J Sched
DOI 10.1007/s10951-013-0345-x

A sequential GRASP for the therapist routing and scheduling problem

Jonathan F. Bard · Yufen Shao · Ahmad I. Jarrah

Received: 28 February 2012 / Accepted: 8 August 2013

Abstract This paper presents a new model and solution methodology for the problem faced by companies that provide rehabilitative services to clinic and home-bound patients. Given a set of multi-skilled therapists and a group of geographically dispersed patients, the objective is to construct weekly tours for the therapists that minimize the travel, treatment, and administrative costs while ensuring that all patients are seen within their time windows and that a host of labor laws and contractual agreements are observed. The problem is complicated by three factors that prevent a daily decomposition: (i) overtime rates kick in only after 40 regular hours are worked during the week, (ii) new patients must be seen by a licensed therapist on their first visit, and (iii) for some patients only the frequency and not the actual days on which they are to be seen is specified. The problem is formulated as a mixed-integer linear program but after repeated attempts to solve small instances with commercial software failed, we developed an adaptive sequential greedy randomized adaptive search procedure. The phase I logic of the procedure builds one daily schedule at a time for each therapist until all patients are routed. In phase II, several neighborhoods are explored to arrive at a local optimum. Extensive testing with both real data provided by a U.S. rehab company and datasets derived from them demonstrated the value of the purposed procedure with respect to current practice. The results indicated that cost reductions averaging over 18.09 % are possible.

J. F. Bard (✉)
Graduate Program in Operations Research and Industrial Engineering, The University of Texas, Austin, TX 78712-1591, USA
e-mail: jbard@mail.utexas.edu

Y. Shao
ExxonMobil Upstream Research, 3120 Buffalo Speedway, Houston, TX 77098, USA
e-mail: shshaoao@gmail.com

A. I. Jarrah
Department of Decision Sciences, School of Business, The George Washington University, Funger 415, Washington, DC 20052, USA
e-mail: jarrah@gwu.edu

Keywords GRASP · Therapist routing · Midterm scheduling · Overtime · m-TSP · Home healthcare

List of symbols

Indices and sets

i, j	Index for patients
k	Index for therapists
d	Index for days
$o(k), d(k)$	Origin, destination of therapist k
K	Set of all therapists; $K = \{\text{PTs}, \text{PTAs}\}$
$K(i, d)$	Set of therapists that can see patient i on day d
I	Set of patients to be seen over the planning horizon, $I = I_{\text{FIX}} \cup I_{\text{FLEX}}$
$IC(i, d, k)$	Set of patients that can either precede or follow patient i in a schedule for therapist k on day d
$IK(k, d)$	Set of patients that can be seen by therapist k on day d
$DK(k)$	Set of days therapist k can be scheduled to work
$DU(i)$	Set of days in the planning horizon that patient i is to be treated (pattern is given)

Data and parameters

c_{ij}^k	Cost of traveling from patient i to patient j by therapist k

Published online: 27 August 2013

c_{id}^k	Cost of a visit by therapist k to see patient i on day d
s_{id}	Time required to provide treatment to patient i on day d (h)
a_{id}, b_{id}	Earliest, latest time treatment can begin for patient i on day d
τ_{ij}	Travel time between patients i and j (assumed symmetric)
e_{kd}, f_{kd}	Earliest start, latest finish time of therapist k on day d

Decision variables

x_{ijd}^k	1 if therapist k visits patients i and j in succession on day d, 0 otherwise
t_{id}	Time at which patient i begins to receive treatment from a therapist on day d

1 Introduction

In the service industry, the workforce accounts for the overwhelming majority of costs, so a key to remain competitive is effective personnel planning at both the strategic and tactical levels. This not only means minimizing operational costs, but also maintaining quality, improving productivity, and meeting or exceeding customer expectations. In healthcare delivery, the focus is invariably on the patient which implies a need for robust staffing. However, because of the difficulty in balancing demand for treatment with staff availability, skill levels, preferences and seniority, managers often overcompensate by employing a larger workforce than necessary (Stansfield et al. 2011). In hospitals, nurse managers typically staff for peak demand and hence incur excessive overhead during low occupancy periods. Achieving the proper balance requires a staffing model that admits adequate flexibility to meet changing demand on a short-term basis.

In this paper, we address the problem of providing rehabilitative services to patients located at clinics, hospitals, nursing homes, and other healthcare facilities scattered around a metropolitan area. We have singled out physical therapy for analysis but the results are generally applicable to all home healthcare organizations that provide most forms of therapeutic, rehabilitative, and palliative services using a "house call" business model. In fact, home healthcare has been a growing sector of the economy for more than two decades with over 10,000 organizations now operating in the U.S. alone. As the population ages and becomes more infirm, those with non-critical medical needs are increasingly seeking treatment at their place of residence (Begur et al. 1997).

The majority of physical therapists are independent agents who work for regional rehab companies that have long-term service contracts with healthcare facilities. At the beginning of the week, each therapist submits his desired work schedule indicating which days he is available, his corresponding time windows, and perhaps his preference for treatment locations. The supervisor assembles this information along with patient requirements and then attempts to construct weekly rosters for the staff. Because therapy is an elective procedure, demand is known with a high degree of certainty. New patients and appointment cancellations are handled on an ad hoc basis as the week unfolds.

Although there has been considerable work in personnel scheduling over the last 30 years, the difficulty of constructing weekly rosters cannot be overstated and is an ongoing source of frustration for managers. For our problem, the reasons are at least fivefold. The first concerns the patient type, which can be either *fixed* or *flexible*. The former has a fixed time window with little or no tolerance; that latter can be seen almost any time during the day. The mixture of the two defines a problem that has both a routing and scheduling aspect. The second is that the treatment days for flexible patients are not specified in advance, as in the case of the generic periodic routing problem (e.g., see Baldacci et al. 2011). What is known is only the total number of days and the set of feasible patterns; e.g., two consecutive days during the week. The third is that some flexible patients require multiple sessions each day that they are scheduled for treatment with a minimum number of hours between each session. The fourth is that all therapists who work 6 or more hours per day are entitled to a 1/2-h lunch break. Finally, new patients must be seen by a fully licensed therapist, as opposed to an assistant therapist, at their first visit.

These factors highly restrict the feasible region and make it nearly impossible for managers to construct cost-efficient rosters. To the best of our knowledge all scheduling in the industry is done manually and no commercial or academic software exists to support the process.

Accordingly, the purpose of this paper is to present a new model that captures the unique features of the therapist routing and scheduling problem (TRSP) and to describe a new algorithm for finding high-quality solutions. The model is designed to minimize the cost of meeting patient demand over the week while adhering to a variety of personal, physical, legal and administrative constraints associated with both patients and therapists. Although the TRSP can be formulated as a mixed-integer linear program, we found that even small instances are beyond the capability of exact methods. In previous work (Shao et al. 2012), we developed a parallel (two-phase) greedy randomized adaptive search procedure (GRASP) that performed well on most instances investigated but ran into trouble on those that were tightly constrained [for a bibliography as well as successful developments and applications of GRASP, see, for example, Boudia et al. (2006), Feo et al. (1991); Festa and Resende (2002, 2009a,b); Kontoravdis and Bard (1995)]. We now present an improved GRASP that contains several effective and generalizable components that were developed for constructing tours sequentially rather than in parallel.

The latter approach is believed to be the better for routing and scheduling problems because it considers more comprehensive neighborhoods. This is the basic motivation for the "cluster first, route second" strategy that is most commonly used to solve routing problems with heuristics (e.g., see Cordeau et al. 2002; Kontoravdis and Bard 1995; Laporte 2009). One of the major contributions of this paper with respect to Shao et al. (2012) is the confirmation that algorithms should be designed with the specifics of the problem in mind. For the TRSP in which feasibility and optimality highly depend on the distances between patients, it is much more efficient to create routes in sequence where patients who are close together are only assigned to the same therapist if time permits; we found that clustering patients at the same location is often suboptimal.

The other major contributions of our algorithm center on two novel features. The first is an innovative technique that aims to balance feasibility with optimality when selecting the next route to explore. The second is the design of an adaptive rule which enables us to trade off solution quality with solution diversity when selecting the next patient to schedule. In addition, both the construction phase and improvement phase of the sequential GRASP embody a host of unique constructs designed to enhance computational efficiency. These include new procedures for determining the candidate lists and computing the benefit functions, the introduction of slack blocks to facilitate the examination of neighborhoods, and methods for checking feasibility and updating the cost functions. It is also worth mentioning that our code is independent of commercial solvers, a critical consideration for some users; our parallel GRASP is built in part around CPLEX.

The work was done in conjunction with Key Rehab, a company that contracts out physical, occupational, and speech therapy services throughout the U.S. Midwest. The limited supply of therapists in this region coupled with increased demand has placed a strain on the company's ability to offer timely, low-cost treatment. On any day of the week, therapists often travel for several hours between multiple sites to treat patients, accruing unsustainable travel and overtime costs along the way.

In generic terms, the TRSP can be viewed as a multi-day, nonhomogeneous m-TSP with time windows. In the next section, we give a formal statement of the problem and discuss the related literature. In Sect. 3, the basic mixed-integer programming (MIP) model is presented followed in Sect. 4 by the details of the GRASP. In Sect. 5, we analyze five real datasets and demonstrate that cost reductions averaging 18.09 % with respect to current practice and 5.58 % with respect to our previously developed algorithm are achievable. In Sect. 6, we investigate a set of 80 randomly generated instances to gain a better understanding of how the sequential and parallel GRASP perform relative to each other. The results strongly favored the new approach with cost reductions averaging 16.30 % and runtime reductions averaging 43.73 %. Conclusions are drawn in Sect. 7 where some plans for future research are outlined.

2 Problem statement and background

We are given a set K of therapists consisting of fully licensed physical therapists (PTs) and physical therapist assistants (PTAs), and a set of patients I who are characterized as either fixed ($I_{\text{FIX}} \subseteq I$) or flexible ($I_{\text{FLEX}} \subseteq I$). Each therapist $k \in K$ is available on a subset of days $d \in D$ in the time interval $[e_{kd}, f_{kd}]$ and each patient $i \in I$ must begin treatment in the time interval $[a_{id}, b_{id}]$ on one or more days d. Full-timers generally work Monday through Friday for up to 8 h a day with e_{kd} = 8:00 a.m. and f_{kd} = 5:00 p.m. with the last patient scheduled no later than 4:30 p.m. In most cases, $[a_{id}, b_{id}]$ = [7:00 a.m., 4:45 p.m.] for $i \in I_{\text{FLEX}}$ and $a_{id} = b_{id}$ for $i \in I_{\text{FIX}}$ for all $d \in D$. Moreover, about 12 % of the flexible patients must be seen multiple times a day each day that they are scheduled for treatment. The number of sessions is specified as input along with the number of days per week (but not the specific days) that each $i \in I_{\text{FLEX}}$ is to be treated. A component of the decision process then is to select the weekly treatment pattern for patient $i \in I_{\text{FLEX}}$ from the set $P(i)$ of feasible patterns.

For each day d that therapist k is assigned a shift, his route begins at location $o(k)$ and ends at location $d(k)$, which are normally the same. If the shift is 6 or more hours, then a 1/2-h uncompensated lunch break must be provided between 11:00 a.m. and 1:00 p.m. The hourly wage rate w_k is uniquely specified and is paid for time spent seeing patients (s_{id} is the time required to provide service to patient i on day d) and for driving between patient locations. The latter is in addition to a mileage reimbursement. For the first and last appointments of the day, agreement is reached as to the number of minutes of drive time that will *not* be compensated while the therapist is traveling to and from his home base. This "allowance" varies by individual.

Therapists are also paid for the administrative time (denoted by adm_i for patient i) required to document treatments. It is preferable that the administrative functions be performed immediately after seeing a patient but this is not a requirement. As a rule of thumb, adm_i is 25 % of the treatment time for PTs and 20 % for PTAs. An important parameter that is also included in each contract concerns *productivity*, which is defined as the ratio of treatment time to treatment plus administrative time. Pay rates are different for treatment, driving, and administrative tasks. Daily mileage is reimbursed in accordance with the following scheme: 0–25

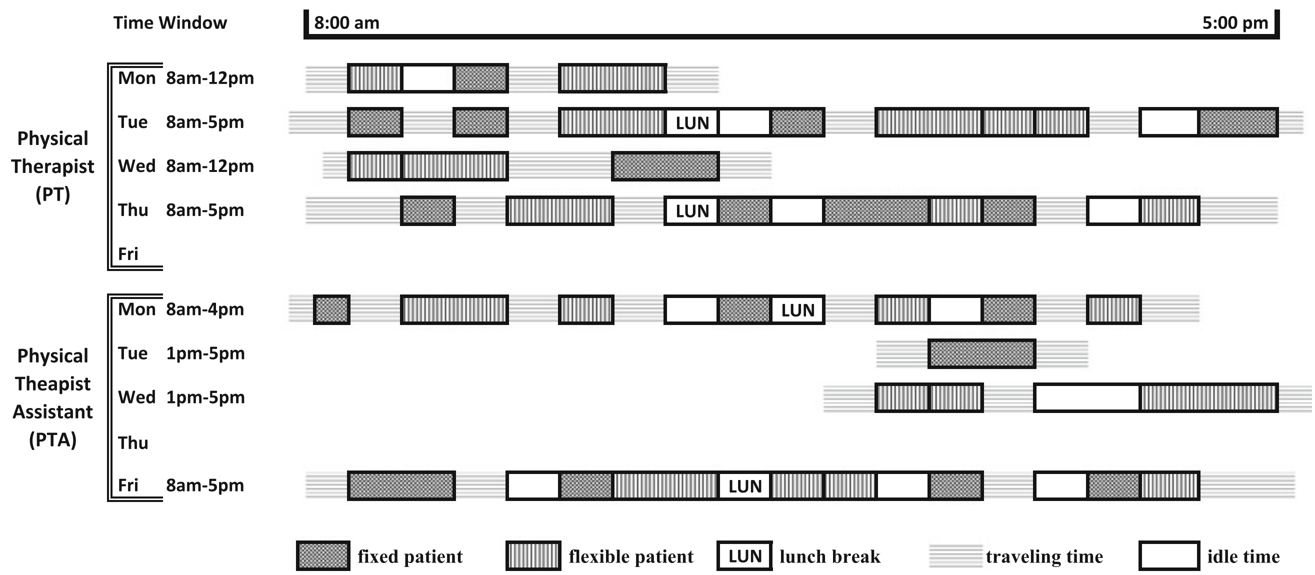


Fig. 1 Example of a weekly roster

miles = no reimbursement, 26–86 miles = \$0.28 per mile, and >87 miles = \$0.19 per mile—a nonconvex function which is difficult to handle. Since a therapist seldom drives over 87 miles during the day, the third segment is assigned a cost of \$0.28/mile instead of \$0.19/mile to ensure that the cost function is convex. The cost of each route is updated when a feasible solution is obtained to account for any overestimation.

