Rehabilitation therapy scheduling accounting for teaming requirements and therapist shortages: a branch and price approach

Sebastian Kling, Sebastian Schiffels, Jens O. Brunner

Angaben zur Veröffentlichung / Publication details:

Kling, Sebastian, Sebastian Schiffels, and Jens O. Brunner. 2025. "Rehabilitation therapy scheduling accounting for teaming requirements and therapist shortages: a branch and price approach." Flexible Services and Manufacturing Journal. https://doi.org/10.1007/s10696-025-09601-0.

Flexible Services and Manufacturing Journal
https://doi.org/10.1007/s10696-025-09601-0

Rehabilitation therapy scheduling accounting for teaming requirements and therapist shortages: a branch and price approach

Sebastian Kling[1] · Sebastian Schiffels[1] · Jens O. Brunner[1,2,3]

Accepted: 3 February 2025

Abstract
Physical therapy in acute care hospitals plays an important role for the rehabilitation of patients. Nevertheless, the profession must deal with staff shortages caused by a lack of qualified employees and stress-induced absenteeism. Both are results of high physical and mental workloads as well as a lack of employee retention strategies. A therapist shortage negatively affects the total number of appointments the department can fulfill daily. Furthermore, severe cases where patients require two therapists at the same time are common in acute care hospitals and contribute to the scheduling complexity. Here, one therapist takes charge of the appointment (lead), while a second therapist fulfills a support function role. This paper develops a multi-criteria optimization model for the daily rehabilitation therapy scheduling problem subject to teaming aspects and appointment priorities. We minimize preference penalties for lead and support visits and the total priority-based violation for unscheduled appointments. The problem is modeled as a vehicle routing problem with time windows and synchronization constraints. We solve the problem using a branch-and-price approach with different visit clustering methods and speed-up techniques. Computational results show the effectiveness of a randomized greedy heuristic implemented to enhance performance for generating new columns. Besides, a problem-specific clustering approach is integrated to speed up subproblems' solution times. Our results show its high effectiveness when compared to a state-of-the-art approach derived from literature.

Keywords OR in health services · Physical therapy · Vehicle routing problem · Time windows · Synchronization constraints · Workforce shortages

Extended author information available on the last page of the article

Published online: 09 March 2025

1 Introduction

Workforce shortages in healthcare professions are a well-known problem in hospitals and affect physical therapies. A shortage of physical therapists is prevalent in many countries like Germany (Bundesagentur für Arbeit 2022) and the U.S. (Bureau of Labor Statistics, U.S. Department of Labor 2022; Zimbelman et al. 2010). Moral distress and time pressure lead to burnout, unpleasant work environments, and high turnover rates among physical therapists (Lau et al. 2016). A multifaceted regulatory environment adds complexity to creating proper therapists' schedules. In many countries, like Germany, health insurance is required by law to have framework contracts with physical therapist unions which specify fixed durations for all potential appointments as well as treatment guidelines (Verband der Ersatzkassen 2022; Schimmelpfeng et al. 2012). From the scheduling perspective, a particularly challenging guideline is that some rehabilitation cases require two therapists simultaneously, where one therapist supports the responsible therapist (e.g., rehabilitation for severely affected neurology patients). As a result of the treatment requirements and the workforce shortages, it is common that the hospital's department for physical therapy struggles to provide treatment for all patients in the daily planning horizon. Depending on the priority of patient treatments, treatments might be moved to the next day. Some treatments might also be fulfilled only later by specialized rehabilitation institutions without endangering the patient. In this context, it is important to keep a focus on therapist satisfaction and development. In the scheduling literature, Erhard et al. (2018) for physicians, and Cheang et al. (2003) as well as Burke et al. (2004) for nurses, describe that employee preferences are often considered to improve job satisfaction and thus reduce absenteeism of hospital personnel. Furthermore, giving employees opportunities to take a lead function for their tasks will improve their identification with their employer (Cloutier et al. 2015). Overall, the management must prioritize patients according to internal priority rules to achieve the best outcome for them, while also keeping a focus on therapist satisfaction and development.

In this paper, we collaborate with one of the largest German hospitals, to tackle the daily (offline) operational rehabilitation scheduling problem in acute care hospitals. Three hierarchical priority classes (high, normal, and low) are used to prioritize patients. For severely affected high priority patients, rehabilitation treatments are a necessity in acute care to be able to fully recover, for normal priority patients, acute care treatments speed up recovery considerably, and for low priority patients, acute care treatments improve the well-being but there is little harm to do the treatment only later in a specialized rehabilitation institution. Thus, patients of a higher class are strictly prioritized, i.e., a single patient of a higher class is always prioritized compared to one or several patients of lower classes, where treatment in the acute care hospital might be missed altogether. Further, not all patients are available for therapy at all times, e.g., patients might be occupied by physicians or must leave the hospital early on in the planning day. In acute care hospitals, synchronization between two therapists is necessary to fulfill treatment for some of the patients. This is often the case when a patient can't follow the lead therapist's instructions alone without another therapist holding the patient upright. We assume that a patient treatment or support for a patient, can either be beneficial, neutral, or detrimental for a therapist's motivation. For example, a therapist who was in the lead for a particular patient before might want to be in the lead again, while a newly qualified therapist might prefer a support role if available. Information on therapist preferences for patients can usually be retrieved from management support systems by matching therapist characteristics and training with patient characteristics. Addressing this common rehabilitation therapy scheduling problem in hospitals, the contribution of the paper is manifold: 1) We model the operational rehabilitation scheduling problem with teaming requirements and workforce shortage-related objectives as a vehicle routing problem with time windows (VRPTW) and synchronization constraints. A complex VRP variant that received little attention (e.g., according to the taxonomy by Drexl (2012) or as described in Jungwirth et al. (2021)). 2) We model the problem with the objective of minimizing unscheduled appointments accounting for their priority while also including therapist preferences for appointments. We enforce the scheduling of all required teaming visits (lead and support) in feasible solutions. This novel modeling idea is uncommon in existing VRP literature (Vidal et al. 2020) and in home healthcare scheduling (HHCS) literature. In HHCS, synchronization is often relevant but not enforced for a feasible solution. Usually, the regulatory setting does not allow unscheduled appointments (Fikar and Hirsch 2017), and missing caregivers are then recruited from external sources. 3) We develop a branch-and-price approach with visit clustering and a problem-specific randomized greedy heuristic in the subproblem to solve realistic problem instances. Branch-and-price is a common technique to solve the NP-hard VRPTW with synchronization constraints (Desaulniers et al. 2014; Drexl 2012). To speed up the runtimes of the subproblem, we develop a randomized greedy heuristic, which we combine with a label correcting algorithm. Additionally, we develop and apply visit clustering approaches (Rasmussen et al. 2012) to manage the number of potential visits in a schedule. 4) In computational experiments, we examine the effects of different synchronization levels, and we show the validity of our speed-up techniques.

In the following Sect. 2, we examine the state of the art of scheduling in a rehabilitation setting and we discuss related research in home healthcare scheduling. Section 3 gives a problem description and introduces the mathematical model. Section 4 describes the branch-and-price procedure. Section 5 presents the computational study. Section 6 summarizes the findings and presents potential future research directions.

2 Literature review

In this paper, we schedule patients' rehabilitation treatments, which has some similarities to appointment scheduling, as described by Gupta and Denton (2008) and Ahmadi-Javid et al. (2017). However, a hospital typically knows all therapy patient treatments in advance for the next day (which is the case for our partner hospital) and does not have to plan with no-shows or walk-ins. Hence, the problem setup only requires scheduling patients occupying a hospital bed. In the introduction, we already described that treatment times are given by regulations to ensure a high treatment quality for patients, and these treatment times are deterministic. In contrast, more general appointment scheduling problems, often deal with stochastic service times. Nevertheless, there is existing inpatient rehabilitation therapy appointment literature with similar (deterministic) settings.

Chien et al. (2008) model a (daily) rehabilitation scheduling problem as a hybrid shop scheduling problem and solve the problem using a genetic algorithm. Like our problem, they assume time windows for patients. However, their objective is to improve patient satisfaction via minimizing makespan and waiting times. The same is true for Huynh et al. (2018) and Zhao et al. (2018), who solve similar scheduling problems with the same objectives using different genetic algorithms. Therapist satisfaction, patient priorities, travel times or teaming requirements are not part of their problem. Ogulata et al. (2008) acknowledge bad working conditions for physiotherapists and capacity restrictions. For a weekly planning problem, they therefore provide a three-stage mathematical programming model, where the first stage maximizes accepted patients according to their priority. In the second stage, they consider fairness among therapists by balancing time and patients among physiotherapists. In the final stage, they schedule patients in the schedule of the assigned therapist. Here, they do not assume teaming requirements or travel times. For a rehabilitation hospital, Griffith et al. (2012) develop a three-stage procedure using different meta-heuristics to solve a weekly rehabilitation scheduling problem including time windows, patient priorities, and teaming requirements for some appointments. They do not consider travel times or therapist preferences since their focus is a fair appointment distribution for patients without consideration of therapist satisfaction. Gartner et al. (2018), Jungwirth et al. (2021), and Frey et al. (2023) deal with a daily rehabilitation scheduling problem with travel times and treatment time windows in an acute care setting. Gartner et al. (2018) develop a time-indexed formulation with the objective to minimize waiting times for patients. They solve the problem using a cutting plane algorithm and a sequential allocation heuristic for larger problem sizes. Jungwirth et al. (2021) model the problem as a VRPTW and solve it using a branch-and-price-and-cut approach with the objective to minimize the cost of all selected routes. Frey et al. (2023) model their problem as a vehicle routing problem with time windows and flexible delivery locations and solve it using a hybrid adaptive large neighborhood search. However, the problems considered in these three papers do not account for patient priorities, therapist satisfaction or teaming requirements. Kling et al. (2024) develop a greedy randomized adaptive search procedure for a daily planning problem. They consider patient priorities, therapist preferences, travel times and treatment time windows, but they do not consider teaming requirements.

Table 1 summarizes important problem attributes and the described findings. As can be seen, many existing papers deal with a daily planning problem and all shown problems include treatment time windows. However, only four papers include travel times and three papers consider patient priorities. Two of the three problems which include patient priorities are not dealing with a daily planning horizon. Teaming requirements is only considered by Griffith et al. (2012) in a weekly planning problem. Due to workforce shortages, it becomes increasingly important to consider

Table 1 Relevant aspects in rehabilitation therapy scheduling

	Patient priorities	Therapist preference	Teaming requirements	Time windows	Travel time	Daily planning problem
Chien et al. (2008)				X		X
Ogulata et al. (2008)	X	X		X		
Griffith et al. (2012)	X		X	X		
Gartner et al. (2018)				X	X	X
Huynh et al. (2018)				X		X
Zhao et al. (2018)				X		X
Jungwirth et al. (2021)				X	X	X
Frey et al. (2023)				X	X	X
Kling et al. (2024)	X	X		X	X	X
This Paper	X	X	X	X	X	X

employee satisfaction. Here, only two problems consider therapist preferences. While Kling et al. (2024) share five of the six important problem attributes, no paper combines all problem requirements. Further, Kling et al. (2024) develop a meta-heuristic. This paper develops an exact method to solve the problem. For larger instances, the exact method is adapted to provide inexact but very good solutions.

