An Enhanced Iterated Greedy Metaheuristic for the Particle Therapy Patient Scheduling Problem: Instance Format and Preprocessing

Johannes Maschler, Thomas Hackl, Martin Riedler, and Günther R. Raidl

Institute of Computer Graphics and Algorithms, TU Wien
Favoritenstraße 9-11, 1040 Vienna, Austria
{maschler|riedler|raidl}@ac.tuwien.ac.at
t9.hackl@gmail.com

This document describes the format in which the benchmark instances used in [1] are given. Moreover, the applied preprocessing is specified in Section 2.

1 Input Format

The instances are encoded in JSON. Note that the instance format described below allows to state instances for a more general problem, hence it contains elements that are not relevant for the PTPSP. Moreover, in the instance format DTs are specified via activities (as in [2]) that can be trivially transformed into our notation. An instance is represented by the following JSON-object:

- `GENERAL`: object, contains globally relevant problem information
  - `beam-resource-id`: integer, ID of the Beam resource
  - `proton-resource-id`: not relevant for PTPSP
  - `carbon-resource-id`: not relevant for PTPSP
  - `IR-rooms`: array of integers, IDs of the irradiation room resources, not relevant for PTPSP

  - `anesthetist-id`: not relevant for PTPSP
  - `working-days`: array of arrays of objects, working days partitioned into weeks sorted in increasing order of the day index
    * `d`: unique positive integer, index of day $d \in D'$
    * `start`: integer, opening time $\widetilde{W}_d^{\text{start}}$ minutes
    * `end`: integer, closing time $\widetilde{W}_d^{\text{end}}$ minutes
    * `unavailable`: array of objects (optional), global unavailability periods, i.e., unavailability periods for all resources $r \in R$
      · `start`: start time in minutes (is interpreted as $\overline{W}_{r,d,w}^{\text{start}}$ for all $r \in R$)
      · `end`: end time in minutes (is interpreted as $\overline{W}_{r,d,w}^{\text{end}}$ for all $r \in R$)
- `RESOURCES`: array of objects, all resources and their availabilities
  - `id`: unique positive integer, resource ID
  - `name`: string (optional, for debugging purposes), name for resource
  - `scatter`: float (optional), not relevant for PTPSP
  - `W`: array of objects
    * `d`: unique integer, day d
    * `start`: integer (optional, default: $\widetilde{W}_d^{\text{start}}$), start time $W_{r,d}^{\text{start}}$ in minutes
    * `end`: integer (optional, default: $\widetilde{W}_d^{\text{end}}$), end time $W_{r,d}^{\text{end}}$ in minutes
    * `unavailable`: array of objects (optional), unavailability periods
      · `start`: start time $\overline{W}_{r,d,w}^{\text{start}}$ in minutes
      · `end`: end time $\overline{W}_{r,d,w}^{\text{end}}$ in minutes
- `THERAPIES`: array of objects, all therapies with their data
  - `id`: unique positive integer, therapy ID
  - `name`: string (optional, for debugging purposes), name of the therapy
  - `n-twmin`: integer (optional, default: 4), minimum number of treatments per week n_t^{twmin}
  - `n-twmax`: integer (optional, default: 5), maximum number of treatments per week n_t^{twmax}
  - `delta-min`: integer (optional, default: 1), min. number of days between two consecutive DTs δ_t^{min}

- `delta-max`: integer (optional, default: 5), max. number of days between two consecutive DTs $\delta_t^{\max}$
- `daily-treatments`: array of objects, all DTs are given in the required order
  * `id`: unique positive integer, DT ID
  * `name`: string (optional, for debugging purposes), name of DT
  * `d-min`: integer (optional, default: 0), earliest possible day $d_{t,u}^{\min}$
  * `d-max`: integer (optional), latest possible day $d_{t,u}^{\max}$; if not specified or -1 no bound is assumed (an implicit limit is given through the number of considered days)
  * `activities`: array of objects, all activities that must be scheduled in this order at a single day
    · `id`: unique positive integer, activity ID
    · `name`: string (optional, for debugging purposes), name of activity
    · `p`: positive integer, processing time in minutes
    · `resources`: array of integers, ID's of required resources

2 Preprocessing

In this section preprocessing techniques are described that are applied on the instance before the Iterated Greedy (IG) metaheuristics are executed. The first technique aims at tighten the earliest and the latest starting day of DTs. Although not all possibilities are exploited (such as resource availabilities) it should provide tight bounds in practice. The following method is concerned with pruning of resource availabilities within working days.