Given this information, we can calculate the cost incurred by therapist k to travel from patient i to patient j, denoted by c_{ij}^k, and the cost of treating patient i, c_{id}^k, which depends on the time required for service s_{id} on day d. The travel time between two locations i and j is denoted by τ_{ij}. Overtime is paid on the total hours worked during the week above 40. The overtime rate is calculated as 150 % of the “average regular time rate” and is based on the actual treatment, driving, and administrative time accumulated by the therapist. Note that the non-reimbursed driving time associated with the negotiated allowance is not considered as working time, nor is the lunch break.

With this background in mind, the TRSP can be stated as follows. For each $k \in K$, construct a weekly schedule that minimizes the total treatment, travel, overtime, administration, and mileage costs of providing service to each $i \in I$ by specifying the weekly patterns for all flexible patients and by ensuring that the time windows of all patients and therapists are respected, that new patients are seen by a PT during their first treatment, and that route continuity is maintained for all therapists each day of the planning horizon. A sample roster for one PT and one PTA is depicted in Fig. 1.

2.1 Healthcare worker routing

An examination of the literature reveals that home healthcare research has mostly focused on nursing services. Cheng and Rich (1998) developed a network flow model for the routing and scheduling problem associated with treating patients in their home. To find solutions, they used a standard two-phase approach. Numerical results are reported for up to 4 nurses and 10 patients. Bertels and Fahle (2006) addressed a similar problem using a combination of linear programming, constraint programming, and metaheuristics to find solutions. Their principal goals were to develop a basic tool that was sufficiently flexible to adapt to various changes in the constraint structure of the problem and to show how to apply the different heuristics efficiently in a real-world setting.

Bennett and Erera (2011) also investigated a routing and scheduling problem in which patient requests arrive dynamically with periodic treatment requirements. They purposed a rolling horizon myopic planning approach for the single-nurse case with the objective of maximizing individual productivity. Solutions were found with a capacity-based insertion heuristic. Similar work is described by Eveborn et al. (2006).

2.2 Patient scheduling

A related line of research concerns patient scheduling. Gupta and Denton (2008) point out that timely access for patients is critical in realizing good medical outcomes and achieving a high level of customer satisfaction. Their model is aimed at scheduling primary and specialty care visits and elective surgeries in clinics and hospitals. An informative discussion is included on dealing with late cancellations and no-shows. Green and Savin (2008) investigated the no-show issue in further detail by conceptualizing an appointment system as a single-server queueing system in which customers who are about to enter service have a state-dependent probability of not being served and may rejoin the queue. They derived stationary distributions of the queue size for both deterministic and exponential service times, and compared their results with those obtained by simulation.

Muthuraman and Lawley (2008) developed a stochastic overbooking model along with an appointment scheduling policy for outpatient clinics. Their three guiding objectives were to minimize patient waiting times, maximize resource utilization, and minimize the number of patients waiting at the end of the day. Such patients must be treated during overtime. Similar work can be found in Cayirli et al. (2008) who investigated two approaches to patient classification when designing an appointment system. Through simulation they showed that by using interval adjustment for patient class, notable reductions in doctors' idle time and overtime, and patients' waiting times could be achieved without any trade-offs. Also see Patrick et al. (2008) who developed a Markov decision process for dynamically scheduling patients with different priorities at a public diagnostic facility. Rather than maximizing revenue, their objective was to adaptively allocate available capacity to incoming demand to achieve wait-time targets in a cost-effective manner. In the TRSP, no-shows are of limited concern because the majority of patients are homebound or in a chronic care facility.

3 Mathematical formulation

From a modeling point of view, there are at least two ways to approach the problem. The first, and the one that we pursue, exploits a routing analogy and considers each therapist as a time-constrained vehicle stationed at one of two depots—her home or her designated clinic (Kontoravdis and Bard 1995). The patients are demand points that must be visited over the course of a week [e.g., see Cheng and Rich (1998) for a home healthcare example]. The second views the therapists as entities flowing through a time-space network. Each node in the network represents either a patient–time period pair with a demand of 1, or an origin or destination for each therapist each day of the planning horizon [e.g., see Bard et al. (2001) for airline re-scheduling example].

A practical simplification is that patient demand is for a specific type of therapy (physical, occupational, speech), which can always be satisfied by a therapist of the appropriate certification. By implication then, the full problem decomposes by category so three separate models can be solved rather than one. We follow this approach and limit the remainder of the paper to scheduling physical therapists.

3.1 Vehicle routing formulation

In the context of vehicle routing, the problem can be modeled on a set of $|K|$ intersecting graphs $G^k = (V^k, A^k)$ consisting of the node set V^k and the arc set A^k. Here, $V^k = \cup_{d \in D} IK(k,d) \cup \{o(k),\ d(k)\}$ where $IK(k,d)$ is the set of all patients that can be seen by therapist k on day d. The set A^k contains all feasible arcs and is a subset of $V^k \times V^k$.

From a temporal point of view arc $(i, j) \in A^k$ can be eliminated for day d when $a_{id} + s_{id} + \tau_{ij} > b_{jd}$ for all $k \in K$ and $i, j \in IK(k,d)$. When the service time s_{id} is not a function of d, it is always possible to put $\tau_{ij} \leftarrow \tau_{ij} + s_{id}$ which is often done in the routing literature.

Several factors tie the days in a week together. The most critical is the requirement for multiple visits during the week, with the first preferably handled by a therapist. A second is the cumulative number of hours worked by each therapist during a week. When this number exceeds 40, overtime is paid at a rate of time and a half.

The notation used to define the model is given in "List of symbols" although many of the symbols have already been introduced. We initially present only a basic MIP omitting overtime, multiple daily visits, pattern selection, the requirement to assign either the first visit or at least one visit to a PT, and lunch breaks. These features are discussed in the next subsection.

Model for planning horizon (week)

$$\text{Minimize} \sum_{k\in K} \sum_{d\in DK(k)} \sum_{i\in IK(k,d)\cup\{o(k)\}} \sum_{j\in IC(i,d,k)\cup\{d(k)\}} \left(c^k_{ij} + c^k_{id}\right) x^k_{ijd} \tag{1a}$$

$$\text{subject to} \sum_{k\in K(i,d)} \sum_{j\in IC(i,d,k)\cup d(k)} x^k_{ijd} = 1,\ \forall\, i \in I,\ d \in DU(i) \tag{1b}$$

$$\sum_{j\in IK(k,d)} x^k_{o(k),j,d} \le 1, \forall\, k \in K,\ d \in DK(k) \tag{1c}$$

$$\sum_{i\in IC(j,d,k)\cup o(k)} x^k_{ijd} - \sum_{i\in IC(j,d,k)\cup d(k)} x^k_{jid} = 0,\quad \forall\, k \in K,\ d \in DK(k),\ j \in IK(k,d) \tag{1d}$$

$$t_{id} + \left(s_{id} + \tau_{ij}\right) - M_{ij}\left(1 - \sum_{k\in K(i,d)} x^k_{ijd}\right) \le t_{jd},\quad \forall\, i \in I,\ d \in DU(i),\ j \in IC(i,d,k) \tag{1e}$$

$$\sum_{k\in K(i,d)} \left(e_{kd} + \tau_{o(k),i}\right) x^k_{o(k),i,d} \le t_{id}, \quad \forall\, i \in I,\ d \in DU(i) \tag{1f}$$

$$\sum_{k\in K(i,d)} \left(f_{kd} - \tau_{i,d(k)} - s_{id} - b_{id}\right) x^k_{i,d(k),d} + b_{id} \ge t_{id},\quad \forall\, i \in I, d \in DU(i) \tag{1g}$$

$$x^k_{ijd} \in \{0,1\}, t_{id} \in [a_{id}, b_{id}],\quad \forall\, i \in I, d \in DU(i),\ j \in IC(i,d,k),\ k \in K(i,d) \tag{1h}$$

The objective in (1a) is to minimize the cost of traveling to and from the home base (residence) of each therapist and between all pairs of patients plus the cost of providing service. The first constraints (1b) ensure that each patient has exactly one successor on a route on day d. The first summation is over the set of therapists who are qualified to treat patient i and the second is over all patients who are permitted to be on the same route as i. If $j \notin IC(i, d, k)$, then either i and j requires different types of treatment from different therapists or the pair is incompatible due to their respective time windows or physical locations. These situations are sorted out in a preprocessing step where all the sets are defined. The requirement that patient i be seen by a particular therapist k on day d can be accommodated by appropriately specifying the set $K(i,d)$.

Constraints (1c) limit each therapist $k \in K$ to at most one route per day. By implication, when $x^k_{o(k),j,d} = 0$ for all $j \in IK(k, d)$, therapist k is not given a schedule on day d. If a subset of therapists, say, n_k of them, have the same profile and can be viewed as interchangeable (that is, homogeneous), then the right-hand side of (1c) can be replaced with n_k and K redefined to be the set of therapist types rather than individuals.

Constraints (1d) impose route continuity for each therapist k and each compatible patient $j \in IK(k, d)$ by requiring that tours (loops) are constructed rather than open paths. The start $o(k)$ and end $d(k)$ of a tour for therapist k is either his home base or residence, depending on the contract terms. Assuming positive flow on the network G^k, when (1d) is combined with (1b), we see that each patient j who is visited by k has a unique predecessor and a unique successor.

Constraints (1e)–(1g) ensure time feasibility of the schedule. The binary flow variables x^k_{ijd} are related to the time variables t_{id} in (1e), which indicates that when patient i is the immediate predecessor of patient j on a tour, the (unspecified) therapist cannot arrive at the location of j before providing service at i and then traveling to j. Here, $M_{ij} = \max\{b_{id} + s_{id} + \tau_{ij} - a_{jd}, 0\}$, and if the time windows of i and j are such that $a_{jd} - b_{id} - s_{id} > \tau_{ij}$, then τ_{ij} can be replaced by $a_{jd} - b_{id} - s_{id}$ to tighten the formulation. When i is not the immediate predecessor of j, (1e) is redundant as the definition of M_{ij} guarantees. An added benefit of this constraint is that it eliminates subtours by requiring that arrival times on a route be increasing.

Constraints (1f) and (1g) respectively place lower and upper bounds on the time t_{id} that a therapist can begin treatment on patient i on day d based on the therapists' contract hours. To ensure feasibility when patient i is not scheduled to be seen on day d, (1g) restricts the service initiation variable, t_{id}, to be less than or equal to b_{id}, the upper end of the time window (in practice, this constraint would not be generated for this case). Bounds are placed on the variables in (1h).

3.2 Additional requirements

Model (1a)–(1h) captures the primary features of the TRSP but omits five sets of constraints. Each of these is outlined below but the specific equations are omitted to avoid introducing additional notation that is not used in the solution methodology. The details are provided by Shao (2011).

3.2.1 Patient patterns

Constraints (1b) assume that the treatment days for each patient are known a*priori*. This is true for the fixed patients but not for the flexible ones. For the latter, we have the option of selecting from a set of visit patterns defined over the week. For example, if a patient is to be seen on two nonconsecutive days, then there are six feasible patterns from which to choice: {(M,W), (M,Th), (M,F), (T,Th), (T,F), (W,F)}. For each patient, we define a binary variable for each such pattern and write constraints which ensure that exactly one of those variables assumes a value of 1 in a solution.

3.2.2 First visit

For some patients, it may be necessary for a PT rather than a PTA to handle the treatment for at least one visit during the planning horizon. There are two cases to consider. The first is more complex and requires that the first visit be by a PT; the second simply requires that at least one visit is by a PT. Both cases are included in the full model.

3.2.3 Multiple sessions per day

It is not uncommon for some patients to require two or more sessions in a single day. The most likely scenario is more than five visits during the week, but a patient might only be scheduled for a single day with one treatment in the morning and another in the afternoon. Rather than defining additional patterns to model such situations, we represent the patient with multiple records, each of which requires no more than one visit per day. The corresponding sessions must be separated by a minimum number of hours.

3.2.4 Lunch breaks

The simplest way to build the lunch break into the schedule for therapist k on day d is by identifying all patient pairs (i,j) that can accommodate an extra 1/2 between them and inserting the break between one of the pairs. In this approach, it is assumed that lunch is taken at either locations i or j, or on the road between them but always within the interval [11:00 a.m., 1:00 p.m.]. To implement this feature, a set of binary variables indexed by i, j, d and k are needed to represent the assignment of the break, and a set of continuous variables indexed by d and k are need to indicate the starting time of the break. In addition, a corresponding set of constraints similar to (1b) and (1e) must be included in the MIP.

3.2.5 Tracking paid time and overtime

To compute the weekly earnings of a therapist, it is necessary to keep track of his paid hours, which includes travel time, administrative time, overtime, as well as treatment time. Therapists may work for more than 8 hours in a day but overtime doesn't start until their paid time exceeds 40 hours in a week. Several additional constraints are needed to keep track of the amount of paid time accrued in a day for each therapist k. Given these daily values, the number of overtime hours worked, T_k^{over}, can be computed. The final step is to add an overtime term to the objective function (1a) of the form $\sum_{k \in K} 0.5 c_k^{\text{reg}} T_k^{\text{over}}$, where c_k^{reg} is the hourly wage rate of therapist k.

As mentioned, the primary factors that tie the days of the week together and prevent the problem from being decomposed by day is the need to (i) account for overtime, (ii) fix patterns for the flexible patients, and (iii) provide at least one PT visit for certain flexible patients. A secondary factor that prevents decomposition is that some patients require two treatments on one or more days.

4 Solution methodology

Letting $|D|$ be the number of days in the planning horizon, $|I|$ the number of patients, $|P|$ the number of feasible patterns, and $|K|$ the number of therapists, model (1a)–(1h) requires approximately $|D| \cdot |I|^2 \cdot |K| + |I| \cdot |P|$ binary variables and $|D| \cdot |I|^2 + |D| \cdot |I| \cdot |K|$ constraints. For a small instance with $|D| = 5$, $|I| = 50$, $|P| = 20$ and $|K| = 4$, it might have on the order of $5 \times 50^2 \times 4 + 50 \times 20 = 51,000$ binary variables and $(5 \times 50^2 + 5 \times 50 \times 4) = 13,500$ constraints, which proved to be far beyond the capability of CPLEX. Real-world instances may be considerably larger.

To ensure reasonable runtimes, we developed an adaptive sequential GRASP. Phase I is designed to uncover a diversity of high-quality solutions by randomly selecting therapist-day pairs in sequence and finding the "optimal" route for each in turn. This process is repeated many times (250 times in the experimental design) to allow for a wide exploration of the feasible region. Phase II attempts to improve a subset of the candidates uncovered in Phase I using a high-level neighborhood search defined by insertions and swaps.

A feasible solution to model (1) is a set of *routes* $\{(k, d), \text{ for all } k \in K, d \in D(k)\}$, where each consists of a sequence of patients who are visited by therapist k on day d. In general, heuristics for routing and scheduling problems can be divided into two categories: sequential and parallel. The former creates one route at a time until all resources are exhausted or all patients are routed; the latter builds multiple routes simultaneously. Depending on the properties of a problem instance, we have found that when feasibility is not an issue and optimality is the focus, parallel heuristics may be superior because they are less myopic with respect to total cost. When feasibility is critical as is often the case here, sequential heuristics have proven to be superior since they are less myopic with respect to idle time within a route.

