In addition, future research will explore alternative objective functions, such as minimizing the Makespan or the total completion time for all patients.

References

Agius M, Absi N, Feillet D, Garaix T (2022) A branch-and-price algorithm for a routing problem with inbound and outbound requests. *Computers & Operations Research* 146:105896.

Agnihothri S, Cappanera P, Nonato M, Visintin F (2024) Appointment scheduling in surgery pre-admission testing clinics. *Omega* 123:102994.

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

Akbarzadeh B, Maenhout B (2025) A dedicated branch-price-and-cut algorithm for advance patient planning and surgeon scheduling. *European Journal of Operational Research* 322(2):448–466.

Apergi L, Baras J, Golden B, Wood K (2020) An optimization model for multi-appointment scheduling in an outpatient cardiology setting. *Operations Research for Healthcare* 26:Article 100267.

Bargetto R, Garaix T, Xie X (2023) A branch-and-price-and-cut algorithm for operating room scheduling under human resource constraints. *Computers & Operations Research* 152:106136.

Barnhart C, Johnson EL, Nemhauser GL, Savelsbergh MW, Vance PH (1998) Branch-and-price: Column generation for solving huge integer programs. *Operations Research* 46(3):316–329.

Beaudry A, Laporte G, Melo T, Nickel S (2010) Dynamic transportation of patients in hospitals. *OR Spectrum* 32:77–107.

Beliën J, Demeulemeester E (2008) A branch-and-price approach for integrating nurse and surgery scheduling. *European Journal of Operational Research* 189(3):652–668.

Bertsimas D, Tsitsiklis JN (1997) *Introduction to linear optimization*, volume 6 (Athena Scientific Belmont, MA).

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

Cissé M, Yalçındağ S, Kergosien Y, Şahin E, Lenté C, Matta A (2017) Or problems related to home health care: A review of relevant routing and scheduling problems. *Operations Research for Health Care* 13:1–22.

Cordier JP, Riane F (2013) Towards a centralised appointments system to optimise the length of patient stay. *Decision Support Systems* 55(2):629–639.

Costa L, Contardo C, Desaulniers G (2019) Exact branch-price-and-cut algorithms for vehicle routing. *Transportation Science* 53(4):946–985.

Deleplanque S, Labbé M, Ponce D, Puerto J (2020) A branch-price-and-cut procedure for the discrete ordered median problem. *INFORMS Journal on Computing* 32(3):582–599.

Dell'Amico M, Iori M, Martello S, Monaci M (2008) Heuristic and exact algorithms for the identical parallel machine scheduling problem. *INFORMS Journal on Computing* 20(3):333–344.

Di Mascolo M, Martinez C, Espinouse ML (2021) Routing and scheduling in home health care: A literature survey and bibliometric analysis. *Computers & Industrial Engineering* 158:107255.

Drexl M (2012) Synchronization in vehicle routing—a survey of vrps with multiple synchronization constraints. *Transportation Science* 46(3):297–316.

Feillet D, Dejax P, Gendreau M, Gueguen C (2004) An exact algorithm for the elementary shortest path problem with resource constraints: Application to some vehicle routing problems. *Networks: An International Journal* 44(3):216–229.

Fink M, Desaulniers G, Frey M, Kiermaier F, Kolisch R, Soumis F (2019) Column generation for vehicle routing problems with multiple synchronization constraints. *European Journal of Operational Research* 272(2):699–711.

Fischer T, Pfetsch ME (2018) Branch-and-cut for linear programs with overlapping sos1 constraints. *Mathematical Programming Computation* 10:33–68.

Gartner D, Frey M, Kolisch R (2018) Hospital-wide therapist scheduling and routing: Exact and heuristic methods. *IISE Transactions on Healthcare Systems Engineering* 8(4):268–279.

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

Gupta D, Wang WY (2012) *Patient Appointments in Ambulatory Care*, 65–104 (Boston, MA: Springer US).

Hanne T, Melo T, Nickel S (2009) Bringing robustness to patient flow management through optimized patient transports in hospitals. *Interfaces* 39(3):241–255.