We model our problem as a VRPTW in a healthcare context. This results in similarities to home healthcare routing and scheduling problems. Cissé et al. (2017) and Fikar and Hirsch (2017) give a broad overview over existing research and describe the home healthcare problem (HHCP) as patients scattered across a region requiring health services. Health service workers visit the patients at home to fulfill patient needs. Ait Haddadene et al. (2016) consider an objective function which minimizes the sum of non-preference of caretakers for patient visits as one part of their objective function and they include synchronization constraints. They solve their problem using a Greedy Randomized Adaptive Search Procedure. Qiu et al. (2022) solve a HHCP with synchronization constraints. Their problem aims to minimize the combined cost of dispatching caretakers and traveling costs. It is solved using a branch-and-price-and-cut algorithm. In general, in many regulatory settings within HHCP, the service provider is forced to fulfill all patient appointments. Soares et al. (2024) give a recent overview of vehicle routing problems with synchronization constraints. Therefore, only few HHCP paper deal with the possibility of missed appointments in a daily planning horizon. Instead, the focus usually lies on minimizing travel costs or travel time, balancing workload or minimizing overtime

(Fikar and Hirsch 2017). An exception is Dohn et al. (2009). They consider a daily planning problem with teaming requirements. The authors maximize the number of fulfilled appointments and solve the problem using a branch-and-price algorithm. They do not consider prioritization or employee preferences. Rasmussen et al. (2012) solve a HHCP with a daily planning horizon and temporal dependencies between caretakers where complete synchronization can occur. They apply a branch-and-price algorithm with different visit clustering methods to speed up solution times. Their objective function minimizes prioritized missed visits, caretaker preferences for patient visits, and travel costs for a daily HHCP. They do not differentiate between a lead and a support role for caretakers since caretakers with temporal dependencies usually have different qualifications and are not forced to work in a team for feasible solutions.

Summarizing, existing literature in rehabilitation therapy scheduling and HHCP has some similarities to the problem of our partner hospital that we consider. However, no existing research considers task specific preferences for leading or supporting individual treatments. In our setting both, the lead and support functions must be scheduled in a feasible solution. Furthermore, in our acute care rehabilitation setting, we allow for unscheduled appointments.

3 Problem definition and mathematical model

Before each day, the planner must consider several influencing factors for the patient appointment scheduling. The daily (offline) operational planning is influenced by regulations, i.e., given treatment times in periods and synchronization requirements. The planner has a set $e \in \boldsymbol{E}$ of therapists with the same qualification. Each therapist e works a schedule from the set $\boldsymbol{S^e}$. Set $\boldsymbol{S^e}$ is the set of possible daily schedules of a therapist. Required therapist treatments are represented by visits on the route. The number of required visits, i.e., visit(s) by one or two therapists is a result of regulations. $\boldsymbol{M}$ is the set of lead visits for the day, i.e., visits with the therapist leading a treatment. Therefore, *s*et $\mathbf{M}$ includes all patients. $\mathbf{H}$ is the set of support visits for patients with synchronization needs. For patients with synchronization requirement, $(i,j) \in \boldsymbol{P}$ is the set of two visits necessary to treat a patient, with $i \in \boldsymbol{M}$ and $j \in \boldsymbol{H}$. Here, the index M^j stands for the lead patient visit $i \in \boldsymbol{M}$ associated with a support visit $j \in \boldsymbol{H}$ if synchronization is required. Finally, set $\boldsymbol{V} = \boldsymbol{M} \cup \boldsymbol{H}$ contains all necessary visits to treat all patients.

The binary parameter $A_{e,i,s} = 1$ if a visit $i \in \boldsymbol{V}$ is included in a schedule $s \in \boldsymbol{S^e}$ of a therapist e, otherwise $A_{e,i,s} = 0$. The parameter $T_{e,i,s}$ is the start period of visit $i \in \boldsymbol{V}$ in a schedule $s \in \boldsymbol{S^e}$ of a therapist e. $T_{e,i,s} = 0$ if a schedule $s \in \boldsymbol{S^e}$ of therapist e does not cover visit i. Parameter Z_i is the cost of missing a patient, i.e., a lead visit $i \in \boldsymbol{M}$ is missed. We use three different priority classes for appointments and formalize their cost relation as follows. Let $\boldsymbol{M}^{low}$, $\boldsymbol{M}^{normal}$, and $\boldsymbol{M}^{high}$ be distinct subsets of $\boldsymbol{M}$. Then $Z_i > \sum_{j \in \boldsymbol{M}^{low}} Z_j \forall i \in \boldsymbol{M}^{normal}$ and $Z_i > \sum_{j \in \boldsymbol{M}^{normal}} Z_j \forall i \in \boldsymbol{M}^{high}$. Parameter $C_{e,s}$ gives the preference cost of a schedule $s \in \boldsymbol{S^e}$ for the individual therapists $e \in \boldsymbol{E}$. Individual therapist's preference costs use a value of -1 for beneficial visits, 0 for neutral visits, and 1 for detrimental visits. To balance our two objectives, we introduce weight w^{Missed} for the importance of missed visits (first objective) compared to the preferences of the chosen schedules for therapists (second objective). Binary decision variable $y_i = 1$ if a visit $i \in \boldsymbol{M}$ is missed. If all visits are covered $y_i = 0$ for all $i \in \boldsymbol{M}$. Binary decision variable $\lambda_{e,s} = 1$ if a therapist e is assigned to schedule $s \in \boldsymbol{S^e}$, otherwise $\lambda_{e,s} = 0$. In the following, we give an overview of the sets and indices, the parameters, and the decision variables, before we provide the model.

$$\min \ w^{\text{Missed}} \sum_{i \in \boldsymbol{M}} Z_i y_i + \sum_{e \in \boldsymbol{E}} \sum_{s \in \boldsymbol{S^e}} C_{e,s} \lambda_{e,s} \tag{1}$$

s.t.

$$\sum_{e \in \boldsymbol{E}} \sum_{s \in \boldsymbol{S^e}} A_{e,i,s} \lambda_{e,s} + y_i = 1, \forall i \in \boldsymbol{M} \tag{2}$$

$$\sum_{e \in \boldsymbol{E}} \sum_{s \in \boldsymbol{S^e}} A_{e,j,s} \lambda_{e,s} + y_{M^j} = 1, \forall j \in \boldsymbol{H} \tag{3}$$

$$\sum_{s \in \boldsymbol{S^e}} \lambda_{e,s} = 1, \forall e \in \boldsymbol{E} \tag{4}$$

$$\sum_{e \in \boldsymbol{E}} \sum_{s \in \boldsymbol{S^e}} T_{e,i,s} \lambda_{e,s} = \sum_{e \in \boldsymbol{E}} \sum_{s \in \boldsymbol{S^e}} T_{e,j,s} \lambda_{e,s}, \forall (i,j) \in \boldsymbol{P} \tag{5}$$

$$y_i \in \{0, 1\}, \forall i \in \boldsymbol{M} \tag{6}$$

$$\lambda_{e,s} \in \{0, 1\}, \forall e \in \boldsymbol{E}, s \in \boldsymbol{S}^e \tag{7}$$

Sets and indices

$e \in \boldsymbol{E}$	Set of therapist employees
$s \in \boldsymbol{S^e}$	Set of potential schedules of a therapist $e \in \boldsymbol{E}$
$\boldsymbol{M}$	Set of lead visits available for the day
$\boldsymbol{H}$	Set of support visits necessary for patients with synchronization needs
$(i,j) \in \boldsymbol{P}$	Set of visit combinations for patients with synchronization with $i \in \boldsymbol{M}$ and $j \in \boldsymbol{H}$
$\boldsymbol{V} = \boldsymbol{M} \cup \boldsymbol{H}$	Set of all patient visits for the day
M^j	Index for a lead visit corresponding to a support visit $j \in \boldsymbol{H}$
Parameters	
M^j	Index for the lead patient visit associated with a support visit j
$C_{e,s}$	Preference cost of a schedule $s \in \boldsymbol{S}^e$ of the therapist $e \in \boldsymbol{E}$
$A_{e,i,s}$	1, if visit $i \in \boldsymbol{V}$ is covered in schedule $s \in \boldsymbol{S^e}$ of therapist $e \in \boldsymbol{E}$,
0 otherwise	
$T_{e,i,s}$	Start period of visit $i \in \boldsymbol{V}$ in schedule $s \in \boldsymbol{S^e}$ of therapist $e \in \boldsymbol{E}$
Z_i	Cost of missing an appointment $i \in \boldsymbol{M}$
w^{Missed}	Objective function weight for the term for missed appointments

continued

Decision variables	
$\lambda_{e,s}$	1, if therapist e is assigned to $s \in \boldsymbol{S^e}$
0, otherwise	
y_i	1, if an appointment $i \in \boldsymbol{M}$ is not covered within its time window,
0, otherwise	

Objective function (1) minimizes the weighted sum of penalties for missed appointments according to priority in the first term. We use weight w^{Missed} to emphasize the first objective function term in relation to the second. The second term considers the sum of preference costs of the chosen schedules (columns). We do not include a weight for preference costs in the second term because each schedule is weighted accordingly in $C_{e,s}$ (see Sect. 4.1). Cost $C_{e,s}$ consists of the sum of weighted preference penalties for leading or supporting individual appointments within schedule $s \in \boldsymbol{S^e}$, where e defines individual therapists $e \in \boldsymbol{E}$. Constraints (2) guarantee that a lead visit $i \in \boldsymbol{M}$ is either covered by a schedule or it is missed. Constraints (3) ensure that a support visit $j \in \boldsymbol{H}$ with synchronization can only be assigned if the corresponding lead visit $i = M^j$ is scheduled (i.e., $y_i = y_{M^j} = 0$). On the other side, the constraints (3) assure that if a lead visit is missed then the corresponding support visit cannot be assigned (i.e., $y_i = y_{M^j} = 1$). Constraints (4) ensure that each therapist works one schedule. Constraints (5) ensure synchronization of lead and support visits in case of a patient with synchronization needs. Decision variable domains are defined in (6) and (7). Please note, the binary condition in (6) is not needed.

To directly solve the introduced extensive formulation, all possible schedules $s \in \boldsymbol{S^e}$ for therapists $e \in \boldsymbol{E}$ would have to be enumerated. However, there are many combinatorial possibilities for schedules and generating all of them is impossible in a reasonable amount of time. Next, we introduce our decomposition idea which is based on a branch-and-price framework including column generation to price out additional columns. Branch-and-price is a well-known approach to solve VRPTWs efficiently (Desaulniers et al. 2014) and it is well suited when dealing with synchronization requirements (Drexl 2012).