A flow diagram for our sequential GRASP is given in Fig. 2. The first step in Phase I is to fix the PT visits for the new patients. We then start building routes one at a time until all therapists have schedules on their available days (these could be void). In Phase II an attempt is made to find a local optimum by exploring the neighborhood of a subset of feasible solutions. Infeasible moves are not accepted due to the complexity of the constraint region. The procedure terminates when a predefined number of iterations is reached. In Appendix 1, the complexity of both phases is examined.

4.1 Phase I

In this phase, feasible solutions are constructed by stringing individual routes together over the week. In Step 1, the day of the week is fixed for patients who must be treated by a PT at least once over the planning horizon. In Step 2, all compatible therapist-day pairs or routes (k,d) are identified, and one, say, (k^*, d^*) is selected at random to build. In Step 3, we identify and sort a subset of patients who are eligible for route (k^*, d^*), and randomly select one at a time from the top of the list. When the route can no longer be extended, it is closed and a new one is started. The final stop of the day is the therapist's home base. Construction terminates when all patients have been routed or when there are no more feasible patient-therapist assignments. In the latter case, which implies infeasibility, the remaining patients are assigned to a "supertherapist" who has an extremely high wage rate ($1000/h).

4.1.1 Fixing (first) visit day when PT is required

The requirement that some patients must be seen by a PT introduces some ambiguity as to whether, say, therapist k is compatible with patient i on day d. For example, if patient i requires several treatments over the week and at least one of them must be provided by a PT, then it may not be clear whether she should be seen by a PT or PTA on day d. Fixing the PT treatment day to, say, Tuesday, however, removes any ambiguity and ensures PTA-patient compatibility on the remaining days.

In the more complex case that requires the first treatment of the week to be provided by a PT, we fix the day by considering all feasible patterns and then randomly selecting one.

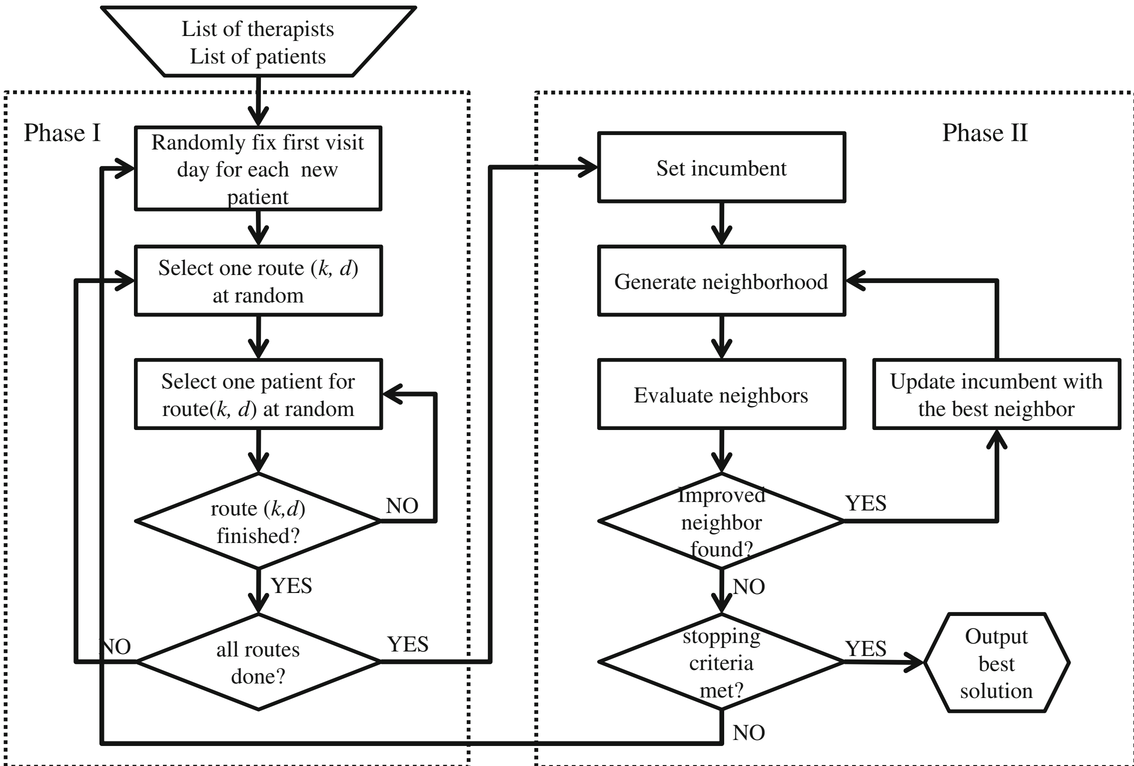


Fig. 2 Flow diagram for sequential GRASP

Let D_1 be the set of first day candidates associated with all feasible patterns. A simple rule is to select $d_1 \in D_1 \subset D$ with probability $1/|D_1|$.

Example 1 Assume patient i has feasible patterns (1, 2, 3), (1, 2, 4), (2, 3, 4) and (3, 4, 5). Thus, the candidate set $D_1 = \{1, 2, 3\}$, so the first visit would be randomly fixed at day 1, 2 or 3 with probability 1/3. □

Another possible rule is to define the probabilities by taking into account the number of times a day is first in a pattern. Letting $a(d_1)$ be the frequency of day $d_1 \in D_1$, in this case we set its probability of being selected at $a(d_1) \Big/ \sum_{d \in D_1} a(d)$.

Example 2 Continuing with Example 1, the frequency set for D_1 is {2, 1, 1}, so the probabilities of selecting days 1, 2 or 3 for the first visit are 2/4, 1/4, and 1/4, respectively. □

Of course, more sophisticated rules that, say, take into account the remaining time of all PTs on a specific day after considering the fixed and flexible patients that must be seen on that day, can be developed. However, the two proposed rules proved sufficient for our purposes.

In the less restrictive case where only one treatment must be provided by a PT during the week, we do not fix the specific day in advance. Instead, we let the algorithm do it. If by the last visit, a patient that must be seen a PT has not been so scheduled, we force that visit to be by a PT. Improvements are left to phase II. In the rare case when no compatible therapists are available, the iteration is declared infeasible and is discarded.

4.1.2 Route selection

A two-step process is used to select the next route (k,d) to build. In step 1, we identify a set of candidate days $\bar{D} \subseteq D$ without regard to therapist. Due to the unbalanced distribution of therapist time and patient demand over the week, the remaining available therapist time might differ greatly from one day to the next. Because it is more difficult to achieve feasibility when patients are first assigned to days with few therapists, such days are given low priority. In step 2, we sort the compatible routes (k, d) for $\bar{D}$ in increasing order of the therapists' wage rates, and then select one randomly from the top of the list.

Step 1: Determining the candidate list of days, $\bar{D}$. Given that the remaining available time for all therapists on each day might be quite different, before we assign a flexible patient to a particular day, it is necessary to know whether there is at least one therapist with enough time remaining on that day to treat the patient. As such, we need to estimate the remaining available therapist time on each day.

On day d, the total amount of available therapist time T_d is fixed; it is the summation of the working time of all available therapists on day d. By examining patients' treatment requirements on this day, we can estimate how much therapist time will be consumed and then calculate the remaining available time. Three types of patients are included in the calculations: (i) fixed patients with scheduled appointments on day d; (ii) flexible patients who must be seen on day d as indicated by their feasible pattern sets; and (iii) flexible patients who have already been assigned to day d.

Letting $I(d)$ be the set of the patients that must be visited on day d, we estimate the amount of therapist time associated with each patient $i \in I(d)$. This includes treatment time, travel time and administrative time. Let τ_{id} be the estimated traveling time imposed by patient i on day d, and prod_d be the estimated mean productivity rate of all therapists who are available on day d. From the latter parameter, the estimated administrative time for patient i is $(1-\text{prod}_d)s_{id}$. Letting $\bar{T}_d$ be the estimated remaining available therapist time, we have

$$\bar{T}_d = T_d - \sum_{i \in I(d)} \left(s_{id} + \tau_{id} + \left(1 - \text{prod}_d\right) s_{id}\right) \tag{2}$$

For arbitrary days d_1 and d_2, if the difference $\left|\bar{T}_{d_1} - \bar{T}_{d_2}\right|$ is significant, then feasibility is more likely to be achieved when we consider the day with more time remaining first. Otherwise, we can say that these two days have the same priority level. To operationalize this idea, we propose a three-level function to classify the priority of each day. Letting s be the estimated average time a therapist needs for a visit, including treatment time, administrative time and traveling time, and level_d be priority level of day d, we define

$$level_d = \begin{cases} 1, & \text{if } \bar{T}_d \in [s \cdot n_1, \infty) \\ 2, & \text{if } \bar{T}_d \in [s \cdot n_2, s \cdot n_1) \\ 3, & \text{if } \bar{T}_d \in [0, s \cdot n_2) \end{cases} \tag{3}$$

where $n_1 > n_2$ are parameters corresponding to the number of additional patients that can be visited on day d. Thus, level 1 indicates that approximately n_1 or more additional patients can be scheduled on day d, and hence offers the most flexibility. At the other extreme we have level 3, which indicates that less than n_2 additional patients can be treated on day d. The intermediate case is represented by level 2.

Letting D^1 be the set of days at level 1, D^2 the set of days at level 2, and D^3 the set of days at level 3, when selecting a day, we always consider D^1 first, then D^2 if $D^1 = \varnothing$, and then D^3 if $D^1 = D^2 = \varnothing$. That is, we set $\bar{D} = D^l$ such that $l = \min\{l : D^l \neq \varnothing, l = 1, 2, 3\}$. If we view available therapist time as a resource, by using this construct we are assigning flexible patients to days with ample resources first. When the algorithm starts, most days will be at level 1 and hence will be treated identically; as the algorithm progresses, more and more patients are routed, which produces a more uniform distribution of days among the levels.

In the implementation, we take the average traveling time over all adjacent arcs as the estimated traveling time τ_{id}, i.e., $\tau_{id} = \sum_{j \in IC(i,d)} \tau_{ij} \Big/ |IC(i,d)|$. Also, we take the average productivity rate over all available therapists on day d as the estimated productivity rate prod_d, and set the estimated average therapist time s to 1 h for all i and d. For the remaining parameters, we set $n_1 = 4$ and $n_2 = 1$ recognizing that one therapist can generally treat no more than 10 patients per day.

Example 3 For a 1-week planning horizon we have $D = \{1, 2, 3, 4, 5\}$. Assume that the corresponding set of daily available therapist time T_d is $\{12, 8, 4, 16, 8\}$, the set of treatment times for patients in $I(d)$ is $\{4, 4, 0, 10, 4\}$, the set of estimated travel time imposed by patients in $I(d)$ is $\{2, 2, 0, 3, 1\}$, and the set of estimated administrative time for patients in $I(d)$ is $\{1, 2, 0, 3, 1\}$. Subtracting the treatment, travel and administrative time from the available time gives the set of estimated remaining available therapist time $\bar{T}_d$ as $\{5, 0, 4, 0, 2\}$. Accordingly, the level definitions in (3) imply that $D^1 = \{1, 3\}$, $D^2 = \{5\}$ and $D^3 = \{2, 4\}$, and $\bar{D} = D^1 = \{1, 3\}$. □

Step 2: Selecting the next route (k,d) Based on the set $\bar{D}$, we determine the set of candidate routes $\bar{R} = \{(k, d) : d \in \bar{D},\ k \in K(d)\}$. The elements in $\bar{R}$ are subsequently sorted in increasing order of the therapists' wage rates; the lower the rate, the higher the priority. From the ordered list we consider only the first m_r entries to get the restricted candidate list (RCL), a basic GRASP parameter, and then select one at random with probability $1/m_r$.

Example 4 Using the data in Example 3, we have $\bar{D} = \{1, 3\}$. Assume that we have three therapists k_1, k_2, k_3 with wage rates 40,50,35, respectively. Among them, k_1 and k_2 work on day 1 and k_s works on day 3, so the set of candidate routes $\bar{R} = \{(k_1, 1), (k_2, 1), (k_3, 3)\}$. After sorting them according to the wage rates, we have the candidate list $CL = [(k_3, 3), (k_1, 1), (k_2, 1)]$, and if the cutoff number $m_r = 2$, then RCL= $[(k_3, 3), (k_1, 1)]$ with the probability of selecting either entry being 0.5. □

The pseudocode for selecting a candidate route is given in Fig. 3.

4.1.3 Route construction

After the therapist-day pair (k, d) is selected, we first determine the set of candidate patients $I(k, d)$ that can be visited by therapist k on day d. Patient i is said to be compatible with route (k, d) if the following two conditions are satisfied: (i) therapist k is certified to treat patient i on day d, i.e., $i \in IK(k, d)$ and (ii) there exists a feasible pattern for patient i that includes day d. The second condition varies iteration by

Procedure	`Candidate_Route_Selection`
Input	Set of therapists K, set of patients I, and set of days D
Output	Route (k, d)

Step 0: Initialization

    For ($d \in D$) {
        Initialize the set of available therapist $K(d)$ on day d;
        Initialize the set of patients $I(d)$ that must be visited on day d;
    }

Step 1: Construct the set of candidate days $\overline{D}$

    For ($d \in D$) {
        Calculate the available remaining time T_d;
        Calculate the estimated required therapist time for patient $i \in I(d)$;
        Calculate the estimated remaining available time $\overline{T}_d$ according to Eq. (2);
        Calculate the level of day d according to Eq. (3) and construct D^l, $l = 1,2,3$
    }
    Determine the set of candidate route days $\overline{D} = D^{l^*} \ni l^* = \min_{l=1,2,3}\{l : D^l \neq \varnothing\}$;

Step 2: Select a route $(k, d) \in \overline{R}$

    Construct $\overline{R} = \{(k, d) : d \in \overline{D}, k \in K(d)\}$
    Sort entries in $\overline{R}$ according to wage rates in increasing order;
    Form the *RCL* and select one randomly with probability $1/m_r$;
    Return (k, d);

Update: Updating $K(d)$ and $I(d)$ after the route (k, d) is constructed

    Put $K(d) \leftarrow K(d) \setminus \{k\}$;
    Put $I(d) \leftarrow I(d) \setminus \{i$: patient i is assigned to route $(k, d)\}$;
    Go to Step 1;

Fig. 3 Pseudocode for selecting a candidate route

iteration depending on the days on which patient i has already been scheduled. For example, if patient i requires three consecutive visits, then only patterns {M,T,W}, {T,W,Th}, and {W,Th,F} are feasible. Initially then, she is available each day of the week. If after several iterations, she is assigned a visit on Tuesday, then the third pattern is no longer feasible because it includes Friday. Now assume that after several more iterations she is scheduled for a visit on Thursday. This leaves only pattern {T,W,Th} so her third visit must be on Wednesday.