Hashemi Doulabi H, Pesant G, Rousseau LM (2020) Vehicle routing problems with synchronized visits and stochastic travel and service times: Applications in healthcare. *Transportation Science* 54(4):1053–1072.

Hashemi Doulabi SH, Rousseau LM, Pesant G (2016) A constraint-programming-based branch-and-price-and-cut approach for operating room planning and scheduling. *INFORMS Journal on Computing* 28(3):432–448.

Jungwirth A, Desaulniers G, Frey M, Kolisch R (2022) Exact branch-price-and-cut for a hospital therapist scheduling problem with flexible service locations and time-dependent location capacity. *INFORMS Journal on Computing* 34(2):1157–1175.

Kramer R, Cordeau JF, Iori M (2019) Rich vehicle routing with auxiliary depots and anticipated deliveries: An application to pharmaceutical distribution. *Transportation Research Part E: Logistics and Transportation Review* 129:162–174.

Lalonde O, Côté JF, Gendron B (2022) A branch-and-price algorithm for the multiple knapsack problem. *INFORMS Journal on Computing* 34(6):3134–3150.

Legrain A, Omer J (2024) A dedicated pricing algorithm to solve a large family of nurse scheduling problems with branch-and-price. *INFORMS Journal on Computing* 36(4):1108–1128.

Li J, Qin H, Baldacci R, Zhu W (2020) Branch-and-price-and-cut for the synchronized vehicle routing problem with split delivery, proportional service time and multiple time windows. *Transportation Research Part E: Logistics and Transportation Review* 140:101955.

Marynissen J, Demeulemeester E (2019) Literature review on multi-appointment scheduling problems in hospitals. *European Journal of Operational Research* 272(2):407–419.

McCarty K, Gallien J, Levi R (2012) Massachusetts general hospital's pre-admission testing area (pata) .

Morrice D, Bard J, Karl K (2020) Designing and scheduling a multi-disciplinary integrated practice unit for patient-centred care. *Health Systems* 9(4):293–316, URL `http://dx.doi.org/10.1080/20476965.2019.1569481`.

Morrison DR, Jacobson SH, Sauppe JJ, Sewell EC (2016) Branch-and-bound algorithms: A survey of recent advances in searching, branching, and pruning. *Discrete Optimization* 19:79–102.

Nemati S, Shylo OV, Prokopyev OA, Schaefer AJ (2016) The surgical patient routing problem: A central planner approach. *INFORMS Journal on Computing* 28(4):657–673.

Oliveira D, Pessoa A (2020) An improved branch-cut-and-price algorithm for parallel machine scheduling problems. *INFORMS Journal on Computing* 32(1):90–100.

Reihaneh M, Ansari S, Farhadi F (2023) Patient appointment scheduling at hemodialysis centers: An exact branch and price approach. *European Journal of Operational Research* 309(1):35–52.

Schmid V, Doerner KF (2014) Examination and operating room scheduling including optimization of intrahospital routing. *Transportation Science* 48(1):59–77.

Soares R, Marques A, Amorim P, Parragh SN (2024) Synchronisation in vehicle routing: classification schema, modelling framework and literature review. *European Journal of Operational Research* 313(3):817–840.

Taccari L (2016) Integer programming formulations for the elementary shortest path problem. *European Journal of Operational Research* 252(1):122–130.

Wang D, Muthuraman K, Morrice D (2019) Coordinated patient appointment scheduling for a multi-station healthcare network. *Operations Research* (67(3)):599–618.

White DL, Froehle CM, Klassen KJ (2011) The effect of integrated scheduling and capacity policies on clinical efficiency. *Production and Operations Management* 20(3):442–455.

Wolsey LA (2020) *Integer programming* (John Wiley & Sons).

Xu X, Li F, Wu T, Huang X, Guan X, Zheng T, Shen L (2025) Location-routing optimization problem of pharmaceutical cold chain logistics with oil-electric mixed fleets under uncertainties. *Computers & Industrial Engineering* 201:110932.