4 Branch-and-price procedure

In this section, we introduce the building blocks for our branch-and-price algorithm. First, we relax the synchronization constraints (5) and integrality in (6) and (7) and handle both during branching (see Sects. 4.2.1 and 4.2.2). The remaining constraints (2) to (4) and the objective function form the restricted master problem (RMP). In each main iteration of the column generation approach the RMP is solved to LP optimality. Once the RMP is solved, the dual solution is used to find additional columns (i.e., new schedules). We define $\chi_i \geq 0$ as the dual variables corresponding to constraints (2) in the RMP. Dual variables for constraints (3) and (4) are defined by

ψ_j and ω_e. Using the dual solution, we can then define the generic reduced cost of a column in the RMP as follows.

$$\overline{c}_{e,s} = C_{e,s} - \sum_{\forall i \in \boldsymbol{M}} A_{e,i,s}\chi_i - \sum_{\forall j \in \boldsymbol{H}} A_{e,j,s}\psi_j - \omega_e \tag{8a}$$

Dual feasibility is achieved when all columns in the RMP have non-negative reduced costs and no columns with negative reduced costs can be found given the solution values for χ_i, ψ_j, and ω_e for all $i \in \boldsymbol{M}$, $j \in \boldsymbol{H}$, and $e \in \boldsymbol{E}$. To verify that no $\overline{c}_{e,s} < 0$ for all $e \in \boldsymbol{E}$, $s \in \boldsymbol{S}^e$ exists, we minimize the objective function of the subproblem (SP, also called pricing problem) for each therapist to find new schedules (i.e., the most promising columns with negative reduced costs). The generic SP is defined in Sect. 4.1. Columns with negative reduced costs are added to the RMP for the next iteration. Here, we add all generated columns with negative reduced costs to the RMP. If no columns with negative reduced costs are found, then the LP relaxation of the respective node is solved. Branching then guarantees feasibility, including constraints (5) to (7). Implementation details, such as the high-level architecture, speed-up approaches with the use of a heuristic and visit clustering to further speed up solution retrieval are discussed in Sects. 4.2 and 4.3.

4.1 Pricing problem

The purpose of the pricing subproblem is to find non-basic feasible schedules (columns) for a therapist with negative reduced cost. These schedules account for therapist preferences for lead and support tasks. Therapists have different preference costs for patients. Therefore, a SP is necessary for each distinct therapist. We use the following notation to develop the generic $SP(e)$ for therapist $e \in \boldsymbol{E}$. To ensure better readability, we omit index e on all parameters and decision variables. We define a generic SP formulation for all pricing problems. Since the problem is modeled as a VRP, index 0 (n) defines the start (end) node of a given therapist $e \in \boldsymbol{E}$. We consider different shift times and lunch breaks. Therefore $\boldsymbol{V}^{SP} \subset \boldsymbol{V}$ is the set of visits which can be scheduled within the schedule of a therapist (if necessary, including a lunch break visit). Lunch breaks are enforced in the individual schedules by giving break visits a high negative penalty score in the individual SP, if required. Between any two appointments $i, j \in \boldsymbol{V}^{SP}$, we consider deterministic travel times $D_{i,j}$ (given in time periods). Each appointment i has a time window. Parameter B_i is the start period of a time window while F_i is the end period of a time window. L_i is the deterministic treatment duration. Parameter P_i^{Lead} $\left(\mathrm{P}_i^{Support}\right)$ defines the preference cost to handle lead (support) appointment $i \in \boldsymbol{M}(\boldsymbol{H})$. As outlined earlier, a patient visit can be beneficial, neutral, or detrimental for a therapist's motivation. We model this for both preference cost with a value of -1 for beneficial visits, 0 for neutral visits, and 1 for detrimental visits. Using a negative value for desirable visits and a value of zero for a visit which does not considerably affect a therapist's motivation simplifies branching significantly as shown in Rasmussen et al. (2012). Enabling management to prioritize the two preference cost terms depending on the needs for leadership training and basic employee development as well as to include a possibility to better manage personnel in case of absences, we introduce the two additional objective function weights w^{Lead} and $w^{Support}$. Weight w^{Lead} defines the importance for lead preferences while weight $w^{Support}$ reflects the importance of support preferences. Binary decision variable $x_{i,j} = 1$ if visit j is directly scheduled after visit i. Auxiliary binary decision variable $z_i^{Lead} = 1$ if a lead visit $i \in \boldsymbol{M}$ is scheduled. The same is true for auxiliary binary variable $z_i^{Support}$. Finally, we use the decision variable t_i for the start period of a visit $i \in \boldsymbol{V}^{SP}$.

$$min \sum_{i \in \boldsymbol{M} \cap \boldsymbol{V}^{SP}} \left(w^{Lead} \mathrm{P}_i^{Lead} - \chi_i\right) z_i^{Lead} + \sum_{i \in \boldsymbol{H} \cap \boldsymbol{V}^{SP}} \left(w^{Support} P_i^{Support} - \psi_i\right) z_i^{Support} - \omega \tag{8b}$$

s.t.

$$\sum_{j \in \boldsymbol{V}^{SP} \cup \{n\}} x_{i,j} = z_i^{Lead}, \forall i \in \boldsymbol{M} \cap \boldsymbol{V}^{SP} \tag{9}$$

$$\sum_{j \in \boldsymbol{V}^{SP} \cup \{n\}} x_{i,j} = z_i^{Support}, \forall i \in \boldsymbol{H} \cap \boldsymbol{V}^{SP} \tag{10}$$

$$\sum_{j \in \boldsymbol{V}^{SP}} x_{i,j} \leq 1, \forall i \in \boldsymbol{V}^{SP} \tag{11}$$

$$\sum_{j \in \boldsymbol{V}^{SP} \cup \{n\}} x_{0,j} = 1 \tag{12}$$

$$\sum_{i \in \boldsymbol{V}^{SP} \cup \{0\}} x_{i,n} = 1 \tag{13}$$

$$\sum_{i \in \boldsymbol{V}^{SP} \cup \{0\}} x_{i,k} - \sum_{j \in \boldsymbol{V}^{SP} \cup n} x_{k,j} = 0, \forall k \in \boldsymbol{V}^{SP} \tag{14}$$

$$B_i \sum_{j \in \boldsymbol{V}^{SP} \cup \{n\}} x_{i,j} \leq t_i \forall i \in \boldsymbol{V}^{SP} \cup \{0\} \tag{15}$$

$$t_i \leq F_i \sum_{j \in \boldsymbol{V}^{SP} \cup \{n\}} x_{i,j} \forall i \in \boldsymbol{V}^{SP} \cup \{0\} \tag{16}$$

$$B_0 \leq t_0 \tag{17}$$

$$t_n \leq F_n \tag{18}$$

$$t_i + \left(L_i - 1 + D_{i,j}\right) x_{i,j} \leq t_j + F_i\left(1 - x_{i,j}\right), \forall i \in \boldsymbol{V}^{SP} \cup 0, j \in \boldsymbol{V}^{SP} \cup \{n\} \tag{19}$$

$$z_i^{Lead} + z_j^{Support} \leq 1, \forall (i,j) \in \boldsymbol{P} \tag{20}$$

$$z_i^{Lead} \in \{0,1\}, \forall i \in \boldsymbol{M};\ \boldsymbol{z_i^{Support}} \in \{\boldsymbol{0},\boldsymbol{1}\}, \forall \boldsymbol{i} \in \boldsymbol{H};\ \boldsymbol{t_i} \in \mathbb{N}, \forall \boldsymbol{i} \in \boldsymbol{V^{SP}} \tag{21}$$

Additional sets and indices

$\boldsymbol{V}^{SP}$	Set of possible visits within a SP (with lunch break visit)
0	Source node
n	Sink node
Parameters	
B_i	Begin time window for a visit $i \in \boldsymbol{V}^{SP}$, i.e., earliest possible start of a visit
F_i	End time window for a visit $i \in \boldsymbol{V}^{SP}$, i.e., latest possible start of a visit
$D_{i,j}$	Travel time between visits $i \in \boldsymbol{V}^{SP}$ and $j \in \boldsymbol{V}^{SP}$ in periods
L_i	Duration in periods for visit $i \in \boldsymbol{V}^{SP}$
P_i^{Lead}	Lead preference cost for visit $i \in \boldsymbol{M}$
$\mathrm{P}_i^{Support}$	Support preference cost for visit $i \in \boldsymbol{H}$
w^{Lead}	Cost weight for lead preferences
$w^{Support}$	Cost weight for support preferences
Dual RMP values	
χ_i	Dual value corresponding to constraints (2)
ψ_j	Dual value corresponding to constraints (3)
ω	Dual value corresponding to constraints (4)
Decision variables	
$x_{i,j}$	1, if visit $j \in \boldsymbol{V}^{SP}$ is directly scheduled after visit $i \in \boldsymbol{V}^{SP}$,
0, otherwise	
z_i^{Lead}	1, if lead visit $i \in \boldsymbol{M}$ is scheduled,
0, otherwise	
$z_i^{Support}$	1, support visit $i \in \boldsymbol{H}$ is scheduled,
0, otherwise	
t_i	Start period of a visit $i \in \boldsymbol{V}$

Objective function (8b) determines the generic reduced costs for a column in $SP(e)$ in subproblem notation. It is derived from (8a) Constraints (9) and (10) connect auxiliary variables z_i^{Lead} or $z_i^{Support}$ with decision variable $x_{i,j}$. If visit i is scheduled, i. e., $x_{i,j} = 1$, then $z_i^{Lead} = 1$ or $z_i^{Support} = 1$, respectively. Remember, lead visits $i \in \boldsymbol{M}$ and support visits $i \in \boldsymbol{H}$ are part of two disjunct subsets. Constraints (11) ensure a visit can at most be left once. Constraints (12) and (13) ensure that the tour starts in node 0 and ends in node n. For any other node, constraints (14) guarantee flow balance. Constraints (15) and (16) model the time windows for appointments. If $x_{i,j} = 1$ (i.e., visit i takes place), then t_i must be between B_i and F_i. Constraints (17) and (18) enforce the shift start and end periods while constraints (19) consider the feasible sequence of appointments within a schedule. If visit j directly follows visit i $\left(x_{i,j} = 1\right)$, then the start period t_j must be later then the finishing period of i (i.e.,

$t_i + L_i$) plus the travel time $D_{i,j}$ between i and j. If visit j does not follow visit i $\left(x_{i,j} = 0\right)$, then the corresponding constraint is not binding. Constraints (20) ensure a therapist cannot handle a lead and a support visit in the same schedule. Including this restriction reduces the number of generated columns. Note, synchronization is not handled directly in the SPs which would forbid the joint assignment. Constraints (21) define the domains of the decision variables.