After the candidate set $I(k, d)$ is created, we build route (k, d) by assigning one patient $j \in I(k, d)$ at a time starting with therapist k's point of origin $o(k)$ and ending at his destination $d(k)$. Given the current patient i, we select the next patient j using a four-step process. In step 1, we decide the length of the RCL for selecting the next patient . In step 2, we construct the RCL by calculating the earliest time that the next treatment can start for a certain number of feasible candidates j (see below). In step 3, a benefit measure is computed for each $j \in \text{RCL} \subseteq I(k, d)$, and in step 4, one of the top candidates, as ranked by their benefit measure, is randomly selected.

Step 1: Length of RCL Depending on the geographic distribution of therapists and patients, it might not be efficient to assign a patient to a therapist if their respective locations are too far apart. For such pairs, it is not necessary to check their compatibility, thus reducing the computational effort. In our problem, we build the RCL by including only those patients that are within a certain distance from the therapist's home base. Since RCL guides the construction process in Phase I, its length, l_{RCL}, must strike a balance between solution quality and diversity, and CPU time. If l_{RCL} is large, it is likely to produce inferior initial solutions and require more computations; if l_{RCL} is small, many good solutions may be missed. Therefore, instead of setting l_{RCL} to a fixed value, we restrict it to within the following set: $l_{\text{RCL}} \in \{l_{\min}, \ldots, l_{\text{std}}, \ldots, l_{\max}\}$, where $l_{\min}$ and $l_{\max}$ are predetermined minimum and maximum lengths, and $l_{\min} < l_{\text{std}} < l_{\max}$.

At the first iteration, we set $l_{\text{RCL}} = l_{\text{std}}$. If Phase I returns a solution in which all demand has been satisfied and some therapists have no patient assignments on a subset of their working days, then we select l_{RCL} from the set $A^1 = \{l_{\min}, \ldots, l_{\mathrm{std}}\}$ in the remaining iterations; otherwise, l_{RCL} is selected from $A^2 = \{l_{\mathrm{std}}+1, \ldots, l_{\max}\}$. For either set, the value of l_{RCL} is adaptively adjusted using a variation of the method proposed by Prais and Ribeiro (2000). In particular, let $A = A^1 \cup A^2 = \{\alpha_1, \alpha_{2,\ldots,}\alpha_m\}$ be the set of considered values for l_{RCL} and let p_i be the corresponding probability of selecting $\alpha_i, i = 1, \ldots, m$. Initially, one Phase I iteration is performed for each $\alpha_i, i = 1, \ldots, m$. Let ϕ^* be the best solution found in all previous GRASP iterations and let A_i be the average value of solutions obtained for $l_{\mathrm{RCL}} = \alpha_i$. Now, define $q_i = (\phi^*/A_i)^\delta, i = 1, \ldots, m$ to be the relative performance of the algorithm under α_i, where δ is a shape parameter. For higher values of δ, q_i will be lower since $\phi^* \le A_i$. Normalizing gives $p_i = q_i / \sum_{\gamma=1}^{m} q_\gamma, i = 1, \ldots, m$.

When α_i yields relatively small average solutions A_i, it will have a high probably p_i of being selected as the iterations progress. In the implementation, $\delta = 50$ and $A = \{l_{\min} = 1, 2, \ldots, 9, l_{\mathrm{std}} = 10, 11, \ldots, 19, l_{\max} = 20\}$.

*Step 2: Starting time of the next patient **j**.* Assume patient i is the current patient in route (k, d). Let t_i be the time that therapist k starts treatment for patient i on day d and let $\hat{t}_j$ be the earliest time that treatment can begin for patient j, where $\hat{t}_j = t_i + s_{id} + \tau_{ij}$. Using $\hat{t}_j$, we can set the starting time t_j for patient j. If j is a fixed patient, then $t_j = a_{id}$; if j is a flexible patient, then t_j is set as early as possible. That is,

$$t_j = \begin{cases} a_{id}, & \text{if } i \in I_{\mathrm{FIX}} \\ \max\left\{a_{id}, \hat{t}_j\right\}, & \text{if } i \in I_{\mathrm{FLEX}} \end{cases} \tag{4}$$

To guarantee the feasibility of patient j, we need to check two conditions. One is that t_j is early enough so that the treatment can be finished within the respective time windows of therapist k (so he can return home no later than f_{kd}) and patient j, i.e., $t_j \le \min\{b_{jd}, f_{kd} - s_{jd}\}$. The other is that t_j should be late enough so that therapist k has ample time to travel from i to j, i.e., $t_j - \hat{t}_j \ge 0$. We denote $t_{ij}^{\mathrm{idle}} = t_j - \hat{t}_j$ as the idle time between i and j. Only patients satisfying these two conditions are considered as *feasible candidates*.

Example 5 Using a 24-h clock, for route (k, d), assume that therapist k is a PT with time window [8.00, 17.00]. Also assume that the current patient i starts treatment at 13.50 and has a 0.5-h treatment time. There are five candidates for the next patient j. The type of each, their time windows and treatment times are listed in Table 1.

Table 2 enumerates the feasibility data for the candidates. Column 2 gives the travel time between patients i and j, and columns 3–5 report the earliest time, start time and idle time for patient j, respectively. The last column indicates whether or not patient j is a feasible successor of patient i. As we can see, fixed patient 3 is infeasible because therapist k does not have sufficient time to arrive at the patient's location before the required starting time, $t_3 = 14.25 < 14.30 = \hat{t}_3$, i.e., $t_{ij}^{\mathrm{idle}} = -0.05 < 0$. Flexible patient 4 is infeasible because therapist k cannot start the treatment within the required time window, i.e., $t_4 = 14.75 > 14.50 = b_{4d}$. □

Table 1 List of candidate patients

Patient j	Patient type	Time windows $[a_{jd}, b_{jd}]$	Treatment time s_{jd}
1	Fixed	[15.00, 15.00]	0.50
2	Flexible	[9.00, 16.75]	0.25
3	Fixed	[14.25, 14.25]	0.50
4	Flexible	[7.00, 14.50]	0.50
5	Flexible	[9.00, 16.50]	0.50

Table 2 Feasibility data for candidate patients

Patient j	Traveling time τ_{ij}	Earliest time $\hat{t}_j$	Start time t_j	Idle time t_{ij}^{idle}	Feasible?
1	0.50	14.50	15.00	0.50	Yes
2	0.75	14.75	14.75	0.00	Yes
3	0.30	14.30	14.25	−0.05	No
4	0.75	14.75	14.75	0.00	No
5	1.25	15.25	15.25	0.00	Yes

The additional features described in Sect. 3.2 can be handled by slightly modifying the definition of t_j. If multiple sessions are required for patient j on day d, let $S_j = \{j_1, j_2, \ldots, j_N\}$ be the set of N sessions that have already been scheduled and let $\{t_{j1}, t_{j2}, \ldots, t_{jN}\}$ be the set of the corresponding starting times. Without loss of generality, we assume that $t_{j1} < t_{j2} < \ldots < t_{jN}$. Now, recalling that each session must be separated by a certain number of hours, say $\tau^{\mathrm{min\_sep}}$ hours, we let $t_{j0} \equiv 0$ and $t_{j,N+1} \equiv b_{jd} + \tau^{\mathrm{min\_sep}}$ be two dummy time points, and define t_{new} to be the starting time of the new session. Using Eq. (4), we set $t_{\mathrm{new}} = t_j$ and try to find the first session γ that satisfies the following two conditions: $t_{j,\gamma-1} \le t_{\mathrm{new}}$ and $t_{j\gamma} - \max\{t_{\mathrm{new}}, t_{j,\gamma-1} + s_{jd} + \tau^{\mathrm{min\_sep}}\} \ge s_{jd} + \tau^{\mathrm{min\_sep}}$. These conditions enable us to identify the earliest start time for the next session while satisfying the $\tau^{\mathrm{min\_sep}}$ -hour separation requirement. If such a γ does not exist, then patient j is not compatible with therapist k on day d so j must be treated by a different therapist or by the same therapist on a different day; otherwise, we put $t_{\mathrm{new}} \leftarrow \max\{t_j, t_{j,\gamma-1} + s_{jd} + \tau^{\mathrm{min\_sep}}\}$ and if necessary, update $t_j = t_{\mathrm{new}}$.

If therapist k qualifies for a lunch break and it has not yet been scheduled, we consider placing it between patients i and j if $t_j + s_{jd} > 13 - \tau^{\mathrm{lun}}$; here, τ^{lun} is the duration of the break. When this condition is met, we add τ^{lun} to the earliest start time $\hat{t}_j$ by putting $\hat{t}_j \leftarrow \hat{t}_j + \tau^{\mathrm{lun}}$, and then recalculate the starting time t_j. If $t_j + s_{jd} \le 13 - \tau^{\mathrm{lun}}, t_j \ge 11 + \tau^{\mathrm{lun}}$

and $t_{ij}^{\text{idle}} \geq \tau^{\text{lun}}$, we can also put the break between patient i and patient j without modifying t_j. This strategy can be shown to lead to a feasible break assignment (see Shao et al. 2012).

In implementation, all unassigned patients are sorted according to their distance to the therapist's home base in nondecreasing order. The feasibility of each patient is checked until l_{RCL} compatible patients have been identified or until the list is exhausted. The procedure terminates with the RCL.

Step 3: Benefit function for a feasible candidate ***j***. When patient j is the immediate successor of patient i on a route, it is likely that some idle time between them will exist when the former is a fixed patient. To obtain a feasible solution, it may be necessary to select a $j \in I_{\text{FIX}}$ such that $t_{ij}^{\text{idle}} > 0$. For $j \in I_{\text{FLEX}}$, we expect $t_{ij}^{\text{idle}} = 0$, but this may not always be the case, particularly when j has multiple sessions or a stipulated time window for receiving treatment. In such situations, it is probably best to schedule the patient later in the sequence, which can be handled by assigning her a low benefit value.

To address these issues, we define σ_j^{idle} as the idle time tolerance for patient j and introduce the factor ω_j that gauges the effect of idle time on the priority of j. For given i, we let

$$\omega_j = \begin{cases} 1 - \log\left(t_{ij}^{\text{idle}} \Big/ \sigma_j^{\text{idle}} + \varepsilon^{\text{idle}}\right), & \text{for } j \in I_{\text{FIX}} \\ \exp\left(-t_{ij}^{\text{idle}} / \sigma_j^{\text{idle}}\right), & \text{for } j \in I_{\text{FLEX}} \end{cases} \tag{5}$$

where $\varepsilon^{\text{idle}}$is an arbitrarily small positive number to ensure the feasibility of the logarithm.

The top component in Eq. (5) is designed for fixed patients. Here, $\omega_j \in (-\infty, \infty)$ is a decreasing function of idle time and is (approximately) equal to 1 when $t_{ij}^{\text{idle}} = \sigma_j^{\text{idle}}$. When $t_{ij}^{\text{idle}} < \sigma_j^{\text{idle}}$ higher priority is given to fixed patient j; when $t_{ij}^{\text{idle}} > \sigma_j^{\text{idle}}$ lower priority is given to fixed patient j.

The bottom component in Eq. (5) is designed for flexible patients. Here, $\omega_j \in (0, 1]$ and again is a decreasing function of idle time with maximum value 1 when $t_{ij}^{\text{idle}} = 0$. Thus, the more idle time the less desirable it is to schedule patient $j \in I_{\text{FLEX}}$.

We now turn to the design of the benefit function, which should tell us how cost-efficient it would be to select patient j as the immediate successor to i. We consider two factors: travel time and treatment time. Smaller travel times are always preferred since the corresponding costs are proportional. However, it can be argued that patients with longer treatment times should be considered first since the therapist associated with the route (k, d) is likely to have a lower wage rate than the therapists assigned to the remaining routes. To capture these factors, we have tentatively designed the benefit function, denoted by $\hat{v}_j$ for patient j, to be a decreasing function of traveling time and an increasing function of treatment time. Specifically, given i we have

Table 3 List of benefit value for feasible patients

Patient j	Idle time factor ω_j	Benefit $\hat{v}_j$	Travel time τ_{j1}	Adjusted benefit v_j
1*	1.3	2.00	–	2.00
2	1.0	0.44	0.00	2.44
5	1.0	0.32	0.25	0.32

* Notation j^* means candidate j has the highest benefit value

$$\hat{v}_j = \omega_j \left(s_{jd}\right)^{\alpha} \Big/ \left(\varepsilon + \left(\tau_{ij}\right)^{\beta}\right) \tag{6}$$

where $\alpha \geq 0$ and $\beta \geq 0$ are two parameters used to trade off the treatment time s_{jd} with the travel time τ_{ij}, and ε is a small positive constant used to guarantee feasibility.

Based on the structure of $\hat{v}_j$, it is possible that there might exist some candidate j^* with a high benefit value but with significant idle time preceding the start of her treatment. As such, it may be desirable to try to insert another patient j between i and j^*. To address this situation, we increase the benefit of patient j by adding the value of patient j^* to it. Let j^* correspond to the patient with largest benefit value, and let v_j denote the modified benefit function for patient j that is,

$$v_j = \begin{cases} \hat{v}_j + v_{j^*}, & \text{if } t_j + s_{jd} + \tau_{jj^*} \leq t_{j^*} \\ \hat{v}_j, & \text{otherwise} \end{cases} \tag{7}$$

Of course, other adjustment schemes could be explored that consider, for example, the treatment time as well as the benefit associated with each patient that can feasibly be scheduled between i and j^*.

Example 6 Continuing with Example 5, we have three feasible candidates, patients 1, 2 and 5. For $\alpha = 1, \beta = 2, \varepsilon = 0.0001$ and $\sigma_1^{\text{idle}} = \sigma_2^{\text{idle}} = \sigma_5^{\text{idle}} = 1$, Table 3 enumerates the calculated values obtained from Eqs. (5)–(7). Taking patient 2 as the example, we have idle time factor $\omega_2 = \exp(0/1) = 1$, and benefit value $\hat{v}_2 = 1(0.25)^1/(0.0001 + (0.75)^2) = 0.44$.

From the third column, we see that patient 1 has the highest original benefit $\hat{v}_j = 2.00$ so $j^* = 1$. In the last column, the adjusted benefit value of patient 2 is increased from $\hat{v}_1 = 0.44$ to $v_j = 2.44$ in light of Eq. (7) since $t_2 + s_{2d} + \tau_{21} = 14.75 + 0.25 + 0.00 = 15 \leq t_1 = 15$. □

Step 4: Selecting next patient ***j***. By using Eqs. (5)–(7), we can calculate the benefit value for each feasible candidate j in RCL. These patients are now sorted in decreasing order of v_j. For a given m_p, we remove all but the top m_p candidates and select one at random with probability $v_j / \sum_{j=1,\ldots,m_p} v_j$, $j = 1, \ldots, m_p$.