Zhang P, Bard J, Morrice D, Koenig K (2019) Extended open shop scheduling with resource constraints: Appointment scheduling for integrated practice units. *IISE Transactions* 51(10):1037–1060.

Zhang Z, Denton BT, Xie X (2020) Branch and price for chance-constrained bin packing. *INFORMS Journal on Computing* 32(3):547–564.

Appendix A: MILP Formulation for PATSP

In this section, we present a MILP formulation for PATSP as outlined in Agnihothri et al. (2024). Let $\chi_{p_ip_j}$ denote a binary variable that takes the value 1 if patient p_j is assigned to the same room immediately following patient p_i and 0 otherwise. Similarly, let $\gamma^p_{l_il_j}$ be a binary variable equal to 1 if test l_j is conducted on patient p immediately after test l_i and 0 otherwise. Additionally, $\xi^l_{p_ip_j}$ is a binary variable that equals 1 if a nurse performs test l on patient p_j immediately after completing the test on patient p_i and 0 otherwise. $\chi_{\mathtt{o}p_j}$ ($\chi_{p_j\mathtt{o}'}$) is a binary variable that equals 1 if p_j is the first (last) patient in a room's schedule and 0 otherwise. Likewise, $\gamma^p_{\mathtt{o}l_j}$ ($\gamma^p_{l_j\mathtt{o}'}$) is a binary variable that equals 1 if l_j is the first (last) test performed on patient p and 0 otherwise. Lastly, $\xi^l_{\mathtt{o}p_j}$ ($\xi^l_{p_j\mathtt{o}'}$) is a binary variable that equals 1 if p_j is the first (last) patient in the schedule of a nurse assigned to perform test l and 0 otherwise.

Variable v^{in}_p represents the first time period when patient p begins using a room, while variable v^{out}_p denotes the earliest time period in which the same room becomes available for another patient after all tests of patient p have been completed. Variable α^p_l denotes the start time period of test l for patient p. With the above notations, the MILP formulation for PATSP is presented as follows:

$$(\mathcal{M}) \quad \min \sum_{p\in P} \chi_{\mathtt{o}p} + \sum_{l\in L}\sum_{p\in P_l} \xi^l_{\mathtt{o}p} \tag{20a}$$

$$\text{s.t.} \sum_{p_i\in P\cup\{\mathtt{o}\}:p_i\neq p_j} \chi_{p_ip_j} = 1 \qquad \forall p_j \in P \tag{20b}$$

$$\sum_{p_j\in P\cup\{\mathtt{o}'\}:p_j\neq p_i} \chi_{p_ip_j} = 1 \qquad \forall p_i \in P \tag{20c}$$

$$v^{out}_p \geq v^{in}_p + \tilde{s}_p \qquad \forall p \in P \tag{20d}$$

$$v^{out}_p \leq T + 1 \qquad \forall p \in P \tag{20e}$$

$$v^{out}_{p_i} \leq v^{in}_{p_j} + (1-\chi_{p_ip_j})T \qquad \forall p_i, p_j \in P \tag{20f}$$

$$\sum_{l\in L_p} \gamma^p_{\mathtt{o}l} = 1 \qquad \forall p \in P \tag{20g}$$

$$\sum_{l_j\in L_p\cup\{\mathtt{o}'\}} \gamma^p_{l_il_j} = 1 \qquad \forall l_i \in L_p, p \in P \tag{20h}$$

$$\sum_{l_j\in L_p\cup\{\mathtt{o}\}} \gamma^p_{l_jl_i} = 1 \qquad \forall l_i \in L_p, p \in P \tag{20i}$$

$$\alpha^p_{l_i} + \hat{s}_{pl_i} \leq \alpha^p_{l_j} + (1-\gamma^p_{l_il_j})T \qquad \forall l_i, l_j \in L_p, p \in P \tag{20j}$$

$$\alpha^p_{l_i} \geq v^{in}_p + \sum_{l_j\in L_p} \gamma^p_{l_jl_i}\hat{s}_{pl_j} \qquad \forall l_i \in L_p, p \in P \tag{20k}$$