The resulting problem is an elementary shortest path problem with time windows (ESPPTW). We solve this ESPPTW by extending a label correcting algorithm by Feillet et al. (2004) with a randomized greedy heuristic and adapt it to our problem setting. The next section describes our branching scheme which handles the omitted constraints (5) to (6) for synchronization and integrality and other building blocks of our solution approach such as the tailored label correcting algorithm.

4.2 Building blocks of the solution approach

In this section, we describe the two branching schemes necessary to reach feasible solutions. We describe how the SPs are solved using a randomized greedy heuristic and a label correcting algorithm. Different visit clustering approaches are also described in this section. These are used to reduce the set of possible visits for very broadly qualified therapists to speed up solving the corresponding SP.

4.2.1 Branching to achieve synchronization

Infeasibility due to missed synchronization can only appear for partner visits $(i,j) \in \boldsymbol{P}$. Partner visits $(i,j) \in \boldsymbol{P}$ have the same start and end periods to their respective time windows, i.e., $B_i = B_j$ and $F_i = F_j$. Figure 1 shows an example where six partial schedules ($\lambda_{e,s} > 0$ in RMP) are shown which include one pair of partner visits $(i,j) \in \boldsymbol{P}$. Since the start times of the two visits are not synchronized due to six different start times, branching is necessary. Lead visit i (labeled with a circle) is assigned to schedules 1, 3, and 5 while matching support visit j (labeled with a square) takes place in schedules 2, 4, and 6. For simplicity, let the six schedules represent schedules of six different therapists $e \in E$. The time window is given from the left to the right and each visit per schedule 1 to 6 is mapped to it. For simplicity, we assume the time window starts at period 1 and ends at period 9 for the

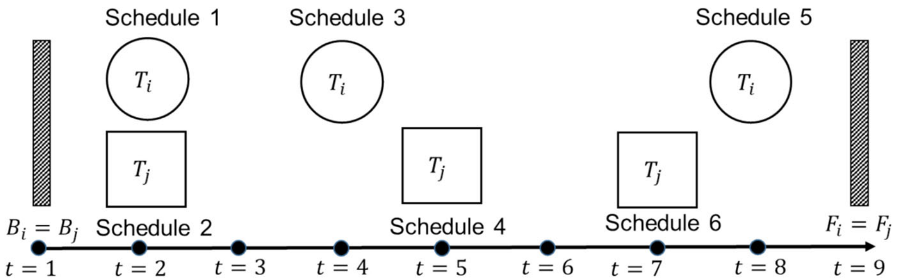

Fig. 1 Start periods of partner visits within six different schedules

lead/partner visits. It is not enough that the lead visit in *Schedule 1* and the support visit in *Schedule 2* start at the same time period $T_i = T_j = 2$. Note, not all six potential start periods are synchronized, i.e., the treatment start period is not the exact same across all six schedules meaning there is no adherence to constraints (5).

We branch on time windows using the fact that $B_i = B_j$ and $F_i = F_j$. Time window branching was shown to be a good possibility for synchronization and temporal dependencies in Dohn et al. (2009), Dohn et al. (2011), and Rasmussen et al. (2012). Let s_i be the split period, where we split the original time window $[B_i, F_i]$ into two distinct time windows $[B_i, s_i - 1]$ and $[s_i, F_i]$ at the midway point between the earliest and the latest start time of two partner visits $(i,j) \in \boldsymbol{P}$ for the two branches. For partner visits $(i,j) \in \boldsymbol{P}$, we calculate the split time s_i as:

$$s_i = \frac{\min\limits_{e \in \boldsymbol{E}, s \in \boldsymbol{S}^e \left| \lambda_{e,s} \right\rangle 0} \left(T_{e,i,s}, T_{e,j,s}\right) + \max\limits_{e \in \boldsymbol{E}, s \in \boldsymbol{S}^e \left| \lambda_{e,s} \right\rangle 0} \left(T_{e,i,s}, T_{e,j,s}\right)}{2} \forall (i,j) \in \boldsymbol{P}.$$

For the given example, $s_i = \lceil \frac{\min(2,2) + \max(8,7)}{2} \rceil = 5$, time windows in the two branches then become $[1, 4]$ and $[5, 8]$ for $(i,j) \in \boldsymbol{P}$.

Ranking the time window branching candidates, we use a similar approach as Dohn et al. (2009). We define $\boldsymbol{S}_i^e$ ($\boldsymbol{S}_j^e$) as the subset of therapist schedules containing a visit i (j) from $(i,j) \in \boldsymbol{P}$, and we define G_i^- as the sum of schedule variables containing a partner visit i or j in $(i,j) \in \boldsymbol{P}$, where the respective visit is scheduled before the split period.

$$G_i^- = \sum_{e \in \boldsymbol{E}, s \in \boldsymbol{S}_i^e \cup \boldsymbol{S}_j^e \left| \lambda_{e,s} \right\rangle 0 \vee T_{e,i,s} < s_i \vee T_{e,j,s} < s_i} \lambda_{e,s} \forall (i,j) \in \boldsymbol{P}$$

The branching candidate is chosen as the pair$(i,j) \in \boldsymbol{P}$, where visits are spread most evenly before and after the split period. We only include schedules where solution values for$\lambda_{e,s} > 0$. Partner visits $(i,j) \in \boldsymbol{P}$ cannot be scheduled in the same schedule. Constraints (2) and (3) ensure $\sum\limits_{e \in \boldsymbol{E}, s \in \boldsymbol{S}_i^e \cup \boldsymbol{S}_j^e \left| \lambda_{e,s} \right\rangle 0} \lambda_{e,s} = 2 \forall (i,j) \in \boldsymbol{P}$ which results in the following formula for the branching candidate $(i,j)^*$.

$$(i,j)^* = \underset{(i,j) \in \boldsymbol{P}}{\operatorname{argmin}} \left| \frac{G_i^-}{2} - 0.5 \right|$$

In the worst case, time windows must be split until only the treatment periods of a visit fit, i.e., there is no potential slack in the time window anymore. When determining a candidate, time windows of $(i,j) \in \boldsymbol{P}$ are split for both partner visits simultaneously.

After branching, modifications in the child nodes are necessary for the RMP and all SPs. In the RMP, we delete all schedules that include i or j with starting periods outside the modified time window. In SPs, potential start periods of the partner visits are restricted to the modified time window of the corresponding child. It has been shown in literature that time window branching is beneficial for achieving integrality enforced by constraints (6) and (7) (Gélinas et al. 1995). However, it is not sufficient. Therefore, a second branching scheme is necessary.

4.2.2 Branching for integer feasibility

Several different integer branching schemes are known for achieving integrality. The most common is to force that any visit can only be done by one therapist e. In one branching node, a visit must be removed as possibility from the chosen candidate therapist. In the second node, the visit is removed as possibility from all other available therapists, accordingly, it can only be fulfilled by the candidate therapist or not at all. We modify RMP and all SPs by removing/adding columns in RMP and enforcing/forbidding visits in SPs. To achieve a balanced branching tree, we choose the branching candidate that is the most fractional. Let Q_i^e be the sum of schedules where a therapist takes visit $i \in \boldsymbol{V}$.

$$Q_i^e = \sum_{s \in \boldsymbol{S}_i^e} \lambda_{e,s} \forall e \in \boldsymbol{E}, i \in \boldsymbol{V}$$

$$(e,i)^* = \underset{(e,i) \in \boldsymbol{E} \times \boldsymbol{V} | 0 < Q_i^e < 1}{\operatorname{argmin}} \left| Q_i^e - 0.5 \right|$$

We then formally define $(e,i)^*$ as the most fractional branching candidate. In our implementation, we prefer to choose branching on time windows whenever possible.

4.2.3 Label correcting algorithm

Our pricing SP is an ESPPTW. To solve the problem, we adapt the algorithm by Feillet et al. (2004). The algorithm finds the path from a source to a sink by extending labels, i.e., resource information connected to partial paths to each considered node, while abiding to resource constraints. Note, not all labels from the start node through several nodes to the considered node must be extended since some partial paths are pareto-dominated by other partial paths. Therefore, the algorithm can be solved to optimality in pseudo polynomial time. For detailed information concerning the algorithm and the dominance rules, we refer the reader to Feillet et al. (2004). Please note, partner visits $(i,j) \in \boldsymbol{P}$ must be scheduled at the same periods (see constraints (20)), i.e., they cannot be in the same schedule of a therapist. Once one of the two partner visits is part of a partial path, the partner visit is added to the vector of unreachable nodes for the respective labels, i.e., it cannot be visited by the (partial) path anymore.

In general, each therapist can be assigned to any visit. For realistically large problem instances, this leads to large networks of potential visits and a lot of processing time for the label correcting algorithm, which is a main bottleneck for our implementation. To speed up solution times of the SP, we introduce a randomized greedy heuristic. We use it before the label correcting algorithm.

4.2.4 Randomized greedy heuristic

The randomized greedy heuristic finds additional columns fast compared to the (exact) label correcting algorithm. The randomized heuristic tries to insert a visit from a restricted candidate list into a therapist's schedule at the earliest possible position. Initially, the therapist's schedule only consists of the therapists' start and end visits. The visit is randomly drawn from the restricted candidate list consisting of several (unscheduled) visits with the most negative objective function contribution, i.e., $\left(w^{Lead}\mathrm{P}_i^{Lead} - \chi_i\right)$ for lead visits $i \in \boldsymbol{M}$ or $\left(w^{Support}\mathrm{P}_i^{Support} - \psi_i\right)$ for support visits $i \in \boldsymbol{H}$. From the second iteration onwards, already scheduled visits move forward in time after an insertion takes place. In other words, we schedule a candidate i at the earliest possible period (i.e., start of time window or after the predecessor). Additionally, we need to check feasibility (time windows of visits and the shift end period of therapist e) for all successor visits (e.g., j and k in Fig. 2 when i is scheduled at *Position 1*). If an insertion is not possible at the current position, we move to the next possible position. If we reach the last position and candidate i cannot be scheduled, then the visit is left unscheduled and is discarded from consideration. After a successful insertion, we rebuild the restricted candidate list (i.e., three visits with the most negative objective function contribution) and try to add additional visits to the current schedule.

In Fig. 2, we give a visual example.

While building the schedule with candidates, we add all found columns with negative reduced costs to the RMP. If a candidate could not be scheduled, or we found a schedule without negative reduced costs, the process is restarted with the initially empty schedule until $SP(e)$ added several columns to the RMP or no additional columns with negative reduced costs are found. A pseudo-code of the algorithm can be seen in Appendix A.

To ensure optimality in each column generation iteration, the label correcting algorithm must be run at least once for each SP, when the randomized greedy heuristic does not find additional columns anymore. Solving the SP therefore remains a bottleneck. To further speed up solving the SP, we apply visit clustering which was introduced by Rasmussen et al. (2012).