Example 7 Using the same data in Example 6, we have RCL= $\{2, 1, 5\}$ after sorting them according to benefit values v_j in decreasing order. If $m_p = 2$, then patients 1 and 2 will be selected with probability 0.45 and 0.55, respectively. □

```
Procedure   Route_Construction
Input       Therapist k, day d, and set of sorted compatible patients IK(k, d), and l_RCL
Output      Sequence of patients in route (k, d)
Step 0: (Initialization)
          Build the set of candidate patients I(k, d);
          Set the first patient i to the origin of therapist k, i.e., i ← o(k); set t_i = e_kd;
Step 1: Calculate starting time for patient j ∈ I(k, d) according to Eq. (4); form RCL.
Step 2: Calculate benefit value for feasible patient j ∈ RCL according to Eq. (7);
Step 3: Select patient j randomly from the top m_p;
Step 4: Updating
          Insert patient j in sequence;
          if (patient j = destination d(k)) {
                Return sequence of patients;
                Stop;
          }else{
                Put I(k, d) ← I(k, d) \ {j}  \\ Remove patient j from I(k, d);
                Put i ← j;
                Goto Step 1;
          }
```

Fig. 4 Pseudocode for route construction

The pseudocode for constructing a route is given in Fig. 4.

4.2 Phase II

In this phase, we seek a local optimum by exploring several neighborhoods around the Phase I solutions. In general, a neighborhood is the set of points that can be reached from the current point with minor adjustments. Over the last 20 years, there has been expanding research on advanced neighborhood structures such as edge exchanges (Savelsbergh 1992), 2-OPT* (Potvin and Rousseau 1995), ejection chains, and CROSS exchanges (Taillard et al. 1997), primarily in the context of vehicle routing. Each of these typically requires the simultaneous modification of multiple customers, which might include revising their visit orders or moving them to different routes. Because of the complexity of a neighborhood, Phase II is not necessarily called after every Phase I iteration but at predetermined frequency.

In our problem, recall that a route (k, d) is a sequence of patients who are visited by therapist k on day d. A route might include fixed patients, flexible patients, new patients, patients requiring treatments by a specific therapist type, and patients requiring multiple visits on the current day. The individual restrictions associated with each patient type greatly increase the likelihood of infeasibility when the appointments of more than one patient are modified simultaneously. After testing several elaborate moves like chain swaps and three-way swaps, we settled on two basic procedures: (i) insertions—moving one patient from her current position to another position either in the same route or in a different route, and (ii) swaps—exchanging two patients who are either in the same route or in different routes. We also decided to forego the use of path relinking (e.g., see Boudia et al. 2006) because we have not found it cost-effective in previous work. In the remainder of this section, we introduce the rules used to guarantee feasibility and to evaluate the value of a move.

4.2.1 Slack block

To facilitate the examination of a neighborhood, we introduce the concept of a *block*, which is a subsequence of patients in a route that begins and ends with either a fixed patient, or the origin or destination of the therapist, whichever is relevant. In our notation, patients are uniquely identified by 4-digit numbers so a typical route (k, d) is represented as follows.

$$o(k) \to 2001 \to 1001 \to 2002 \to 2003 \to 2004 \to 2005 \to 1002 \to 2006 \to d(k) \quad (8)$$

In the sequence, flexible patients start with a "2" and fixed patients start with a "1." This route can be decomposed into three blocks: $o(k) \to 2001 \to 1001$, $1001 \to 2002 \to 2003 \to 2004 \to 2005 \to 1002$ and $1002 \to 2006 \to d(k)$. Because the visit times for fixed patients are firm, any schedule modifications for the flexible patients in one block do not affect the schedules of patients in other blocks. This gives rise to what we refer to as a *slack block*.

Given a route (k, d) with $N + 1$ patients, let $IR = \{r_0, r_1, \ldots, r_N\}$ be the corresponding patient sequence. We define a pair of integers $[a, b]$ such that $0 \leq a \leq b \leq N$ as a *slack block* for therapist k if and only if the following three conditions are satisfied: 1) $r_a \in \{o(k)\} \cup I_{\text{FIX}}$, i.e., the first patient is either the origin-location or a fixed patient; 2) $r_b \in \{d(k)\} \cup I_{\text{FIX}}$, i.e., the last patient is either the destination-location or a fixed patient; and 3) $r_\eta \in I_{\text{FLEX}}$ for all $a < \eta < b$, i.e., all intermediate entries are flexible patients. Letting patient $i = r_\eta \in IR$ be the η^{th} patient in the sequence, we define her slack block as $[a_i, b_i]$ such that $a_i \leq \eta < b_i$. Accordingly, each patient in the sequence belongs to a unique slack block; any modifications to her schedule will only affect the patients that succeed her in the same block. If we modify the start time of patient r_η, for example, we only need to recalculate the visit times of the flexible patients $r_{\eta+1}, \ldots, r_{b_i-1}$ since flexible patients are assigned as early as possible.

Example 8 Referring to the sequence in (8), we have three slack blocks [0,3], [3,8] and [8,10]. For the patient in the fourth position, 2002, her slack block is [3,8]. If we exchange this patient with patient 2010 from another route, we only need to recalculate the visit times for the patients 2003, 2004 and 2005, i.e., the patients in positions 5, 6 and 7. □

4.2.2 Route feasibility after modification

Each of the two neighborhood procedures can be executed in two steps. For an insertion, we remove a patient i from her current position in a route and insert her in another position or in another route. In swapping, we exchange the positions of patients i and j in their respective routes. Thus, for patient i in route (k, d) and patient j from either the same route (k, d) or another route (k_1, d_1), we only need to consider the following three elemental modifications.

Type 1. Remove patient i from the route (9a)

Type 2. Insert another patient j immediately before patient i (9b)

Type 3. Replace patient i with another patient j (9c)

Specifically, an insertion requires modifications (9a) and (9b) in sequence, while swapping requires modification (9c) to be executed twice. In order to ensure route feasibility after modifications (9a)–(9c), two conditions must be satisfied: (i) pattern feasibility—therapist k can treat patient j on day d if the resultant schedule matches one of patient j's patterns, and (ii) time feasibility—therapist k has enough time to treat patient j on day d based on his current schedule.

Pattern feasibility Let patient j be in route (k_1, d_1) and consider a move to route (k_2, d_2) using modifications (9b)–(9c). We need to check whether this move is feasible given the set of patterns $P(j)$ and therapist k's restriction, $K(j, d_2)$. With respect to the latter, therapist k is acceptable as long as $k \in K(j, d_2)$.

Next, we determine if day d_2 is acceptable based on the current schedule. Let p_1 be the pattern for patient j in the current schedule and let $p_1(d_1 \to d_2)$ denote the pattern that is generated by replacing day d_1 with day d_2 in pattern p_1. Since $p_1(d_1 \to d_2)$ is the new pattern for patient j after modification (9b) or (9c), day d_2 is acceptable as long as the $p_1(d_1 \to d_2)$ is feasible for patient *j*, i.e., $p_1(d_1 \to d_2) \in P(j)$. Collectively, the following conditions must be satisfied for a modification to be pattern feasible.

$$k_2 \in K(j, d_2) \tag{10a}$$

$$p_1(d_1 \to d_2) \in P(j) \tag{10b}$$

Example 9 Assume patient *j* requires three treatments during the week, each on separate days, and that her current pattern is (1, 2, 3). If we want to move the treatment on day 3 to day 2, the pattern becomes (1, 2, 2) which is infeasible. Thus, starting from the current feasible solution, patient j is not compatible with route $(k, 2)$ for all $k \in K$. □

The requirement that a treatment be provided by a PT can easily be handled here. In the case where the first visit must be a PT, we need to check whether day d_2 becomes the first visit day for *j*. If d_2 is the first visit day, then therapist k_2 must be a PT.

Time feasibility For route (k, d) with sequence $IR = \{r_0, r_1, \ldots, r_N\}$, if patient i is in position η and has slack block $[a_i, b_i]$, then any modification to her schedule will affect the visit times of patients $r_{\eta+1}, \ldots, r_{b_i-1}, r_{b_i}$. Starting with patient $r_{\eta-1}$, we need to recalculate the visit times of each of her successors in turn. The time feasibility conditions are designed to ensure that these patients can be visited without violating any time constraints after the modifications in (9) are made. Notice that the visit time for fixed patient r_{b_i} does not have to be recalculated, but we include her here for convenience.

Figure 5 illustrates the role of slack blocks. Block 1 is from a feasible route that is a candidate for an insertion. In particular, we would like to insert patient j immediately before patient 2. Block 2 results and is feasible because it is possible to delay the visit times of patients 2 through 4 without missing the visit of patient 5 who cannot be displaced in time. If patient j's treatment time is too large, however, Block 3 results, which is infeasible because the therapist cannot arrive on time for fixed patient 5.

We now consider time feasibility for a general sequence of patients $i_0, i_1, \ldots, i_N$. Given the first patient i_0, we can calculate the visit time for patient i_1 with Eq. (4) and then

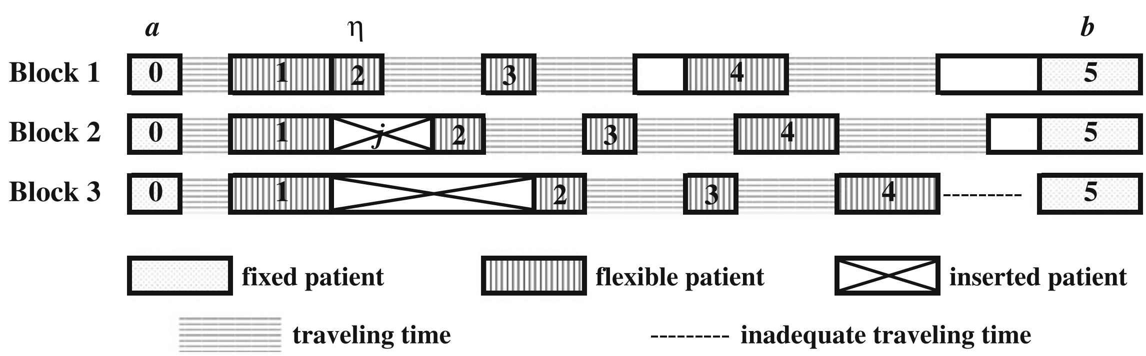


Fig. 5 Time feasibility within a slack block

determine whether patient i_1 is a feasible successor. If patient i_1 fails to satisfy the necessary conditions, we terminate consideration of the sequence because it cannot be feasible; otherwise we continue until an infeasible patient is encountered or all patients have been updated. The general condition is as follows.

$$\text{Patient } i_\eta \text{ is a feasible successor for patient } i_{\eta-1},\quad \eta = 1, \ldots, N \tag{11}$$

Specifically, we have three separate cases with respect to (9a)–(9c):

$$\text{Type 1. } (i_0, i_1 \ldots, i_N) = (r_{\eta-1}, r_{\eta+1} \ldots, r_{b_i}) \tag{12a}$$

$$\text{Type 2. } (i_0, i_1 \ldots, i_N) = (r_{\eta-1}, j, r_\eta, r_{\eta+1}, \ldots, r_{b_i}) \tag{12b}$$

$$\text{Type 3. } (i_0, i_1 \ldots, i_N) = (r_{\eta-1}, j, r_{\eta+1} \ldots, r_{b_i}) \tag{12c}$$

A modification must lead to a solution that is be both pattern and time feasible for it to be in the neighborhood of the current solution.

Example 10 Considering the route in (8), if patient j is to be inserted before 2004, we have $(i_0, i_1 \ldots, i_N) = (2003, j, 2004, 2005, 1002)$. This modification gives visit times (12.50, 13.25, 13.75, 14.50, 14.00) and idle times (0.0, 0.0, 0.0, 0.0, –1), which indicates a time infeasibility according to (11). In particular, patient 1002 cannot be the successor of patient 2005 in the new schedule. □

Any modification to a route might also affect the lunch time for a therapist and one or more of the multiple sessions for a patient. The requirement for multiple sessions can be handled in the same way as presented in Sect. 4.1.3. If lunch is included in the original sequence $(i_0, i_1 \ldots, i_N)$, then we need to reset its start time the same way we shifted patient start times.

4.2.3 Cost efficiency of modification

After route (k, d) is modified, we need to evaluate the benefit of the corresponding move, i.e., we need to calculate the cost of the new solution and compare it with the cost of the current solution to see if a reduction results. As mentioned, the total cost of a schedule consists of four parts: regular cost including treatment and administrative costs, travel cost, traveling mileage reimbursement, and overtime cost. The treatment cost c^k_{id} for patient i on day d is a function of the therapist and treatment time and not the sequence. The travel cost c^k_{ij} does depend on the route sequence or at least the successor j. Mileage reimbursement is a function of the total mileage for therapist k on day d. Letting $\theta^{\text{total}}_{dk}$ be the total mileage, the corresponding reimbursement is,

$$\text{Cost}^{\text{mileage}}_{dk}\left(\theta^{\text{total}}_{dk}\right) = 0.28 \cdot \max\left\{\theta^{\text{total}}_{dk} - 25, 0\right\},\quad \forall k \in K, d \in DK(k) \tag{13}$$

Finally, overtime cost is a function of the total time that therapist k is on the clock over the entire planning horizon. Letting T^{total}_k be the total working time and c^{reg}_k the hourly wage rate for therapist k (as introduced in Sect. 3.2), the corresponding overtime cost is,

$$\text{Cost}^{\text{overtime}}_{k}\left(T^{\text{total}}_{k}\right) = 0.5 \cdot c^{\text{reg}}_k \cdot \max\left\{T^{\text{total}}_{k} - 40, 0\right\},\quad \forall k \in K. \tag{14}$$

When evaluating the benefit of a route modification, some or all of these costs must be taken into account.

To describe the change in cost that results from modifying route (k, d), we make use of the following additional notation.