2.1 Day Bound Preprocessing

It can be assumed, that in general instances provide for $d_{t,u}^{\min}$ and $d_{t,u}^{\max}$ for $u > 0$ only default values, i.e., no bounds are provided. In those cases tighter bounds can be inferred from the earliest and the latest starting day from previous DTs in combination by considering the minimal and the maximal number of allowed days between two consecutive DTs and the minimal and the maximal number of DTs allowed per week. We consider here the pruning of the earliest and the latest starting day of DTs separately.

Algorithm 1 starts by setting $d_{t,0}^{\min}$ to the next working day if $d_{t,0}^{\min} \notin D'$. We set $d_{t,0}^{\min}$ to the first day in the next week if starting at day $d_{t,0}^{\min}$ would violate the minimal number of DTs per week. For each subsequent DT $d_{t,u}^{\min}$ is set to the maximum of the following

for $t \in T$ **do**
  $d_{t,0}^{\min} := \min\{d \in D' \mid d \geq d_{t,0}^{\min}\}$;
  let $v \in V$ be the week index s.t. $d_{t,0}^{\min} \in D'_v$;
  if $|\{d \in D_v \mid d \geq d_{t,0}^{\min}\}| \leq \min(n_t^{\text{twmin}}, |D_v|)$ **then**
    $d_{t,0}^{\min} := \min\{D_{v+1}\}$;
  end
  for $u \in \{1, \ldots, \tau_t - 1\}$ **do**
    $d_{t,u}^{\min} := \max(d_{t,u}^{\min}, d_{t,u-1}^{\min} + \delta_t^{\min})$;
    $d_{t,u}^{\min} := \min\{d \in D' \mid d \geq d_{t,u}^{\min}\}$;
    let $v \in V$ be the week index s.t. $d_{t,u-1}^{\min} \in D'_v$;
    if $|\{d_{t,u'}^{\min} \in D_v \mid u' < u\}| \geq \min(n_t^{\text{twmax}}, |D_v|)$ **then**
      $d_{t,u}^{\min} := \max(d_{t,u}^{\min}, \min\{D_{v+1}\})$;
    end
  end
end

Algorithm 1: Day bound preprocessing for the earliest starting day

values, the given $d_{t,u}^{\min}$ from the instance, $d_{t,u-1}^{\min} + \delta_t^{\min}$, and the first day in the next week if already $\min(n_t^{\text{twmax}}, |D_v|)$ predecessors of the DT have the earliest starting day in the current week.

Algorithm 2 sets $d_{t,0}^{\max}$ to the closest working day smaller than or equal to the $d_{t,0}^{\max}$ given from the input instance that allows at least n_t^{twmin} consecutive DTs in the current week. For each subsequent DT we set first $d_{t,u}^{\max}$ to the minimum of $d_{t,u}^{\max}$ and $d_{t,u-1}^{\max} + \delta_t^{\max}$. Next we check if we can prune $d_{t,u}^{\max}$ further by considering the minimal number of DTs per week. Moreover, $d_{t,u}^{\max}$ is pruned if the latest starting days come close to the time horizon. Finally, we set $d_{t,u}^{\max}$ to the closest working day smaller than or equal to $d_{t,u}^{\max}$.

Note that this preprocessing could be strengthened further by considering resource availabilities. However, the impact of such an extension should be limited.

2.2 Preprocessing of Resource Availabilities

This preprocessing technique utilizes the observation that, due to the structure of the DTs, some resources cannot be used close to global resource availability changes. Hence, some resource availabilities can be pruned.

For example, if the preparation steps that need to be performed before the irradiation take for every DT at least 20 minutes, then the beam resource cannot be used during the first 20 minutes of the considered day and after each global unavailability period.

We start by removing the availability of resources on days on which no DTs can be scheduled that require the considered resource, i.e., we set $W_{r,d} = [\widehat{W}_{r,d}^{\text{start}}, \widehat{W}_{r,d}^{\text{start}})$ and

for $t \in T$ **do**
let $v \in V$ be the week index s.t. $d^{\max}_{t,0} \in D'_v$;
$d^{\max}_{t,0} := \min(d^{\max}_{t,0}, \max(\max\{D_v\} - n^{\text{twmin}}_t, \min\{D_v\}))$;
$d^{\max}_{t,0} := \max\{d \in D' \mid d \leq d^{\max}_{t,0}\}$;
for $u \in \{1, \dots, \tau_t - 1\}$ **do**
$d^{\max}_{t,u} := \min(d^{\max}_{t,u}, d^{\max}_{t,u-1} + \delta^{\max}_t)$;
let $v \in V$ be the week index s.t. $d^{\max}_{t,u-1} \in D'_v$;
if $|\{d^{\max}_{t,u'} \in D_v \mid u' < u\}| \geq n^{\text{twmin}}_t$ **then**
$d^{\max}_{t,u} := \min(d^{\max}_{t,u}, \max\{D_{v+1}\} - n^{\text{twmin}}_t)$;
else
$d^{\max}_{t,u} := \min(d^{\max}_{t,u}, \min\{D_v\} + |\{d^{\max}_{t,u'} \in D_v \mid u' < u\}|)$;
end
$d^{\max}_{t,u} := \min(d^{\max}_{t,u}, n_D - \tau_t + u)$;
$d^{\max}_{t,u} := \max\{d \in D' \mid d \leq d^{\max}_{t,u}\}$;
end
end