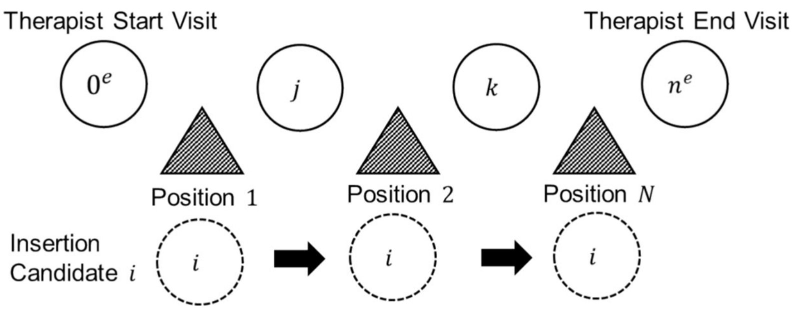

Fig. 2 Insertion testing of randomized greedy heuristic

4.2.5 Visit clustering and extension

Rasmussen et al. (2012) introduced visit clustering to restrict the number of visits in the individual visit pools of a care worker. They divide the pool of possible visits between the workers in a preprocessing step, i.e., the set of possible visits in V^{SP} is restricted to reduce the size of the individual ESPPTW in a SP. Please remember, V^{SP} is therapist specific. As a result, feasible solutions for larger and more complex problem instances become retrievable. On the other hand, clustering renders the algorithm heuristic. Therefore, for comparison, we implement different clustering approaches for determining V^{SP}.

Our problem includes comparable characteristics as Rasmussen et al. (2012), namely preferences for patients and the possibility of missing visits by accounting for their priority. For one of the approaches (*PREF*), we therefore follow ideas proposed by the authors. In a good solution, it is likely that a therapist mostly treats patients with high preference adherence. We therefore choose potential visits randomly from the list of visits which are not part of a cluster yet. Therapists are then ordered in non-increasing order according to preference adherence, i.e., therapists with a high preference for the chosen visit are considered first. If the first therapist in the list did not reach the desired cluster size yet and a potential partner visit is not in the cluster, the visit is clustered to the therapist. Then the visit is removed from the list of remaining visits to be clustered. Otherwise, the second therapist in the list is considered. If all therapists have reached the desired cluster size and some visits are not clustered yet, all remaining visits are added to all therapists equally. Using a defined cluster size for adding visits to therapist visits according to preferences gives several advantages from a managerial point of view. First, the scheduler has input on how many visits should be in a cluster depending on the daily ratio between visits and therapists. Second, preferences are usually not divided equally among all therapists, i.e., while one therapist has many visits, other therapists might have only a few visits which they truly prefer. Here, control over the cluster size might increase fairness and workload balancing aspects.

The first clustering approach only focuses on preferences. However, patient priority plays an important role for the problem setting. Omitting consideration of priorities when clustering might lead to situations where many appointments with high priority might be clustered to the same therapists due to high preferences. To avoid this behavior, we introduce a second clustering approach (*PRIO*). We first sort the available visits accounting for their priority in non-increasing order, i.e., we consider the most important visits first. Support visits get the same priority as their lead visit. For each considered visit, we order the therapists first according to the number of visits already in their cluster and second according to non-increasing order of preference adherence. We therefore treat the potential therapists in a round-robin fashion where therapists with the biggest difference between already scheduled visits and cluster size are considered first and ties are broken with preference adherence. If a visit is clustered to a therapist, the visit cannot be clustered to a second therapist. A visit cannot be clustered to a therapist if the partner visit is already in the cluster, or the therapist reached the desired cluster size. If all therapists reached the cluster size, remaining visits are clustered to all therapists equally.

To compare the two problem specific clustering approaches, a third clustering approach (*RAND*) schedules visits to therapists randomly (unless a partner visit is in the cluster), without consideration of preferences or priorities. After all therapists reached the cluster size, all remaining visits are added to all clusters equally.

After the root node is solved, some visits might not be scheduled, simply due to the cluster they were assigned to not because it is optimal. To improve results, Rasmussen et al. (2012) introduce the concept of cluster expansion. Once a visit is missed, i.e., $\sum_{e \in \boldsymbol{E}} \sum_{s \in \boldsymbol{S}^e} A_{e,i,s} \lambda_{e,s} = 0$ for a visit in the RMP, the visit is added to all therapists' clusters. To get a predictable behavior in the branching, we apply cluster extension only in the root node.

4.3 Implementation

Figure 3 shows the high-level architecture of the branch-and-price procedure. Before running the algorithm, we first apply one of the introduced visit clustering approaches to restrict the possible visits for each therapist. Clustering is therefore utilized regardless of how the SP is solved (i.e., with algorithmic approaches or commercial solver). We start the algorithm with an initial solution, where each therapist $e \in E$ only visits their respective start visit 0 and end visit n. After we solve the RMP, we first use a randomized greedy heuristic to find new schedules with

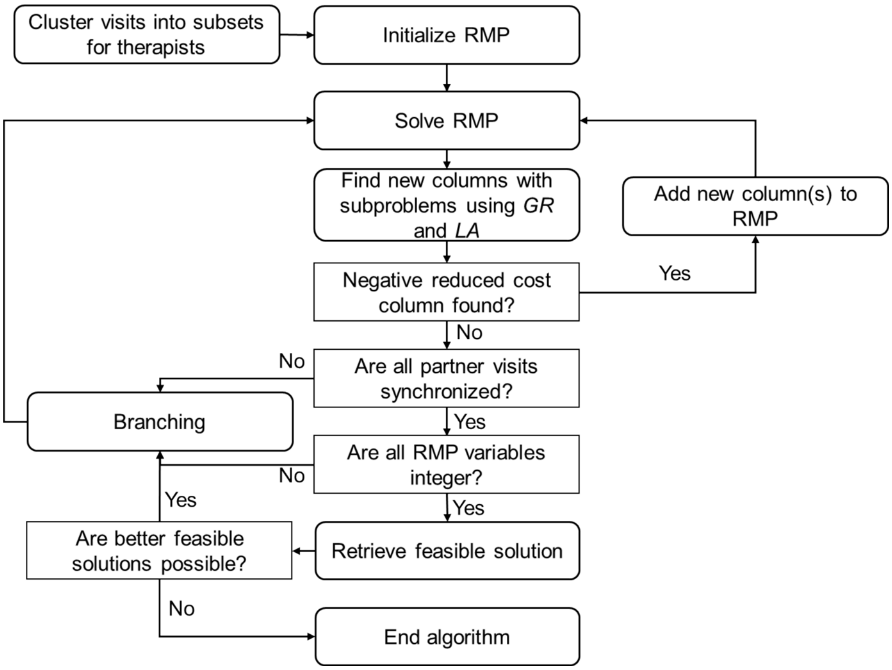

Fig. 3 High-level architecture of the branch-and-price algorithm

negative reduced costs for the next RMP iteration. If the heuristic is unable to find additional schedules for a therapist e, a label correcting algorithm is run to potentially find more schedules for the therapist. Adding several columns to the RMP instead of only the column with the most negative reduced cost, populates the column pool fast and leads to shorter overall processing times (Tanoumand and Ünlüyurt 2021). If the label correcting algorithm does not find additional schedules with negative reduced costs for any of the therapists, we need to determine if the achieved solution of the RMP in the root node is feasible regarding constraints (5) to (7). If that is the case, we derive the (integer) solution. Otherwise, we branch on time windows or apply a $0/1$ branching on combinations between a therapist and a visit. We prefer time window branching to $0/1$ branching if a suitable branching candidate exists. If nodes with potentially better lower bounds exist, we continue branching. We branch until all nodes are checked for better solutions or pruned.

5 Experimental study

The branch-and-price algorithm was built using Python 3.8, and experiments were run on a virtual machine with an Intel Xeon Gold 5218 CPU @ 2.30 GHz processor and 16 GB RAM. The restricted master problem and an implementation of the subproblem as MILP model for comparison are solved with Gurobi 10.0 (Gurobi Optimization 2022). In Sect. 5.1, we describe the data for the experiments. Section 5.2 solves a small problem instance to optimality. Section 5.3 discusses the performance of the label correcting algorithm and the additional benefits of the randomized greedy heuristic compared to an implementation of the subproblem as MILP in Gurobi. Section 5.4 shows the benefits of *PRIO* compared to *PREF* clustering and *RAND* clustering. Finally, Sect. 5.5 discusses effects of different levels of teaming requirements.

5.1 Data

For the experiments, we use input data from a German university hospital. Available data includes the different shift types and working times of physical therapists. Some therapists work part-time while other therapists work full-time and require a lunch break. Therapists in the hospital work in four large teams according to different wards and their treatment requirements. A team working in the neurology ward, where synchronization is required, consists of up to ten physical therapists. Up to 100 daily patients might be treated. The university hospital is a maximum care provider. To test the algorithm for different smaller care settings and situations, we vary the number of available therapists and patients in several different instances. We have access to the patient prioritization rules of the hospital. Additionally, we have access to the different duration rules of potential treatments. Between five percent and 15 percent of patients need treatments with teaming requirements. We divide the day into 5-min periods, where time period 1 equals the start time of the earliest time window. Patient availability ranges from very short time windows in the morning, before a patient is discharged, to time windows spanning the whole day, when no physician visits are planned. We assume for all instances that preference adherence for lead and support visits is equally important. Therefore, we set $w^{Lead} = w^{Support} = 1$. Usually, in a hospital setting, treating patients is seen as more important than adhering to therapist preferences. Therefore, we set $w^{Missed} = |V|$ for the different instances.

5.2 Optimal solution on exemplary data using the branch-and price approach

To further improve the understanding of the problem setting, we show input data and results for a small example instance we solved to optimality using the branch-and-price without clustering.

Table 2 shows the patient data, for the test instance with 30 patients and three therapists. Time windows and treatment durations are given in periods. The first four patients have the same time window and treatment durations. Given the three therapists, only three of the four patients can be scheduled, with the first patient remaining unscheduled due to the *low* priority. Three patients require teaming for their treatment. All three therapists are available from period zero to period 99 with a lunch break from period 54 to 59 which is modelled as a visit in $\boldsymbol{V^{SP}}$. Time windows and treatment durations allow for all patients, but the first patient, to be scheduled. Since patient priority is more important than preference adherence, all seven remaining patients with a *low* priority must be scheduled in an optimal schedule. For preference adherence, 3 Therapists $\times$ 33 Visits $=$ 99 Possibilities for scheduling are generated randomly. Of these 99 possibilities, 35 combinations are *beneficial*, 35 combinations are *neutral*, and 29 combinations are *detrimental*.

Figure 4 shows the resulting optimal schedule for the three therapists *T1*, *T2*, and T3 and time periods in steps of 5. Visits for patients 5, 13, and 23 are completely synchronized, as required for all our instances. While no travel time is necessary for some patients (e.g., patients 5 and 7), as they are situated in the same wardroom, usually a travel time between one to three periods is necessary. Larger gaps than three periods are due to time window restrictions or potential slack. No detrimental visit must be scheduled for any of the therapists. Most visits are *beneficial* for therapist satisfaction in their therapist schedule. However, nine visits are *neutral* for satisfaction in the optimal result. Neither of the *neutral* visits has a therapist available for whom the respective visits would be *beneficial*.