$$\alpha^p_{l_i} + \hat{s}_{pl_i} + \sum_{l_j \in L_p} \gamma^p_{l_i l_j} \hat{s}_{pl_j} \leq \upsilon^{out}_p \quad \forall l_i \in L_p, p \in P \tag{20l}$$

$$\sum_{p_j \in P_l \cup \{\mathtt{o}'\}: p_j \neq p_i} \xi^l_{p_i p_j} = 1 \quad \forall p_i \in P, l \in L_{p_i} \tag{20m}$$

$$\sum_{p_j \in P_l \cup \{\mathtt{o}\}: p_j \neq p_i} \xi^l_{p_j p_i} = 1 \quad \forall p_i \in P, l \in L_{p_i} \tag{20n}$$

$$\alpha^{p_i}_l + \hat{s}_{p_i l} \leq \alpha^{p_j}_l + (1 - \xi^l_{p_i p_j})T \quad \forall l \in L, p_i, p_j \in P_l \tag{20o}$$

$$\sum_{p \in P} \chi_{\mathtt{o}p} \geq \widetilde{LB}_r \tag{20p}$$

$$\sum_{p \in P_l} \xi^l_{\mathtt{o}p} \geq \widehat{LB}_l \quad \forall l \in L \tag{20q}$$

$$\upsilon^{in}_p \leq \upsilon^{in}_{p+1} \quad \forall p \in P_k \setminus p^L_k, k \in K \tag{20r}$$

$$\chi_{p_i p_j} \in \{0,1\} \quad \forall p_i, p_j \in P \cup \{\mathtt{o}, \mathtt{o}'\} : p_i \neq p_j \tag{20s}$$

$$\gamma^p_{l_i l_j} \in \{0,1\} \quad \forall l_i, l_j \in L_p \cup \{\mathtt{o}, \mathtt{o}'\} : l_i \neq l_j, p \in P \tag{20t}$$

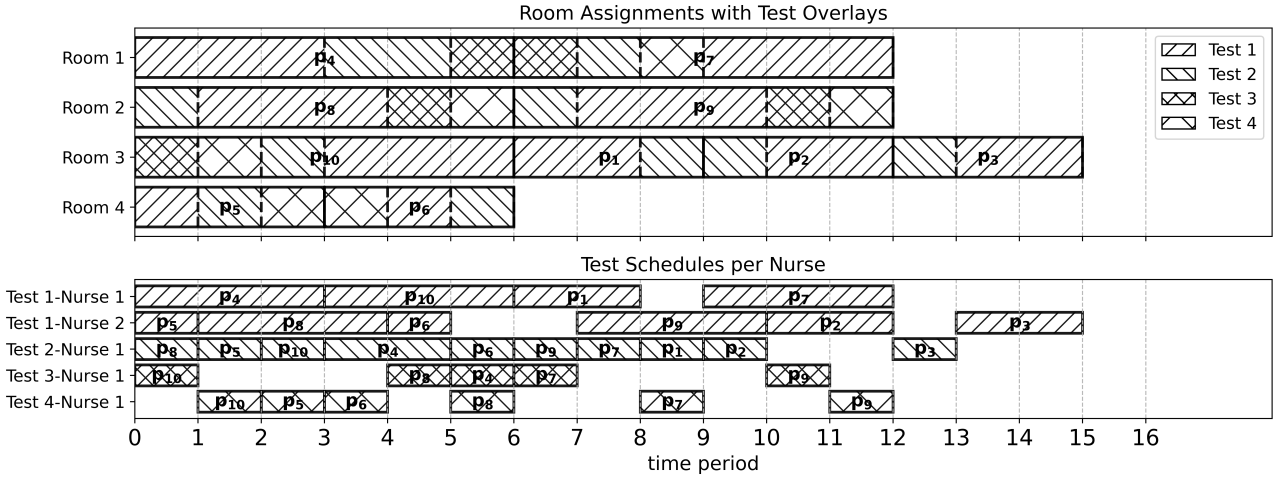
$$\xi^l_{p_i p_j} \in \{0,1\} \quad \forall l \in L_{p_i} \cap L_{p_j}, \forall p_i, p_j \in P_l \cup \{\mathtt{o}, \mathtt{o}'\} : p_i \neq p_j \tag{20u}$$

$$\upsilon^{in}_p \geq 1, \upsilon^{out}_p \geq 2 \quad \forall p \in P \tag{20v}$$