$\Delta^{\text{regular}}_{(k,d)}$	Regular cost increment for route (k, d)
$\Delta^{\text{travel}}_{(k,d)}$	Traveling cost increment for route (k, d)
$\frown{\theta}_{dk}$	Mileage reimbursement increment for route (k, d)
$\frown{T}_k$	Billable time increment for therapist k
$\Delta^{\text{mileage}}_{(k,d)}$	Mileage reimbursement increment for route (k, d)
$\Delta^{\text{overtime}}_{k}$	Overtime cost increment for therapist k
$\Delta^{\text{cost}}_{(k,d)}$	Total cost increment

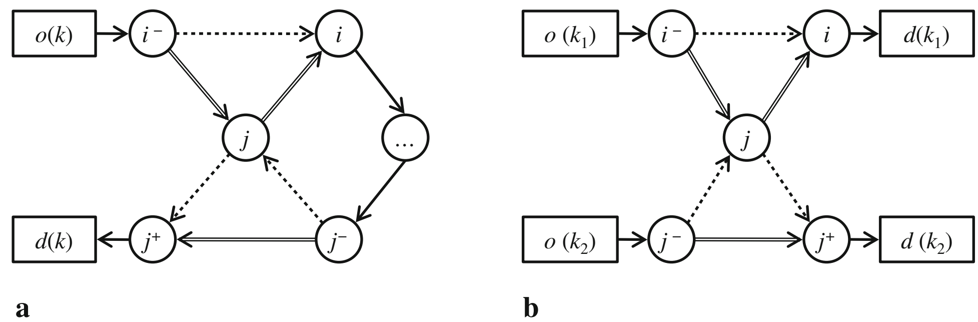


Fig. 6 Two types of insertion neighborhoods. **a** Within route (k, d). **b** Between routes (k_1, d_1) and (k_2, d_2)

Using Eqs. (13)–(14) gives,

$$\Delta^{\text{mileage}}_{(k,d)} = \text{Cost}^{\text{mileage}}_{dk}\left(\theta^{\text{total}}_{dk} + \widehat{\theta}_{dk}\right) - \text{Cost}^{\text{mileage}}_{dk}\left(\theta^{\text{total}}_{dk}\right) \tag{15}$$

$$\Delta^{\text{overtime}}_{k} = \text{Cost}^{\text{overtime}}_{k}\left(T^{\text{total}}_{k} + \widehat{T}_{k}\right) - \text{Cost}^{\text{overtime}}_{k}\left(T^{\text{total}}_{k}\right) \tag{16}$$

with the total cost increment given by

$$\Delta^{\text{cost}}_{(k,d)} = \Delta^{\text{regular}}_{(k,d)} + \Delta^{\text{travel}}_{(k,d)} + \Delta^{\text{mileage}}_{(k,d)} + \Delta^{\text{overtime}}_{k} \tag{17}$$

When $\Delta^{\text{cost}}_{(k,d)} < 0$, we say the modification is *cost efficient* since the new solution provides an improved schedule. Given a cost reduction threshold Δ^{thres}, if $-\Delta^{\text{cost}}_{(k,d)} \geq \Delta^{\text{thres}}$, then the move is accepted. Using Eqs. (15)–(17), the incremental benefit of the new solution can be evaluated quickly in $O(1)$ time by keeping track of the regular cost increment $\Delta^{\text{regular}}_{(k,d)}$, travel cost increment $\Delta^{\text{travel}}_{(k,d)}$, mileage increment $\widehat{\theta}_{dk}$ and total time increment $\widehat{T}_k$ for each move.

4.2.4 Insertion

In the neighborhood defined by an insertion, a new feasible solution is constructed by either moving a patient to another position in the same route or to a new position in another route. Figure 6 illustrates these two cases, where the dotted arcs are in the original solution and the double-line arcs are in modified solution. In Fig. 6a, patient j is moved from a position between patients j^- and j^+ to a position between patients i^- and i in the same route (k, d). In Fig. 6b, j is moved from a position between patients j^- and j^+ in route (k_2, d_2) to a position between patients i^- and i in route (k_1, d_1). We need to treat these two cases separately with respect to both route feasibility and cost efficiency.

Case 1: same route In this case, modifications (9a) and (9b) are made to the same route (k, d) so pattern feasibility is automatically maintained. Checking time feasibility, however, can be complicated. If the sequences defined in (12a) and (12b) either overlap or are consecutive, we need to merge them and check conditions (11) for the resultant sequence. Otherwise, they are treated separately; the insertion is feasible only if both sequences satisfy the time feasibility conditions (11).

Fortunately, the benefit of a same-route insertion can be evaluated quickly. Because treatment time and administrative time in route (k, d) remain unchanged, the regular cost increment is zero, i.e., $\Delta^{\text{regular}}_{(k,d)} = 0$. Regarding the other factors, the travel time increment is

$$\Delta^{\text{travel}}_{(k,d)} = c^k_{i^-j} + c^k_{ji} + c^k_{j^-j^+} - c^k_{i^-i} - c^k_{j^-j} - c^k_{jj^+}$$

while the mileage and total time increment are:

$$\widehat{\theta}_{dk} = \theta_{i^-j} + \theta_{ji} + \theta_{j^-j^+} - \theta_{i^-i} - \theta_{j^-j} - \theta_{jj^+}$$

$$\widehat{T}_k = \tau_{i^-j} + \tau_{ji} + \tau_{j^-j^+} - \tau_{i^-i} - \tau_{j^-j} - \tau_{jj^+}$$

Now, using Eqs. (15)–(17) to calculate the cost increment $\Delta^{\text{cost}}_{(k,d)}$, we define the benefit of the insertion as benefit$= -\Delta^{\text{cost}}_{(k,d)}$.

Case 2: different routes In this case, patient j is moved from route (k_2, d_2) to route (k_1, d_1) by applying modification (9a) to route (k_2, d_2), and modification (9b) to route (k_1, d_1). For route (k_1, d_1), both the pattern feasibility conditions (10) and the time feasibility conditions (11) need to be verified. For route (k_2, d_2), we only need to check the time feasibility conditions (11).

To evaluate the benefit of an insertion, we need to calculate the cost increments for the two routes. Their corresponding regular cost increments, travel cost increments, mileage increments, and total time increments are as follows:

$$\Delta^{\text{regular}}_{(k_1,d_1)} = c^{k_1}_{jd_1}; \quad \Delta^{\text{regular}}_{(k_2,d_2)} = -c^{k_2}_{jd_2}$$

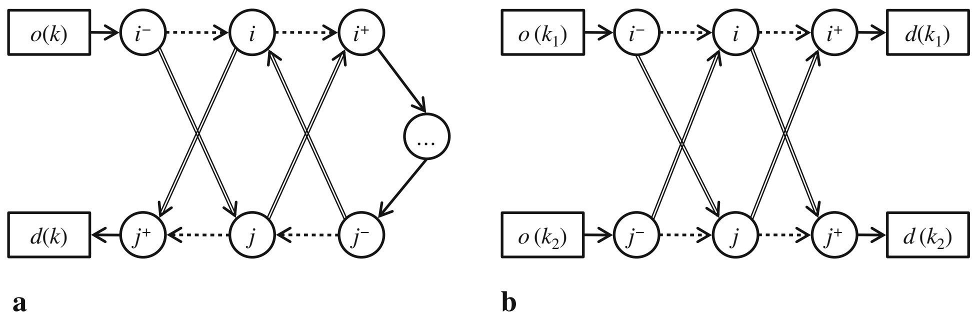
$$\Delta^{\text{travel}}_{(k_1,d_1)} = c^{k_1}_{i^-j} + c^{k_1}_{ji} - c^{k_1}_{i^-i};$$

$$\Delta^{\text{travel}}_{(k_2,d_2)} = -c^{k_2}_{j^-j} - c^{k_2}_{jj^+} + c^{k_2}_{j^-j^+}$$

$$\widehat{\theta}_{d_1k_1} = \theta_{i^-j} + \theta_{ji} - \theta_{i^-i};$$

$$\widehat{\theta}_{d_2k_2} = -\theta_{j^-j} - \theta_{jj^+} + \theta_{j^-j^+}$$

Fig. 7 Two swap neighborhoods. **a** Within route (k, d). **b** Between routes (k_1, d_1) and (k_2, d_2)



$$\widehat{T}_{k_1} = s_{jd_1} + \tau_{i^-j} + \tau_{ji} - \tau_{i^-i};$$
$$\widehat{T}_{k_2} = -s_{jd_2} - \tau_{j^-j} - \tau_{jj^+} + \tau_{j^-j^+}$$

If the routes are associated with different therapists, i.e., $k_1 \neq k_2$, then we define the benefit of the insertion using Eqs. (15)–(17) to be benefit$= -\left(\Delta^{\text{cost}}_{(k_1,d_1)} + \Delta^{\text{cost}}_{(k_2,d_2)}\right)$. If $k = k_1 = k_2$, then the total time increment for therapist k is $\widehat{T}_k = \widehat{T}_{k_1} + \widehat{T}_{k_2}$. Again using Eqs. (15)–(17), we define the benefit of insertion as benefit$= -\left(\Delta^{\text{regular}}_{(k_1,d_1)} + \Delta^{\text{travel}}_{(k_1,d_1)} + \Delta^{\text{mileage}}_{(k_1,d_1)} + \Delta^{\text{regular}}_{(k_2,d_2)} + \Delta^{\text{travel}}_{(k_2,d_2)} + \Delta^{\text{mileage}}_{(k_2,d_2)} + \Delta^{\text{overtime}}_{k}\right)$.

4.2.5 Swapping

In the neighborhood defined by swapping, we construct a new feasible solution by interchanging two patients, either within the same route or between routes. Figure 7 illustrates the two cases, where the dotted arcs represent the original solution and the double-line arcs correspond to the modified solution.

In Fig. 7a, patient i exchanges position with patient j within the same route (k, d). In Fig. 7b, patient i in route (k_1, d_1) changes position with patient j in route (k_2, d_2). Again, these two cases need to be treated separately with respect to route feasibility and cost efficiency.

Case 1: same route In this case, modification (9c) is performed twice within route (k, d): once when patient i who is between patients i^- and i^+ is replaced by patient j, and once when patient j who is between patients j^- and j^+ is replaced by patient i. Using Eq. (12c), we can determine the corresponding sequence.

As with an insertion, pattern feasibility is also automatically maintained here. With respect to time feasibility, if the two new sequences either overlap or are consecutive, we merge them into one and check the time feasibility conditions (11) for the resultant sequence. Otherwise, we consider each separately.

The benefit calculation is again straightforward. Because treatment time and administrative time in route (k, d) remain unchanged, the regular cost increment $\Delta^{\text{regular}}_{(k,d)} = 0$, the travel time increment is

$$\Delta^{\text{travel}}_{(k,d)} = c^k_{i^-j} + c^k_{ji^+} + c^k_{j^-i} + c^k_{ij^+} - c^k_{i^-i} - c^k_{ii^+} - c^k_{j^-j} - c^k_{jj^+}$$

and the mileage increment and total time increment are respectively:

$$\widehat{\theta}_{dk} = \theta_{i^-j} + \theta_{ji^+} + \theta_{j^-i} + \theta_{ij^+} - \theta_{i^-i} - \theta_{ii^+} - \theta_{j^-j} - \theta_{jj^+}$$
$$\widehat{T}_k = \tau_{i^-j} + \tau_{ji^+} + \tau_{j^-i} + \tau_{ij^+} - \tau_{i^-i} - \tau_{ii^+} - \tau_{j^-j} - \tau_{jj^+}$$

Using Eqs. (15)–(17), we now calculate the cost increment $\Delta^{\text{cost}}_{(k,d)}$ and define the benefit of the swap as benefit$= -\Delta^{\text{cost}}_{(k,d)}$.

Case 2: different routes Here, patient i in route (k_1, d_1) is replaced by patient j and patient j in route (k_2,d_2) is replaced by patient i, so modification (9c) is applied once for each route. To ensure route feasibility, both the pattern feasibility conditions (10) and time feasibility conditions (11) must be satisfied for routes (k_1, d_1) and (k_2, d_2).

To evaluate the benefit of a swap, we need to calculate the cost increments for both routes. The corresponding regular cost increments, travel cost increments, mileage increments, and total time increments are as follows.

$$\Delta^{\text{regular}}_{(k_1,d_1)} = c^{k_1}_{jd_1} - c^{k_1}_{id_1}, \quad \Delta^{\text{regular}}_{(k_2,d_2)} = c^{k_2}_{id_2} - c^{k_2}_{jd_2}$$
$$\Delta^{\text{travel}}_{(k_1,d_1)} = c^{k_1}_{i^-j} + c^{k_1}_{ji^+} - c^{k_1}_{i^-i} - c^{k_1}_{ii^+},$$
$$\Delta^{\text{travel}}_{(k_2,d_2)} = c^{k_2}_{j^-i} + c^{k_2}_{ij^+} - c^{k_2}_{j^-j} - c^{k_2}_{jj^+}$$
$$\widehat{\theta}_{d_1k_1} = \theta_{i^-j} + \theta_{ji^+} - \theta_{i^-i} - \theta_{ii^+},$$
$$\widehat{\theta}_{d_2k_2} = \theta_{j^-i} + \theta_{ij^+} - \theta_{j^-j} - \theta_{jj^+}$$
$$\widehat{T}_{k_1} = s_{jd_1} - s_{id_1} + \tau_{i^-j} + \tau_{ji^+} - \tau_{i^-i} - \tau_{ii^+},$$
$$\widehat{T}_{k_2} = s_{id_2} - s_{jd_2} + \tau_{j^-i} + \tau_{ij^+} - \tau_{j^-j} - \tau_{jj^+}$$

If the two routes are for different therapists, i.e., $k_1 \neq k_2$, then we define the benefit of the swap by using Eqs. (15)–(17) as benefit $= -\left(\Delta^{\text{cost}}_{(k_1,d_1)} + \Delta^{\text{cost}}_{(k_2,d_2)}\right)$. If $k = k_1 = k_2$, then the total time increment for therapist k is $\widehat{T}_k = \widehat{T}_{k_1} + \widehat{T}_{k_2}$, and using Eqs. (15)–(17), we define the benefit as benefit$= -\Big(\Delta^{\text{regular}}_{(k_1,d_1)} + \Delta^{\text{travel}}_{(k_1,d_1)} + \Delta^{\text{mileage}}_{(k_1,d_1)} + \Delta^{\text{regular}}_{(k_2,d_2)}$ $+\Delta^{\text{travel}}_{(k_2,d_2)} + \Delta^{\text{mileage}}_{(k_2,d_2)} + \Delta^{\text{overtime}}_{k}\Big)$.

4.2.6 Neighborhood search

Because of the complexity involved in generating neighborhood points and checking their feasibility and benefit value, we do not construct a candidate list but instead move to the first feasible point that provides a cost reduction of at least Δ^{thres}, which is treated as an input parameter. Given the new solution, its neighborhood is examined using the same logic. The procedure is repeated by cycling through all patients and their assigned routes until no improvement beyond Δ^{thres} is possible. The corresponding pseudocode is given in Fig. 8.

5 Computations for real instances

The first set of experiments centered on five datasets provided by Key Rehab, each reflecting the requirements for a single week of operations in Eureka and Wichita, Kansas in May, November and December 2010, and March and April 2011 (all input and output data sets are available from the authors). Table 4 lists the patient characteristics for the selected weeks. Column 1 identifies the instance; the next four columns report the number of fixed patients, their demand, the average number of visits per week and the number of fixed patients that require a PT on the first visit. The remaining five columns report the same information for flexible patients as well as the number who require two visits per day. For example, for the May dataset, there are 43 fixed patients, 7 of whom must be treated by a PT during the first visit. In contrast, there are 22 flexible patients, 5 of whom must be scheduled for two sessions a day, and 10 of whom must be treated by a PT during the first visit.

Table 5 provides the average characteristics of the therapists. For the most part, the column headings are self-explanatory. For the May dataset, for example, there are 3 PTs whose average wage is \$43.33/h and who are available an average of 3.67 days/week. In each of the three datasets, there is only one PTA.

The proposed GRASP was implemented in C++ and run on a PC with 12 GB of memory and an Intel i7 950 processor that has four 3.06 GHz cores. After extensive experimentation, the following parameter settings were seen to provide the best balance between algorithmic performance and solution quality.