In the next section, we test and discuss the performance of the label correcting algorithm and the randomized greedy heuristic compared to an implementation of the SP as MILP.

5.3 Performance of the label correcting algorithm and the randomized greedy heuristic

Experiments are performed with six different instances ranging from 50 patients and five therapists up to 100 patients and ten therapists. All instances in this section include teaming requirements for ten percent of the patients. For performance comparisons, we solve the problems until we reach the lower bound in the root node.

Table 2 Patient data for an instance with 30 patients and 3 therapists

Time window						Time window					
Patient	Start	End	Duration	Priority	Teaming	Patient	Start	End	Duration	Priority	Teaming
1	1	3	3	Low	–	16	39	66	6	Medium	–
2	1	3	3	Medium	–	17	42	54	4	High	–
3	1	3	3	Medium	–	18	48	66	3	Medium	–
4	1	3	3	High	–	19	51	72	4	Low	–
5	3	12	5	Low	Yes	20	60	78	6	Medium	–
6	6	18	9	Medium	–	21	60	78	3	High	–
7	6	18	5	Low	–	22	60	78	8	High	–
8	6	42	6	Low	–	23	66	75	9	Medium	Yes
9	18	24	3	Medium	–	24	72	84	6	Low	–
10	18	48	9	Medium	–	25	72	98	4	Low	–
11	24	54	5	High	–	26	75	90	5	Medium	–
12	30	48	8	High	–	27	78	96	4	High	–
13	30	54	7	Medium	Yes	28	84	99	6	Low	–
14	30	54	6	High	–	29	84	99	8	Medium	–
15	36	66	3	High	–	30	90	96	3	Medium	–

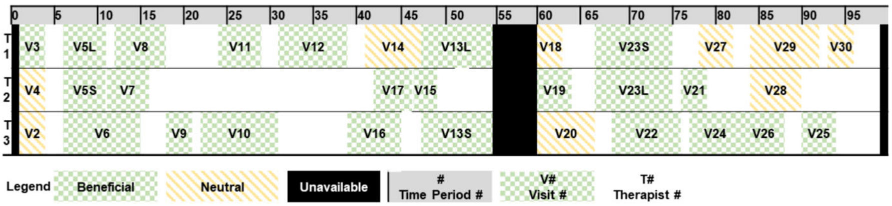

Fig. 4 Optimal schedule for the example instance

We set a runtime limit of two hours. To compare the label correcting algorithm and the randomized greedy heuristic, we implemented the pricing problem as a MILP in Gurobi and solved all individual SPs to optimality. *PREF* visit clustering, which is known from literature, is used in this section. For all solution methods, i.e., implementation as a MILP, label correcting algorithm and label correcting algorithm in combination with the randomized greedy heuristic, we set the cluster size to ten. A cluster size of ten showed the best tradeoff between solution quality and runtime in preliminary testing. Larger cluster sizes might lead to disjunct sets for some instances while smaller cluster sizes lead to more visits which are added to all therapists, which increases processing times again.

Table 3 shows the performance results for the different SP methods (*M*), i.e., the MILP formulation (*IP*), using the label correcting algorithm (*LA*) alone, and using the label correcting algorithm and the randomized greedy heuristic in combination (*GR*). $|\boldsymbol{E}|/\ |\boldsymbol{T}|/\ |\boldsymbol{V}|$ gives the number of therapists/patients/visits. *SUM* is the total runtime until the root node is solved. *TOT* informs about the total runtime for collective SPs. *AVG, MIN,* and *MAX* give the average per subproblem and iteration, minimum and maximum times of the SPs. All processing times are given in seconds. The number of column generation iterations, i.e., how often subproblems were used to generate new columns, is given by *ITER*. *COL(GR)* shows how many columns were generated for the different approaches. For the combination between randomized greedy heuristic and label correcting algorithm, the number in brackets shows how many columns were generated by the randomized greedy heuristic.

Time spent generating columns in all instances nearly equals the total processing time, i.e., the RMP is not a bottleneck for the problem. The MILP formulation is unable to reach the lower bound of the root node for the four largest instances. The label correcting algorithm is faster than the MILP implementation. It reaches *LB* in all tested instances. Combining the randomized greedy heuristic and the label correcting algorithm improves processing times substantially. The label correcting algorithm alone and in combination with the randomized greedy heuristic produce significantly more columns, compared to the MILP (for instances where the MILP reaches the lower bound). Usually, using the randomized greedy heuristic with the label correcting algorithm leads to more SP iterations. For the combination of the two algorithms, the randomized greedy heuristic generates the (vast) majority of columns for all instances. The label correcting algorithm is mainly necessary to prove optimality. This explains the difference in total processing times and the maximum observed time for individual SPs. The label correcting algorithm requires the most

Table 3 Performance comparisons between MILP, label correcting algorithm and randomized greedy heuristic

$\lvert E\rvert/\lvert T\rvert/\lvert V\rvert$	M	LB	SUM	TOT	AVG	MIN	MAX	ITER	COL(GR)
5/50/55	IP	6789	33.04	32.93	0.47	0.02	9.11	14	65
	LA	6789	4.44	4.40	0.40	0.33	0.51	11	380
	GR	6789	3.69	3.61	0.24	0.09	0.46	15	595(558)
6/60/66	IP	4710	2188.01	2187.56	14.02	0.05	526.03	26	146
	LA	4710	58.87	58.67	1.73	1.01	5.05	34	735
	GR	4710	34.22	33.97	1.10	0.08	1.77	31	935(616)
7/70/77	IP	–	>7200	>7200	>7,200	>7200	>7200	–	0
	LA	6258	697.18	696.83	31.67	9.51	74.96	22	1,289
	GR	6258	505.74	505.30	16.30	0.17	91.57	31	1,500 (1118)
8/80/88	IP	–	>7200	>7200	–	12.41	>7200	–	0
	LA	10,319	2263.97	2263.57	133.15	26.96	462.01	17	1164
	GR	10,319	1204.23	1203.73	42.99	0.14	42.99	28	1235(961)
9/90/99	IP	–	>7200	>7200	–	48.96	>7200	–	0
	LA	12,995	2559.25	2558.43	159.90	45.73	748.59	16	1308
	GR	12,995	380.47	379.93	17.27	0.22	56.13	22	1673(1308)
10/100/110	IP	–	>7200	>7200	51.83	0.06	2480.48	17	158
	LA	19,833	531.48	531.03	24.14	14.98	136.29	22	1006
	GR	19,833	436.14	435.45	10.62	0.17	21.50	41	1475(1176)

time when many partial schedules which are not pareto-dominated must be extended. The randomized greedy heuristic is most helpful for early pricing problems. For later pricing problem runs, fewer possibilities for new schedules exist, reducing the processing times for the label correcting algorithm. This can be seen by comparing the maximum SP runtime for the label correcting algorithm alone and the combination with the randomized greedy heuristic. *MAX* for *LA* exceeds *MAX* for *GR* for most instances.

5.4 Comparison of clustering approaches

Next, we compare *PREF* visit clustering from literature with *PRIO* visit clustering and *RAND* visit clustering. We assume teaming requirements for ten percent of the patients and use a cluster size of ten. We use the best performing method to solve the SP, i.e., randomized greedy heuristic followed by the label correcting algorithm. We run the instances without time limits until the final solution is reached. Table 4 shows the results. The second column shows the clustering method (*M*), The next three columns show the objective function value (*OBJ*), the number of unscheduled visits (*US*) and the total preference penalty (*TP*). The next six columns have similar names as the headings in Table 3. The last three columns show the number of branched nodes (*BN*), Nodes with time window branching (*TW*) and the depth of the final solution in the branching tree (*SD*).

The randomized greedy heuristic provides most of the columns added to the RMP for all three clustering methods. No pattern between the three clustering methods can be detected comparing the number of generated columns, necessary GC iterations, necessary branching nodes, solution depth and processing times for pricing problems or total processing times. There is a general tendency that larger instances require more processing time and more branching nodes for all three visit clustering approaches. However, instance *10/100/110* requires less processing time for all three tested clustering approaches than *9/90/99*. This means necessary processing times is not only dependent on the number of patients and therapists, but also on the pattern of schedules in pricing problems. Combined with visit clustering, time window branching (*TW*) is very effective. For most instances *TW* equals *BN*, i.e., only time window branching was necessary to arrive at the optimal solution. Minor exceptions can be seen for *PREF* in instance *5/50/55* and *RAND* in instance *6/60/66* where $0/1$ branching is necessary once. A more significant exception is instance *9/90/99*. Here several $0/1$ branches are necessary. Comparing solution quality of the three clustering approaches, *RAND* is not suitable for preference adherence. For preference penalty score *TP,* a negative value is better than a positive value. *RAND* is outperformed substantially by *PRIO* and *PREF.* For the instance *6/60/66*, *RAND* provides a total preference score which equals a detrimental result for therapist satisfaction. For preference adherence *PREF* outperforms *PRIO*. However, *PRIO* still provides very good results for preference adherence. For the objective function value *OBJ* and the number of unscheduled visits *US*, *PRIO* delivers consistently good results and a low number of unscheduled visits. *PREF* and *RAND* show high variation compared to the *OBJ* of *PRIO*. To summarize, *PRIO* is the most consistent clustering approach for the combined objective of minimizing unscheduled visits accounting for priority and minimizing preference adherence penalties. *PREF* might be favored if preference adherence is the most important objective.

5.5 Effects of different levels of synchronization

In this section, effects of different realistic teaming requirement levels are examined. We use *PRIO* visit clustering, as it showed the best results, considering both objectives, in Sect. 5.4. In previous sections, teaming requirements for ten percent of the patients were assumed. In this section results for the same instance sizes are compared to teaming requirements of five percent and 15 percent. If the exact percentage of the teaming requirement level is not integer, we round up to the next integer value for the number of visits. SPs are solved with the randomized greedy heuristic and the label correcting algorithm. No runtime limit was specified. Table 5 shows the results. $|\boldsymbol{E}|/\ |\boldsymbol{T}|/\ |\boldsymbol{V}|/\ \boldsymbol{S}$ shows the number of therapists/patients/visits/synchronization level. Other column headings are the same as for Table 4, except for *UT*. *UT* shows the number of unscheduled appointments with teaming requirements.