Algorithm 2: Day bound preprocessing for the latest starting day

$\widehat{W}_{r,d} = [\widehat{W}^{\text{start}}_{r,d}, \widehat{W}^{\text{start}}_{r,d})$ for all resources $r \in R$ and $d \in D'$ where $\{(t,u) \mid t \in T,\ u \in U_t, r \in Q_{t,u},\ d^{\min}_{t,u} \leq d \leq d^{\max}_{t,u}\} = \emptyset$.

Let $p^{\text{rampup}}_{r,d}$ be a lower bound on the earliest time a resource $r \in R$ might be used on day $d \in D'$ iff all resources become available at the same time.

$$p^{\text{rampup}}_{r,d} = \min_{t \in T,\ u \in U_t\ | r \in Q_{t,u},\ d^{\min}_{t,u} \leq d \leq d^{\max}_{t,u}} P^{\text{start}}_{t,u,r} \qquad \forall r \in R, \forall d \in D^{\text{res}}_r$$

Analogously, let $p^{\text{winddown}}_{r,d}$ be the minimum offset between the latest use of resource r and the end of the DT considering all DTs that might be scheduled on day d.

$$p^{\text{winddown}}_{r,d} = \min_{t \in T,\ u \in U_t\ | r \in Q_{t,u},\ d^{\min}_{t,u} \leq d \leq d^{\max}_{t,u}} (p_{t,u} - P^{\text{end}}_{t,u,r}) \qquad \forall r \in R, \forall d \in D^{\text{res}}_r$$

The resource availabilities can be pruned by setting

$$\begin{aligned}
\widehat{W}_{r,d} :=& \begin{matrix}[\max(\widehat{W}^{\text{start}}_{r,d}, \widetilde{W}^{\text{start}}_{d} + p^{\text{rampup}}_{r,d}), \\ \min(\widehat{W}^{\text{end}}_{r,d}, \widetilde{W}^{\text{end}}_{d} - p^{\text{winddown}}_{r,d}))\end{matrix} && \forall r \in R, \forall d \in D^{\text{res}}_r \\
W_{r,d} :=& [\max(W^{\text{start}}_{r,d}, \widehat{W}^{\text{start}}_{r,d}), \min(W^{\text{end}}_{r,d}, \widehat{W}^{\text{end}}_{r,d})) && \forall r \in R, \forall d \in D^{\text{res}}_r.
\end{aligned}$$

Moreover, let $\overline{\mathcal{W}}_d = \bigcup_{w=0,\dots,\omega_d-1} \overline{\mathcal{W}}_{d,w}$ be the set of global unavailability periods with $\overline{\mathcal{W}}_{d,w} = [\overline{\mathcal{W}}^{\text{start}}_{d,w}, \overline{\mathcal{W}}^{\text{end}}_{d,w}) \subset \widetilde{W}_d,\ w = 0, \dots, \omega_d - 1$, where $\overline{\mathcal{W}}^{\text{start}}_{d,w}$ and $\overline{\mathcal{W}}^{\text{end}}_{d,w}$ denote the start and end times of the w-th global unavailability period. We derive the following (regular) unavailability periods:

$$[\overline{\mathcal{W}}^{\text{start}}_{d,w} - p^{\text{winddown}}_{r,d}, \overline{\mathcal{W}}^{\text{end}}_{d,w} + p^{\text{rampup}}_{r,d}) \qquad \forall r \in R, \forall d \in D^{\text{res}}_r, w = 0, \dots, \omega_d - 1$$

References

[1] J. Maschler, T. Hackl, M. Riedler, and G. R. Raidl. An enhanced iterated greedy metaheuristic for the particle therapy patient scheduling problem. In *Proceedings of the 12th Metaheuristics International Conference*, 2017. to appear.

[2] J. Maschler, M. Riedler, M. Stock, and G. R. Raidl. Particle therapy patient scheduling: First heuristic approaches. In *Proceedings of the 11th International Conference on the Practice and Theory of Automated Timetabling*, pages 223–244, Udine, Italy, 2016.