- Number of iterations = 250; frequency of Phase II iterations = 5
- Number of candidate routes in the RCL, $m_r = 10$ [Sect. 4.1.2, Step 2]
- Parameters to determine length of RCL, $l_{\min} = 1$, $l_{\text{std}} = 10$, $l_{\max} = 20$ [Sect. 4.1.3, Step 1]
- Number of the top candidate patients considered in RCL, $m_p = 3$ [Sect. 4.1.3, Step 4]
- Benefit function, $\alpha = 1$, $\beta = 2$, $\varepsilon = 0.0001$ [Eq. (6)]
- Improvement threshold for accepting Phase II move, $\Delta^{\text{thres}} = 1$
- No time limit

Tables 6 and 7 summarize the weekly costs normalized with respect to therapist type for the GRASP runs and the schedules provided by Key Rehab, respectively. To put the comparisons on an equal footing, we developed a separate C++ code that took the daily routes actually followed by each therapist and calculated their corresponding costs using the same mileage, travel time, and administrative formulas used in the GRASP. (These individual cost figures are not available from Key's payroll system.)

Tables 6 and 7 breaks down the costs by therapist type and category; i.e., treatment time, travel time, administrative time, overtime, and reimbursed miles. The costs associated with treatment dominate each of the others by more than 50 % and are somewhat higher for the Key Rehab schedule. An item-by-item comparison further indicates the improvements realized in the GRASP solution. This is highlighted in the last columns which report the total weekly costs summed over all therapists. A quick calculation shows cost reductions of 16.78, 11.20, 26.57 16.92, and 18.95 %, respectively, for the five weeks investigated.

Table 8 summarizes the percentage differences by category. In all cases, it can be seen that the GRASP schedules are superior. For the May and November datasets, the overtime reductions for PTs stand out. In these two months, the overtime costs for Key are quit large due to a disproportionate number of patients assigned to PT 1. The last column reports the improved efficiency of the GRASP solutions over the Key Rehab solutions, and is based on the following definition: Eff = treatment cost/total cost. This measure reflects Key's principal mission. It can be interpreted as the fraction of the total cost that is "unavoidable" or direct. All other costs can be viewed as indirect and to some extent avoidable under ideal conditions. For the December dataset, treatment costs decreased by 16.15 % while efficiency increased by 14.19 %. The results were similar for the other months except that the reduction in treatment costs was marginal for May and November.

The performance of the GRASP is summarized in Table 9, which breaks down the computations by phases. For the November dataset, for example, the best solution found in Phase I was \$5316 at iteration 153. Phase II improved the average Phase I solution by 49.89 % and uncovered the best overall solution, \$4836, at iteration 47, which was 72.33 % better than the best Phase I solution. Preprocessing includes

```
Procedure     Neighborhood_Search
Input         feasible solution X_initial
              set of visits V = {vst_1, vst_2, …, vst_|V|} \\ combination of route and patient
Output        local optimum X_opt
Step 0: Initialization
          Set the current solution as the initial solution, i.e., X_curr ← X_initial;
          Set the local optimum as the initial solution, i.e., X_opt ← X_initial;
          Set the initial visit location vst* = vst_1;
Step 1: Find the first neighborhood solution that offers a benefit of no less than Δ^thres
        benefit_new = 0;
        For (patient i in visits vst^i = vst*,…,vst_|V|, vst_1, …, vst*) {
                For (patient j in visits vst^j = vst_1, vst_2, …, vst_|V|){
                        Construct new solution X_new by inserting patient i before patient j;
                        Check insertion feasibility;
                        Calculate insertion benefit benefit_new;
                        If (insertion feasible and benefit_new > Δ^thres){
                                X_opt ← X_new;
                                vst* ← next visit of vst^i;
                                Goto Step 2;
                        }
                }
        }
        For (patient i in visits vst^i = vst*, …, vst_|V|, vst_1, …, vst*) {
                For (patient j in visits vst^j = vst_1, vst_2, …, vst_|V|){
                        Construct new solution X_new by swapping patient i and patient j;
                        Check swapping feasibility;
                        Calculate swapping benefit benefit_new;
                        If (insertion feasible and benefit_new > Δ^thres){
                                X_opt ← X_new;
                                vst*← next visit of vst^i;
                                Goto Step 2;
                        }
                }
        }
Step 2: Updating
        If (benefit_new > 0) {
                X_curr ← X_opt; update vst*;
                Goto Step 1;
        }else{
                Return X_opt;
                Stop;
        }
```

Fig. 8 Pseudocode for neighborhood search

data input, network construction, and time calculations for all feasible links. These steps took less than a second. The total time for the 250 iterations was 343.2 s (see next to last column), which was roughly split 50–50 % on average between the two phases. Various experiments showed that the total runtime scaled linearly with the number of iterations, and that for instances comparable in size to problem nos. 1–5, the best Phase II solution was always found within 250 iterations.

Table 4 Patient characteristics of Key Rehab input datasets

Prob. no.	Fixed patients				Flexible patients				
	No.	No. visits	Avg. no. visits/week	No. first visit	No.	No. visits	Avg. no. visits/week	No. two visits/day	No. first visit
1. May 2010	43	79	1.84	7	22	61	2.77	5	10
2. Nov 2010	28	62	2.21	5	13	48	3.69	5	4
3. Dec 2010	21	36	1.71	3	17	57	3.35	6	4
4. Mar 2011	33	60	1.82	2	44	167	3.80	7	20
5. Apr 2011	35	64	1.83	7	78	214	2.74	0	6

Table 5 Therapist characteristics of Key Rehab input datasets

Prob. no.	PT					PTA				
	No.	Avg. wage/h	Avg. no. days/week	Avg. h/week	Avg. prod. (%)	No.	Avg. wage/h	Avg. no. days/week	Avg. h/week	Avg. prod. (%)
1. May 2010	3	43.33	3.67	27.33	73	1	25.0	5	42.5	87
2. Nov 2010	3	43.33	4.00	30.08	73	1	25.0	5	42.5	87
3. Dec 2010	2	45.65	5.00	42.50	79	1	26.5	5	42.5	87
4. Mar 2011	4	44.33	4.25	29.38	71.75	5	29.4	5	36.3	80.2
5. Apr 2011	5	44.36	4.20	29.30	73.40	7	28.4	5	42.1	80.9

Table 6 Average costs per week for each therapist type for GRASP ($)

Prob. no.	PT						PTA						Grand total
	Treat.	Travel	Admin.	OT	Miles	Total	Treat.	Travel	Admin.	OT	Miles	Total	
1. May 2010	680.83	366.42	257.71	18.49	88.76	1,412.21	787.50	105.94	117.67	5.56	14.98	1,031.65	5,268.29
2. Nov 2010	642.22	350.11	245.38	0.00	86.38	1,324.09	647.92	99.00	96.82	0.00	20.13	863.87	4,836.14
3. Dec 2010	401.01	153.44	102.49	0.00	32.23	689.17	901.44	24.32	134.70	0.23	0.86	1,061.55	2,439.90
4. Mar 2011	265.42	41.73	142.92	0.00	3.47	453.54	660.75	111.30	145.79	0.37	18.82	937.03	6,499.28
5. Apr 2011	164.27	75.59	82.70	0.00	13.00	335.56	614.10	144.59	124.20	0.00	34.47	917.36	8,099.32

Comparing the entries in columns 5 and 12 of Table 9 (best solutions), we see that Phase II greatly improved the Phase I solutions in all cases. Column 9 indicates that the insertion move is much more effective than swaps. For the May dataset, the feasible region is extremely tight making it difficult for the GRASP to construct a solution without using of a supertherapist. On Friday there is only one feasible schedule but it was not uncovered in Phase I. The best solution found in Phase I was at iteration 7 and required a supertherapist. Phase II uncovered several feasible solutions, the best one at iteration 4. Recall that the complexity of finding a feasible solution to routing problems with time windows is the same as finding an optimal solution.

Finally, it should be mentioned that we tried to solve model (1), omitting lunch breaks but including overtime without success using CPLEX 12.2. We made numerous runs with different parameter settings that involved changing the emphasis, omitting cuts, and ordering the variables for branching, but in all cases no feasible solutions were found within 15 h. We then tried solving five 1-day problems for each instance but quickly ran out of memory. In some of those cases, though, we were able to find integer solutions with optimality gaps in the range of 8–75 %. Consequently, as part of the comparisons, we decided to report the LP solution of the full problem obtained at the root node of the search tree. These values represent a lower bound on the GRASP solution and are contained in the last column of Table 9. The corresponding gaps are 26.6, 7.2, 10.0, 19.8 and 29.5 % for May, November, December, March and April, respectively, averaging 18.6 %.

Also detailed in Table 9 are GRASP performance statistics, including time per iteration, total time, the iteration at which the best solution was found, and best result, each for Phase I and Phase II. As can be seen, the runtimes are the same order of magnitude but there is substantial improvement between the Phase I and Phase II solutions. For the 50

Table 7 Average costs per week for each therapist type for Key Rehab ($)

Prob. no.	PT						PTA						Grand total
	Treat.	Travel	Admin.	OT	Miles	Total	Treat.	Travel	Admin.	OT	Miles	Total	
1. May 2010	777.50	430.76	320.69	153.19	96.07	1,778.21	606.25	243.77	90.59	0.00	55.62	996.23	6,330.84
2. Nov 2010	689.44	426.77	272.01	57.21	99.67	1545.1	568.75	157.19	84.99	0.00	0.00	810.92	5,446.24
3. Dec 2010	757.68	298.04	207.48	0.00	61.76	1,324.96	516.31	66.13	77.15	0.00	13.46	673.04	3,322.96
4. Mar 2011	466.14	185.50	196.73	0.00	31.10	879.48	577.94	112.70	165.12	0.00	5.17	860.94	7,822.60
5. Apr 2011	397.79	211.18	163.57	0.00	18.47	791.01	529.84	142.62	138.32	22.17	29.70	862.65	9,993.58

Table 8 Comparison of primary performance measures: 100 × (Key Rehab—GRASP) / Key Rehab (%)

Prob. no.	Treat. cost	Travel cost	Admin. cost	OT cost	Miles cost	Total cost	Improved Eff.[a]
1. May 2010	3.70	21.54	15.38	86.72	18.20	16.78	15.72
2. Nov 2010	2.37	20.05	7.55	100.00	6.60	11.20	9.95
3. Dec 2010	16.15	49.99	30.97	–	52.31	26.57	14.19
4. Mar 2011	8.18	44.59	19.34	–	28.13	16.92	10.52
5. Apr 2011	10.14	32.33	28.17	100.00	−2.01	18.95	10.88

[a] 100 × (GRASP—Key Rehab)/Key Rehab (%)

Phase II iterations, for example, the average improvement for November was 49.89 %, while at the best iteration the improvement was 72.33 %. Recall that in Phase II we try to improve the best solution obtained in the 5 most recent iterations.

Table 10 summarizes the performance of the sequential GRASP and our previously developed parallel GRASP. Both employ similar Phase II logic but the Phase I procedures are radically different. In four out of the five cases, the former provided a noticeable cost reduction overall averaging 5.67 %. For problems 1–3, parallel GRASP was 32.61 % faster on average but for the largest two problems it was 117.97 % slower. We will see in the next section that as the size of the problem grows, the sequential GRASP dominates with respect to both cost and time.

6 Experiments with random data

For general testing purposes, we developed random datasets by varying the three most critical parameters: the number of therapists (num_TH), the number of patients (num_PN), and the ratio between fixed and total patients (ratio_FF). The values considered were as follows: num_TH $\in$ {4,8,12, 16}, num_PN = 10 × num_TH, and ratio_FF $\in$ {0.3,0.4,0.5,0.6}, giving a total of 16 problem sets since the second parameter depends on the first. For each set, we generated five instances by randomly selecting the therapists and patients from their real counterparts. The database contained 16 therapists, 109 flexible patients and 106 fixed patients. When selecting a therapist, we sampled four components: cost-related data, address, working time window, and preference for patient location such as nursing home, hospital or residence. When selecting a patient, we did not sample any components but instead took all attributes as given. These included treatment time, location, first visit indicator, and appointment time when relevant. Duplicates were discarded. In total, we created and solved 80 (= 16 × 5) instances. Again, all input and output datasets are available from the authors.

The advantage of this sampling procedure is that it allowed us to generate a variety of instances that reflect real-world conditions while limiting their number. We felt that if we also considered the less significant parameters such as the number of working days and hours per day for each therapist, wage rates, the number of visits per day for each patient, the therapist type, and the first visit requirement, this would have vastly complicated the experimental design without adding to our knowledge of algorithmic performance.

In the computations, we again allowed 250 Phase I iterations for all instances. For the 4- and 8-therapist problem sets, Phase II was called every five iterations as previously but for the 12- and 16-therapist instances, Phase II was called every 10 iterations, a concession aimed at reducing runtimes. Also, the threshold parameter, Δ^{thres}, was increased from 1 to 10 for the 16-threapist instances.

Table 11 summarizes the results obtained with the two algorithms and CPLEX for the random datasets. The first column identifies the size of an instance, where size is specified by: number of therapists × number of fixed patients × number of flexible patients. From the last two fields we can determine ratio_FF. The entries in each row are an average of 5 instances.

Table 9 GRASP performance

Prob. no.	Phase I				Phase II							Overall Time (sec)	CPLEX LP soln. ($)
	Time/iter. (sec)	Total time (sec)	Best iter.	Best soln. ($)	Time/iter. (sec)	Total time (sec)	Avg. no. swaps/ insertions	Avg. % improve/ iter.	Best iter.	% improve at best iter.	Best soln. ($)		
1. May 2010	2.37	593.40	7	8,847.59	11.42	571.20	1.50/33.34	91.89	4	193.84	5268.29	1,164.76	4,160.69
2. Nov 2010	0.80	199.14	154	5,316.47	2.88	143.98	1.10/28.50	49.89	48	72.33	4,836.14	343.20	4,513.25
3. Dec 2010	0.56	139.42	2	2,867.36	1.82	90.80	2.30/29.28	27.99	22	29.78	2,439.90	230.31	2,219.11
4. Mar 2011	10.06	2,515.47	209	6,961.53	88.24	4,411.85	1.22/30.34	8.71	93	10.39	6,499.28	6,928.01	5,424.64
5. Apr 2011	31.87	7,967.38	1	8,699.62	347.57	8,689.15	1.16/46.88	9.17	171	14.41	8,099.32	16,659.21	6,252.33

Table 10 Algorithmic comparisons for real datasets

Prob. no.	Sequential GRASP		Parallel GRASP		CPLEX - LP		Δcost[a] (%)	Δtime[a] (%)
	Cost ($)	Time (sec)	Cost ($)	Time (sec)	Cost ($)	Time (sec)		
1. May 2010	5,268.29	1,164.76	5,563.50	807.68	4,160.69	57.70	5.60	−30.66
2. Nov 2010	4,836.14	343.20	4,819.79	230.21	4,513.25	15.30	−0.34	−32.92
3. Dec 2010	2,439.90	230.31	2,516.73	151.46	2,219.11	63.57	3.15	−34.24
4. Mar 2011	6,499.28	6,928.01	6,688.71	18,963.02	5,424.64	232.13	2.91	173.72
5. Apr 2011	8,099.32	16,659.21	8,962.54	27,023.21	6,252.33	276.16	10.66	62.21

[a] 100 × (Parallel GRASP—Sequential GRASP)/Sequential GRASP

The first observation is that the sequential GRASP offers significant improvement over the parallel GRASP in all cases with respect to cost and in virtually all of cases with respect to time for the 8-, 12- and 16-therapist instances. When the Δcost percentages are above 15 %, as they are in the first four problem sets, for example, it generally means that the parallel GRASP could not find a feasible solution without the use of a supertherapist in at least one of the five instances being averaged. Recall that a supertherapist has a wage rate of $1000/h.