For most tested instances, a higher teaming requirement level leads to a higher total processing time *SUM*. Here, most time is spent in the pricing problems. Exceptions are the instances with 60 patients, where five percent synchronization

Table 4 Results for instances with ten percent synchronization

$\|E\|/\|T\|/\|V\|$	M	OBJ	US	TP	SUM	TOT	AVG	ITER	COL	GR	BN	TW	SD
5/50/55	PRIO	1237	4	−28	16.78	16.39	0.32	51	804	765	5	5	5
	PREF	6790	10	−30	61.00	58.72	0.35	166	2211	2003	21	20	9
	RAND	2358	5	−7	15.91	15.49	0.29	52	822	767	5	5	5
6/60/66	PRIO	5976	7	−30	30.93	30.27	0.55	55	587	388	7	7	7
	PREF	4777	10	−41	959.57	947.96	180	526	2983	1506	60	60	5
	RAND	5942	6	2	260.54	256.88	0.92	278	1516	1028	24	23	6
7/70/77	PRIO	8123	6	−30	4271.34	4256.90	7.00	608	4239	2681	41	41	4
	PREF	6258	7	−56	4546.34	4536.61	15.86	286	4588	2415	17	17	5
	RAND	8380	9	−13	1562.96	1539.14	2.09	736	5731	4174	67	67	7
8/80/88	PRIO	2943	6	−49	14,592.73	14,490.98	7.60	1907	6711	3211	253	253	5
	PREF	12,868	7	−68	14,043.73	14,021.16	35.95	390	4851	2885	40	40	3
	RAND	13,192	10	−8	357.10	351.66	2.62	134	3083	2979	18	18	16
9/90/99	PRIO	12,615	8	−57	15,260.8	14,891.29	6.57	2267	11422	9308	303	295	5
	PREF	12,998	12	−70	28,083.17	27,972.41	24.39	1147	8618	3319	136	127	9
	RAND	12,460	6	−14	22,752,40	22,532.51	16.32	1381	14,287	11,695	206	205	14
10/100/110	PRIO	19,510	8	−70	10,529.28	10,372.28	8.23	1261	8526	6357	175	175	29
	PREF	19,945	12	−75	9831.16	9,792.12	16.68	587	5762	3771	82	82	14
	RAND	23,421	9	−9	4656.89	4,527.97	5.23	865	8599	6599	134	134	23

Table 5 Results with different levels of synchronization

$\|E\|/\|T\|/\|V\|/S$	OBJ	US	UT	TP	SUM	TOT	AVG	ITER	COL	GR	BN	TW	SD
5/50/53/5%	3257	5	0	−29	7.75	7.55	0.24	31	487	483	4	4	4
5/50/55/10%	1237	4	0	−28	16.78	16.39	0.32	51	804	765	5	5	5
5/50/58/15%	2412	4	1	−24	271.35	265.18	0.71	376	3,258	2,650	30	30	7
6/60/63/5%	5644	6	1	−26	33.43	32.80	0.41	80	682	664	11	11	3
6/60/66/10%	5976	7	0	−30	30.93	30.27	0.55	55	587	388	7	7	7
6/60/69/15%	6245	7	1	−34	4504.60	4256.85	1.56	2734	14,071	6,203	226	221	4
7/70/74/5%	6039	7	1	−29	485.52	482.86	2.36	205	1925	1448	12	12	4
7/70/77/10%	8123	6	1	−30	4271.34	4256.90	7.00	608	4239	2681	41	41	4
7/70/81/15%	8545	6	2	−41	49,215.22	47,497.55	8.42	5641	39,823	28,852	495	463	5
8/80/84/5%	7434	5	2	−42	420.83	418.66	2.95	142	1444	1108	12	12	6
8/80/88/10%	2943	6	0	−49	14,592.73	14,490.98	7.60	1907	6711	3211	253	253	5
8/80/92/15%	7581	5	1	−55	42,327.75	40,677.41	10.73	3791	45,737	44,214	445	444	6
9/90/95/5%	11,916	6	0	−54	184.03	182.91	3.73	49	1055	1020	6	6	6
9/90/99/10%	12,615	8	2	−57	15,260.8	14,891.29	6.57	2267	11,422	9,308	303	295	5
9/90/104/15%	9,237	4	1	−69	7999.32	7797.51	8.34	935	15,757	14,710	121	121	32
10/100/105/5%	11,375	7	1	−70	739.81	733.21	6.67	110	1656	1467	13	13	12
10/100/110/10%	19,510	8	1	−70	10,529.28	10,372.28	8.23	1261	8526	6357	175	175	29
10/100/115/15%	16,377	7	1	−68	75,072.06	74,811.44	42.10	1777	13,563	7371	169	167	13

requires more time than ten percent synchronization and 90 patients where ten percent synchronization requires more time than 15 percent synchronization. This can be explained by schedule patterns, where the randomized greedy heuristic has more difficulties to find additional columns and the label correcting algorithm provides many columns. However, 15 percent synchronization for 80 patients shows that even when few runs of the label correcting algorithm are necessary, processing times can be extensive, if one individual SP has many non-dominated partial paths. Higher levels of synchronization usually require more SP runs and leads to more generated columns. It also leads to larger branching trees and an increase in depth in the branching tree for the optimal solution. While time window branching remains very effective, it is not enough to guarantee feasible solutions for most of the tested instances with 15 percent synchronization. Here, integer branching is necessary.

For *OBJ*, slight increases with higher levels of teaming requirement can be observed. It is possible that results with a higher level of synchronization are better than results with lower levels of synchronization. This can be explained by the visit clustering and more unclustered visits which are assigned to all therapists after each therapist received ten cluster visits, i.e., more scheduling flexibility. An example is the instance *5/50/53* with three support visits, i.e., five percent synchronization compared to *5/50/55* with ten percent synchronization, i.e., five visits. The latter instance yields the lower, i.e., the better objective function value. In the tested instances, patients with teaming requirements are spread across all different treatment priority classes. Lead and support visits represent all possible preference adherence classes. *UT* shows that most patients with teaming requirements are scheduled for all instances. The total number of unscheduled patients does also not change, with increasing teaming requirements. There are differences for preference adherence. For all instance sizes but two, a higher level of synchronization leads to better preference adherence scores. For instances with 50 patients and with 100 patients, a higher level of synchronization leads to a worse preference adherence score. However, here *OBJ* is better with a higher synchronization level. Scheduling patients accounting for priority is the main objective, minimizing preference adherence penalties is the secondary objective. Therefore, a worse preference adherence score is not unexpected for these two instances.

6 Summary and outlook

We describe the challenging workforce situation of physical therapy departments and the resulting difficulties of treating patients accounting for their priority while adhering to therapist satisfaction objectives. We derive a complex operational planning problem that is not considered in the literature yet. We propose a VRPTW with synchronization constraints which minimizes penalties for missing visits accounting for their priority and minimizes preference scores for lead and support tasks. In the model, it is impossible to enumerate all possible schedules for all therapists. Therefore, we develop a branch-and-price approach where pricing problems add promising schedules to an RMP. A randomized greedy heuristic is developed to speed up processing times. Additionally, a visit clustering approach from literature is included and compared to a more problem specific priority-based clustering approach and a random clustering. Experiments show the high effectiveness of the randomized greedy heuristic for the processing time and the number of generated columns. We also show that the more problem specific priority-based clustering procedure outperforms the clustering known from literature and random clustering. Analyzing different levels of teaming requirements, i.e., synchronization, we show that clustering is very effective in reducing the number of necessary $0/1$-branches and we show clustering and the randomized greedy heuristic in the subproblems remain effective even when more visits must be synchronized.

Based on this contribution, future research possibilities can be derived. We are the first paper to focus on workforce shortages and therapist satisfaction in rehabilitation therapy while including preferences for lead and support tasks. Similar problem settings might be necessary in different industries and sectors where individual employees, while generally on the same hierarchy level, might take responsibility for a task while others handle support tasks. Examples are agile project management (e.g., in software development) or task management (e.g., in police departments). In our problem setting, we had access to deterministic treatment time information and fixed treatment time windows. In future research, stochastic settings might be considered, e.g., stochastic treatment times or uncertain patient availability time windows. Another possibility to model preferences for lead or support tasks might be to assume therapist-specific treatment times, e.g., depending on the experience of the lead therapist. Further, different therapist qualifications, additional fairness aspects or additional resources might be necessary in other problem settings. Finally, meta-heuristics for larger problem sizes or improvements to parts of the branch-and-price might be future research directions.

Appendix Appendix A: Pseudocode for the randomized greedy heuristic

1	**Input:** Sets, parameters, dual variables, and decision variable t_i of $SP(e)$, RMP
2	**Initialize:** $SortedVisits = \emptyset$, $RCL = \emptyset$, $AddedColumns = 0$, $FoundColumns$ = **True**, $Iterations = 20$
3	**while** $AddedColumns < Iterations$ **and** $FoundColumns$ **begin**
4	$SortedVisits$ = sorted list of lead and support visits in non-decreasing order according to preference cost
5	$Schedule = [0, n]$, $NegativeReducedCost$ = **True**
6	**while** $NegativeReducedCost$ **begin**
7	$FoundColumns$ = **False**
8	RCL = list of the first three items in $SortedVisits$
9	k = random pick from RCL
10	**if** partner of k **not in** $Schedule$
11	**for** l **in** a list of possible insertion points within $Schedule$
12	**if** $F_k > F_{Schedule[l-1]} + L_{Schedule[l-1]} + D_{Schedule[l-1],k}$ **and** k **not in** $Schedule$
13	$t_k = \mathbf{max}\{B_k, t_{Schedule[l]} + L_{Schedule[l]} + D_{Schedule[l],k} - 1\}$
14	$Schedule = Schedule$ until $l + k + Schedule$ from l
15	**for** o **in** the list of successor visits of k in the current $Schedule$
16	**update** t_o like t_k in **line 13**
17	**if** $t_o \geq F_0$
18	$Schedule = Schedule$ without k
19	**break**
20	**end if**
21	**if** o = the last item in the list of successor visits of k in the current $Schedule$
22	**if** $Schedule$ has negative reduced cost according to (8b)
23	**add** $Schedule$ to RMP
24	**remove** k from $SortedVisits$
25	$AddedColumns += 1$
26	$FoundColumns$ = **True**
27	**end if**
28	**else**
29	$NegativeReducedCost$ = **False**
30	**end else**
31	**end if**
32	**if** l = the list of possible insertion points within $Schedule$ **and** k **not in** $Schedule$
33	**remove** k from $SortedVisits$
34	**end if**
35	**end for**
36	**end if**
37	**end while**
38	**end while**

Appendix B: Pseudocode for the clustering approaches

Input: Sets and parameters of the master problem, $ClusteringApproaches = [PREF, PRIO]$
Initialize: $Cluster(e) = \emptyset$, $ClusterSize(e) = 0$, $MaximumClusterSize = n$, $Clusters$ = $[Cluster(e)\ \forall e \in \boldsymbol{E}]$
If $ClusteringApproach$ **is** $PREF$
    While the list of visits $i \in \boldsymbol{V}$ **not** $\emptyset$ **begin**
        i = random pick from $\boldsymbol{V}$
        $\boldsymbol{V} = \boldsymbol{V}$ without i
        $SortedTherapists$ = list of therapists $e \in \boldsymbol{E}$ sorted according to preference for $i \in V$ in non-increasing order
        for $\boldsymbol{e} \in SortedTherapists$
            If $ClusterSize(e) < MaximumClusterSize$
                $Cluster(e) = Cluster(e) \cup \{i\}$
                $ClusterSize(e) += 1$
                break
            end if
        end for
        If i **not in** one of the clusters in $Clusters$
            Add i **to all** clusters in $Clusters$
        end if
    end while
end if
If $ClusteringApproach$ **is** $PRIO$
    While the list of visits $i \in \boldsymbol{V}$ **not** $\emptyset$ **begin**
        $SortedVisits$ = list of visits $i \in \boldsymbol{V}$ sorted according to cost of missing them in non-increasing order
        i = first item in $SortedVisits$
        $\boldsymbol{V} = \boldsymbol{V}$ without i
        $SortedTherapists$ = list of therapists $e \in \boldsymbol{E}$ sorted first according to their cluster size $ClusterSize(e)$ in non-decreasing order and then according to preference for i in non-increasing order
        for $\boldsymbol{e} \in SortedTherapists$
            If $ClusterSize(e) < MaximumClusterSize$
                $Cluster(e) = Cluster(e) \cup \{i\}$
                $ClusterSize(e) += 1$
                break
            end if
        end for
        If i **not in 1** cluster in $Clusters$
            Add i **to all** cluster in $Clusters$
        end if
    end while
end if

Funding Open access funding provided by Technical University of Denmark. This research did not receive external funding.