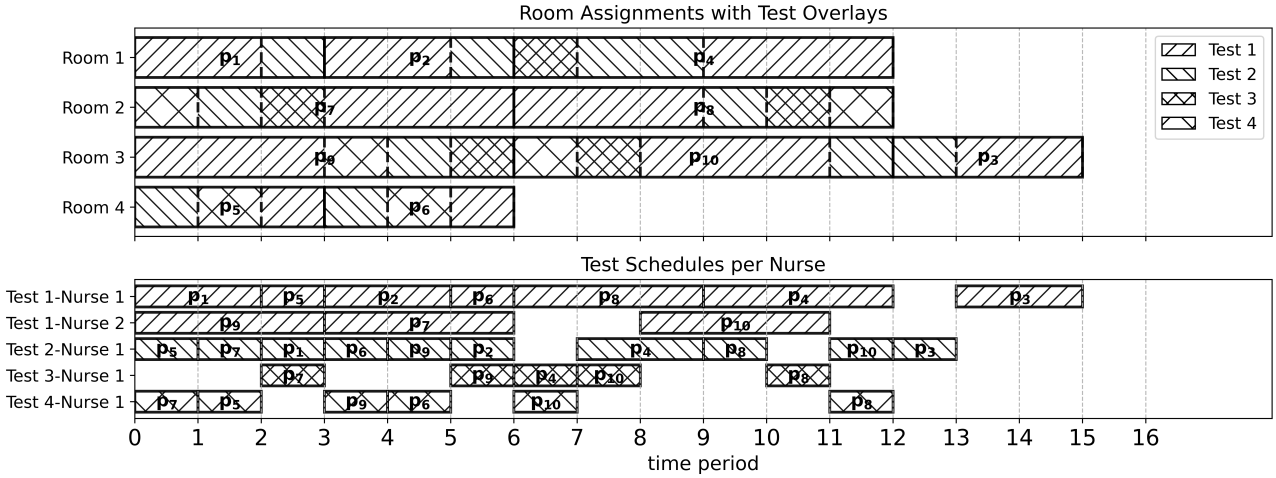
$$\alpha^p_l \geq 1 \quad \forall l \in L_p, p \in P \tag{20w}$$

In (20), the objective function (20a) minimizes the total number of utilized rooms and nurses. Constraints (20b) and (20c) ensure that each patient is assigned exactly one room, and that each room's schedule starts with $\mathtt{o}$ and ends with $\mathtt{o}'$. Constraints (20d) ensure that patient p is assigned a room for at-least $\tilde{s}_p$ time periods while Constraints (20e) ensure that all tests are performed on patients within the planning horizon. Constraints (20f) guarantee that if patient p_j is scheduled to use the same room immediately following patient p_i, the room assignment to p_j occurs only after p_i has vacated the room. Constraints (20g), (20h), and (20i) ensure that each test in L_p is sequentially performed on patient p exactly once. Constraints (20j), (20k), and (20l) guarantee that the scheduled tests for the patients are conducted within the time window during which they occupy a room. Constraints (20m) and (20n) ensure that each patient is assigned exactly one nurse for each of their tests and that each nurse's schedule starts with $\mathtt{o}$ and ends with $\mathtt{o}'$. Constraints (20o) ensure that a nurse cannot begin administering a test to a patient until the test for the immediately preceding patient has been completed. Constraints (20p) and (20q) are lower bounding constraints presented in Section 4.5 while Constraints (20r) are the symmetry breaking constraints discussed in Section 4.6. Constraints (20s) through (20w) define the variables domains.

Appendix B: Schedules for an Instance Obtained Using LTF and PD



(a) LTF



(b) PD

Figure 8: Room and nurse schedules generated using LTF and PD for an instance with $|P|= 10$, $T = 16$, and $\phi = 1$; p_k represents patient k.