As with the real cases, the parallel GRASP reached the 250 iteration limit much more quickly than did the sequential GRASP for the small instances (4 therapists), but then began to lag. Since the Phase II procedure for both algorithms are similar, the explanation would seem to lie in Phase I and the corresponding amount of data processing and checking that is required to find partial solutions; however, this is only partially true. At each Phase I iteration of the sequential GRASP, a separate route is constructed each day for each therapist. For a given therapist, we need to determine which patients are compatible with him and then construct the candidate list. From the candidate list we adaptively construct the restricted candidate list which ranges in size from 10 to 20 patients, regardless of problem size. Given an average of 2.5 treatment days a week per patient, this translates into 25–50 possible visits per day although fewer may actually be scheduled.

With the parallel GRASP, we first determine the patient patterns in a way that evenly distributes demand over the week. For the 4-therapist instances this translates into about 20 visits per day, which is generally less than for the sequential GRASP, at least for the first few days. At each Phase I iteration, a series of assignment problems are solved that assign patients to therapists each day. This requires an examination of each therapist-patient pair to determine their compatibility before setting up and solving each assignment problem. When an instance is limited to a few therapists, most of the effort is in solving the assignment problems with CPLEX. This can be done very quickly in general. As the problem size grows, setup time begins to dominate, and becomes much more intensive for the parallel GRASP because the number of visits each day increases. In addition, the parallel GRASP includes a step following the solution of each assignment problem that involves route augmentation. This step requires a more detailed assessment of therapist-patient compatibility as well as an assessment of patient–patient compatibility, and hence more work.

In the majority of cases, the Phase I solutions provided by the sequential GRASP are better than those provided by the parallel GRASP. As a consequence, the parallel GRASP spends more time in Phase II (executes more insertions and swaps) than the sequential GRASP improving the Phase I solutions. The 4-therapist instances are the exception and hence the result that the parallel GRASP is somewhat faster for the small instances.

From the cost data in Table 11 we can compute the gap between the relaxed CPLEX solutions and those obtained with the two heuristics. For the sequential GRASP, the gap averaged 44.2 %, and for the parallel GRASP, 74.6 %. These values could only be calculated using the first 12 problems sets which contain 4, 8 and 12 therapists; CPLEX could not find an LP solution for any of the 16-therapist instances within 14 hours. Those instances roughly contain 2 million variables and 1 million constraints and are highly degenerate. Finally, additional tables highlighting input and output statistics in a format similar to Tables 4–9 can be obtained from the authors, as can the details of each of the 80 runs.

7 Summary and conclusions

The TRSP offers new challenges for researchers interested in personnel planning in the home healthcare industry. In this paper, we described a new algorithm for constructing weekly schedules for agency therapists treating patients scattered throughout a wide service area. The collective features of the problem that set it apart from previous work include heterogeneous therapists both full and part time, a mixture of fixed and flexible patients, first visit requirements, multiple sessions per day, and the need to assign lunch breaks and account for contractual overtime agreements.

As part of the research we developed a mixed-integer linear programming model for the problem but quickly discov-

Table 11 Algorithmic comparisons for random datasets

Prob. no.	Sequential GRASP		Parallel GRASP		CPLEX-LP		Δcost[a] (%)	Δtime[a] (%)
	Cost ($)	Time (sec)	Cost ($)	Time (sec)	Cost ($)	Time (sec)		
1. 4 × 12 × 28	5,070.85	717.41	6,891.13	508.46	3,425.17	9.50	35.83	−29.46
2. 4 × 16 × 24	4,895.71	527.57	6,830.65	367.20	3,408.32	7.71	38.63	−29.92
3. 4 × 20 × 20	4,726.89	541.10	6,260.37	350.28	3,221.95	4.42	31.45	−33.47
4. 4 × 24 × 16	4,374.85	383.03	6,652.34	279.12	3,195.74	3.92	50.45	−26.41
5. 8 × 24 × 56	8,934.66	14,383.01	9,130.36	20,199.24	6,099.82	425.93	2.20	44.26
6. 8 × 32 × 48	8,727.97	12,043.44	9,402.48	16,345.22	6,125.39	160.51	7.21	40.26
7. 8 × 40 × 40	9,548.80	10,397.75	10,643.57	13,505.71	6,814.05	239.43	10.35	31.36
8. 8 × 48 × 32	8,237.96	9,892.20	10,537.06	9,176.03	5,693.67	85.95	26.03	−7.68
9. 12 × 36 × 84	12,713.94	60,743.08	13,210.39	106,304.80	8,984.78	10,625.05	3.93	80.56
10. 12 × 48 × 72	12,811.09	52,188.47	15,036.56	81,316.56	8,397.66	3,097.59	17.10	62.09
11. 12 × 60 × 60	12,634.92	40,258.57	13,857.60	84,060.13	8,855.80	2561.02	9.81	108.69
12. 12 × 72 × 48	11,361.44	37,325.95	12,826.25	62,145.48	8,060.58	846.46	12.28	67.96
13. 16 × 48 × 112	17,313.42	137,081.06	17,716.47	261,302.50	b	b	2.50	91.43
14. 16 × 64 × 96	15,468.09	107,896.12	16,165.23	230,588.94	b	b	4.52	118.08
15. 16 × 80 × 80	15,137.90	111,464.76	15,791.88	203,126.49	b	b	4.49	85.67
16. 16 × 96 × 64	15,303.96	84,900.31	15,874.24	156,747.95	b	b	4.05	96.19

[a] 100 × (Parallel GRASP—Sequential GRASP) / Sequential GRASP (%)
[b] No solution after 14 hours

ered that exact solutions were outside the reach of current technology. Even a 1-day instance could not be solved optimally with commercial software. As an alternative we developed a sequential GRASP in an effort to improve upon our previously developed parallel GRASP. Testing on five real datasets and a large number or randomly generated instances confirmed the effectiveness of the new algorithm. In particular, it provided a an average cost reduction of 18.09 % with respect to current practice and a 5.58 % improvement over the parallel GRASP. Moreover, it outperformed the parallel GRASP on most of the randomly generated instances and was able to find solutions for all of the dozen or so unsolved cases.

To further improve performance on the largest instances, which have long runtimes, we are planning to investigate the idea of decomposing the therapists and patients by geographic region so that each can be solved separately. This would require the development of a specialized clustering algorithm. Given the individual solutions, a post-processing step similar to our Phase II procedure could then be used to arrive at a local optimum. On a different track, we are now exploring a column generation algorithm for solving the full problem as well as several special cases defined by either all flexible or all fixed patients.

Appendix 1: Complexity of Phase I and Phase II

Let $|D|$ be the number of days in the planning horizon, $|I|$ the number of patients, $|P|$ the number of feasible patterns, and $|K|$ the number of therapists. Accordingly, there are at most $|D| \cdot |K|$ routes and $O(|I| \cdot |D|)$ visits, where each route has at most $O(|I|)$ patients. To construct a feasible solution in Phase I, the GRASP fixes the first visit day for new patients at the first step, as described in Sect. 4.1.1. This procedure is order $O(|I| \cdot |P|)$ since the first visit day depends on the number of patterns for each patient.

The GRASP then repeatedly selects a route to construct until all patients are scheduled. In this step, as discussed in Sect. 4.1.2, therapists are ordered according to their wage rates using a bubble sort (although faster algorithms are available) which has complexity $O(|K|^2)$. This is done only once. Also, the working hours remaining for the available therapists are estimated for each day based on the visit information. Even though an estimate is made each time the subroutine is called, it only takes $O(|I| \cdot |D|)$ time in total because each patient is only examined twice: first during the initialization step and second when she is assigned to a route. Lastly, we need to sort the days for each route, which takes $O(|D|^2)$ time, and then construct a route for each therapist for each day, giving a complexity of $O(|D|^3 \cdot |K|)$ for this step. Thus, the procedure in Sect. 4.1.2 can be executed in $O(|K|^2 + |I| \cdot |D| + |D|^3 \cdot |K|)$ time.

After a route is selected, a sequence is constructed by selecting one patient at a time as described in Sect. 4.1.3. Given an initial patient in the route, all candidates are sorted according to the distance to the therapist's home base $o(k)$ and then all feasible candidates in RCL are sorted according to their benefits, all of which can be done in $O(|I|^2)$ time. Since a route length is $O(|I|)$, it takes $O(|I|^3)$ to construct a route, or $O(|I|^3 \cdot |D| \cdot |K|)$ for all $|D| \cdot |K|$ routes.

Accordingly, Phase I takes $O(|I| \cdot |P| + |K|^2 + |I| \cdot |D| + |D|^3 \cdot |K| + |I|^3 \cdot |D| \cdot |K|)$ or $O(|I| \cdot |P| + |K|^2 + |D|^3 \cdot |K| + |I|^3 \cdot |D| \cdot |K|)$ time to construct a feasible solution. Thus, a feasible solution can be found in polynomial time.

For each Phase II iteration, we inspect $O(|I|^2 \cdot |D|^2)$ neighborhoods. It takes $O(|I|)$ time to check the feasibility of a candidate move and $O(1)$ to evaluate the benefit of a move so the complexity of each iteration is $O(|I|^3 \cdot |D|^2)$. This implies that the neighborhood of a solution can be generated and evaluated in polynomial time and so each iteration can be implemented efficiently; however, the number of iterations necessary to find a local optimum may be exponential.

References

Baldacci, R., Bartolini, E., Mingozzi, A., & Valletta, A. (2011). An exact algorithm for the period routing problem. *Operations Research*, *59*(1), 228–241.

Bard, J. F., Yu, G., & Arguello, M. F. (2001). Optimizing aircraft routings in response to groundings and delays. *IIE Transactions on Operations Engineering*, *33*(10), 931–947.

Begur, S. V., Miller, D. M., & Weaver, J. R. (1997). An integrated spatial DSS for scheduling and routing home-health-care nurses. *Interfaces*, *27*(4), 35–48.

Bennett, A. R., & Erera, A. L. (2011). Dynamic periodic fixed appointment scheduling for home health. *IIE Transactions on Healthcare Systems Engineering*, *1*(1), 6–19.

Bertels, S., & Fahle, T. (2006). A hybrid setup for a hybrid scenario: Combining heuristics for the home health care problem. *Computers & Operations Research*, *33*(10), 2866–2890.

Boudia, M., Louly, M. A. O., & Prins, C. (2006). A reactive GRASP and path relinking for a combined production–distribution problem. *Computers & Operations Research*, *34*(11), 3402–3419.

Cayirli, T., Veral, E., & Rosen, H. (2008). Assessment of patient classification in appointment system design. *Production and Operations Management*, *17*(3), 338–353.

Cheng, E., Rich, J. L. (1998). A home health care routing and scheduling problem. Tech Report Tr. 98–04, Computational and Applied Mathematics, Rice University, Houston, TX. http://www.caam.rice.edu/tech_reports/1998.html.

Cordeau, J.-F., Gendreau, M., Laporte, L., Potvin, J.-Y., & Semet, F. (2002). A guide to vehicle routing heuristics. *Journal of the Operational Research Society*, *53*(5), 512–522.

Eveborn, P., Flisberg, P., & Rönnqvist, M. (2006). LAPS CARE: An operational system for staff planning of home care. *European Journal of Operational Research*, *171*(3), 962–976.

Festa, P., & Resende, M. G. C. (2002). GRASP: An annotated bibliography. In P. Hansen & C. C. Rebeiro (Eds.), *Essays and surveys on metaheuristics* (pp. 325–326). Dordrecht: Kluwer Academic Publishers.

Festa, P., & Resende, M. G. C. (2009a). An annotated bibliography of GRASP - Part I: algorithms. *International Transactions in Operational Research*, *16*(1), 1–24.

Festa, P., & Resende, M. G. C. (2009b). An annotated bibliography of GRASP—Part II: Applications. *International Transactions in Operational Research*, *16*(2), 131–172.

Feo, T. A., Venkatraman, K., & Bard, J. F. (1991). A GRASP for a difficult single machine scheduling problem. *Computers & Operations Research*, *18*(8), 635–643.

Green, L. V., & Savin, S. (2008). Reducing delays for medical appointments: A queueing approach. *Operations Research*, *56*(6), 1525–1538.

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

Kontoravdis, G., & Bard, J. F. (1995). A GRASP for the vehicle routing problem with time windows. *ORSA Journal on Computing*, *7*(1), 10–23.

Laporte, G. (2009). Fifty years of vehicle routing. *Transportation Science*, *43*(4), 408–415.

Muthuraman, K., & Lawley, M. (2008). A stochastic overbooking model for outpatient clinical scheduling with no-shows. *IIE Transaction on Operations Engineering*, *40*(9), 820–837.

Prais, M., & Ribeiro, C. C. (2000). Reactive GRASP: An application to a matrix decomposition problem in TDMA traffic assignment. *INFORMS Journal on Computing*, *12*(3), 164–176.

Patrick, J., Puterman, M. L., & Queyranne, M. (2008). Dynamic multipriority patient scheduling for a diagnostic resource. *Operations Research*, *56*(6), 1507–1525.

Potvin, J. Y., & Rousseau, J. M. (1995). An exchange heuristic for routing problems with time windows. *Journal of the Operational Research Society*, *46*(12), 1433–1446.

Savelsbergh, M. (1992). The vehicle routing problem with time windows: Minimizing route duration. *ORSA Journal on Computing*, *4*(2), 146–154.

Shao, Y. (2011). *Home therapist network modeling*. Dissertation, Graduate Program in Operations Research & Industrial Engineering, The University of Texas, Austin.

Shao, Y., Bard, J. F., & Jarrah, A. I. (2012). The therapist routing and scheduling problem. *IIE Transactions on Operations Engineering & Analysis*, *44*(11), 868–893

Stansfield, T. C., Massey, R., & Manuel, J. (2011). Life support for hospital staff. *Industrial Engineering*, *43*(2), 28–33.

Taillard, E. D., Badeau, P., Gendreau, M., Guertin, F., & Potvin, J. Y. (1997). A tabu search heuristic for the vehicle routing problem with soft time windows. *Transportation Science*, *31*(2), 170–186.