Data and materials availability The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflict of interest There is no conflict of interest in the study.

Ethics approval, consent to participate and consent to publication The study does not require an ethics approval, consent to participate and consent to publication.

References

Ahmadi-Javid A, Jalali Z, Klassen KJ (2017) Outpatient appointment systems in healthcare: a review of optimization studies. Eur J Oper Res 258:3–34. https://doi.org/10.1016/j.ejor.2016.06.064

Ait Haddadene SR, Labadie N, Prodhon C (2016) A GRASP × ILS for the vehicle routing problem with time windows, synchronization and precedence constraints. Expert Syst Appl 66:274–294. https://doi.org/10.1016/j.eswa.2016.09.002

Bundesagentur für Arbeit (2022) Fachkräfteengpassanalyse 2021: Blickpunkt Arbeitsmarkt | Mai 2022. https://statistik.arbeitsagentur.de/DE/Navigation/Footer/Top-Produkte/Fachkraefteengpassanalyse-Nav.html. Accessed 14 November 2022

Bureau of Labor Statistics, U.S. Department of Labor (2022) Occupational Outlook Handbook, Physical Therapists. https://www.bls.gov/ooh/healthcare/physical-therapists.htm. Accessed 23 August 2023

Burke EK, de Causmaecker P, Vanden Berghe G, van Landeghem H (2004) The state of the art of nurse rostering. J Sched 7:441–499. https://doi.org/10.1023/B:JOSH.0000046076.75950.0b

Cheang B, Li H, Lim A, Rodrigues B (2003) Nurse rostering problems—a bibliographic survey. Eur J Oper Res 151:447–460. https://doi.org/10.1016/S0377-2217(03)00021-3

Chien C-F, Tseng F-P, Chen C-H (2008) An evolutionary approach to rehabilitation patient scheduling: a case study. Eur J Oper Res 189:1234–1253. https://doi.org/10.1016/j.ejor.2007.01.062

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. Oper Res Health Care 13–14:1–22. https://doi.org/10.1016/j.orhc.2017.06.001

Cloutier O, Felusiak L, Hill C, Pemberton-Jones EJ (2015) The importance of developing strategies for employee retention. J Leadersh Acc Ethics 12:119–129

Desaulniers G, Madsen OBG, Ropke S (2014) The vehicle routing problem with time windows. In: Toth P, Vigo D (eds) Vehicle routing: problems, methods, and applications, 2nd edn. SIAM, Philadelphia, Pa, pp 119–159

Dohn A, Kolind E, Clausen J (2009) The manpower allocation problem with time windows and job-teaming constraints: a branch-and-price approach. Comput Oper Res 36:1145–1157. https://doi.org/10.1016/j.cor.2007.12.011

Dohn A, Rasmussen MS, Larsen J (2011) The vehicle routing problem with time windows and temporal dependencies. Networks 58:273–289. https://doi.org/10.1002/net.20472

Drexl M (2012) Synchronization in vehicle routing—a survey of VRPs with multiple synchronization constraints. Transp Sci 46:297–316. https://doi.org/10.1287/trsc.1110.0400

Erhard M, Schoenfelder J, Fügener A, Brunner JO (2018) State of the art in physician scheduling. Eur J Oper Res 265:1–18. https://doi.org/10.1016/j.ejor.2017.06.037

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 44:216–229. https://doi.org/10.1002/net.20033

Fikar C, Hirsch P (2017) Home health care routing and scheduling: a review. Comput Oper Res 77:86–95. https://doi.org/10.1016/j.cor.2016.07.019

Frey CM, Jungwirth A, Frey M, Kolisch R (2023) The vehicle routing problem with time windows and flexible delivery locations. Eur J Oper Res 308:1142–1159. https://doi.org/10.1016/j.ejor.2022.11.029

Gartner D, Frey M, Kolisch R (2018) Hospital-wide therapist scheduling and routing: exact and heuristic methods. IISE Transact Healthc Syst Eng 8:268–279. https://doi.org/10.1080/24725579.2018.1530314

Gélinas S, Desrochers M, Desrosiers J, Solomon MM (1995) A new branching strategy for time constrained routing problems with application to backhauling. Ann Oper Res 61:91–109. https://doi.org/10.1007/BF02098283

Griffith JD, Williams JE, Wood RM (2012) Scheduling physiotherapy treatment in an inpatient setting. Oper Res Health Care 1:65–72. https://doi.org/10.1016/j.orhc.2012.08.001

Gupta D, Denton B (2008) Appointment scheduling in health care: challenges and opportunities. IIE Trans 40:800–819. https://doi.org/10.1080/07408170802165880

Gurobi Optimization L (2022) Gurobi Optimizer Reference Manual. https://www.gurobi.com

Huynh N-T, Huang Y-C, Chien C-F (2018) A hybrid genetic algorithm with 2D encoding for the scheduling of rehabilitation patients. Comput Ind Eng 125:221–231. https://doi.org/10.1016/j.cie.2018.08.030

Jungwirth A, Desaulniers G, Frey M, Kolisch R (2021) Exact branch-price-and-cut for a hospital therapist scheduling problem with flexible service locations and time-dependent location capacity. INFORMS J Comput. https://doi.org/10.1287/ijoc.2021.1119

Kling S, Kraul S, Brunner JO (2024) Customized GRASP for rehabilitation therapy scheduling with appointment priorities and accounting for therapist satisfaction. Or Spectr. https://doi.org/10.1007/s00291-023-00742-y

Lau B, Skinner EH, Lo K, Bearman M (2016) Experiences of physical therapists working in the acute hospital setting: systematic review. Phys Ther 96:1317–1332. https://doi.org/10.2522/ptj.20150261

Ogulata SN, Koyuncu M, Karaskas E (2008) Personnel and patient scheduling in the high demanded hospital services: a case study in the physiotherapy service. J Med Syst 32:221–228. https://doi.org/10.1007/s10916-007-9126-4

Qiu H, Wang D, Yin Y, Cheng TCE, Wang Y (2022) An exact solution method for home health care scheduling with synchronized services. Nav Res Logist (NRL) 69:715–733. https://doi.org/10.1002/nav.22044

Rasmussen MS, Justesen T, Dohn A, Larsen J (2012) The home care crew scheduling problem: preference-based visit clustering and temporal dependencies. Eur J Oper Res 219:598–610. https://doi.org/10.1016/j.ejor.2011.10.048

Schimmelpfeng K, Helber S, Kasper S (2012) Decision support for rehabilitation hospital scheduling. Or Spectr 34:461–489. https://doi.org/10.1007/s00291-011-0273-0

Soares R, Marques A, Amorim P, Parragh SN (2024) Synchronisation in vehicle routing: classification schema, modelling framework and literature review. Eur J Oper Res 313:817–840. https://doi.org/10.1016/j.ejor.2023.04.007

Tanoumand N, Ünlüyurt T (2021) An exact algorithm for the resource constrained home health care vehicle routing problem. Ann Oper Res 304:397–425. https://doi.org/10.1007/s10479-021-04061-9

Verband der Ersatzkassen (2022) Rahmenverträge. https://www.vdek.com/vertragspartner/heilmittel/rahmenvertrag.html. Accessed 12 April 2022

Vidal T, Laporte G, Matl P (2020) A concise guide to existing and emerging vehicle routing problem variants. Eur J Oper Res 286:401–416. https://doi.org/10.1016/j.ejor.2019.10.010

Zhao L, Chien C-F, Gen M (2018) A bi-objective genetic algorithm for intelligent rehabilitation scheduling considering therapy precedence constraints. J Intell Manuf 29:973–988. https://doi.org/10.1007/s10845-015-1149-y

Zimbelman JL, Juraschek SP, Zhang X, Lin VW-H (2010) Physical therapy workforce in the United States: forecasting nationwide shortages. PM R 2:1021–1029. https://doi.org/10.1016/j.pmrj.2010.06.015

Sebastian Kling has worked as a research associate at the Faculty of Business and Economics at the University of Augsburg from April 2019 to December 2023. His research focuses on operational planning problems in an acute care setting. After his scientific career, he works as a project manager for digitalization topics at Linde plc in Germany.

Sebastian Schiffels is a professor of Digital Health & Medical Decision Making at the University of Augsburg. Previously, he was an Assistant Professor at Lancaster University in Leipzig and completed his Ph.D. and habilitation at the Technical University of Munich. He specializes in (Behavioral) Operations Management and (Healthcare) Analytics. As a member of the Center for Interdisciplinary Health Research, he contributes to advancing data-driven decision-making in healthcare and business.

Jens O. Brunner has been permanently appointed as Professor of Decision Science in Healthcare at the Department of Technology, Management, and Economics at the Technical University of Denmark. He is heading the Center of Excellence in Healthcare Operations Management in Regions Zealand. He was appointed as Professor of Health Care Operations/Health Information Management at the Faculty of Business and Economics at the University of Augsburg until March 2023. From 2013 to 2020 he was (co-) director of the University Center for Health Care at Klinikum Augsburg (UNIKAT). Professor Brunner's

research interests center on the design and analysis of service systems using quantitative methods. A special focus is on processes in healthcare.

Authors and Affiliations

Sebastian Kling[1] · **Sebastian Schiffels**[1] · **Jens O. Brunner**[1,2,3]

✉ Jens O. Brunner
jotbr@dtu.dk

Sebastian Kling
sebastian.kling@uni-a.de

Sebastian Schiffels
sebastian.schiffels@uni-a.de

1 Present Address: Faculty of Business and Economics, University of Augsburg, Augsburg, Germany

2 Present Address: Department of Technology, Management, and Economics, Technical University of Denmark, Kongens Lyngby, Denmark

3 Center for Excellence in Healthcare Operations Planning, Next Generation Technology, Region Zealand, Denmark