Models for Radiation Therapy Patient Scheduling

Sara Frimodig[1,2] and Christian Schulte[1]

[1] KTH Royal Institute of Technology, Stockholm, Sweden
[2] RaySearch Laboratories, Stockholm, Sweden
{sarhal,cschulte}@kth.se

Abstract. In Europe, around half of all patients diagnosed with cancer are treated with radiation therapy. To reduce waiting times, optimizing the use of linear accelerators for treatment is crucial. This paper introduces an Integer Programming (IP) and two Constraint Programming (CP) models for the non-block radiotherapy patient scheduling problem. Patients are scheduled considering priority, pattern, duration, and start day of their treatment. The models include expected future patient arrivals. Treatment time of the day is included in the models as time windows which enable more realistic objectives and constraints. The models are thoroughly evaluated for multiple different scenarios, altering: planning day, machine availability, arrival rates, patient backlog, and the number of time windows in a day. The results demonstrate that the CP models find feasible solutions earlier, while the IP model reaches optimality considerably faster.

1 Introduction

Radiation therapy (RT), chemotherapy, and surgery are the most commonly used cancer therapies worldwide. In RT, machines called linear accelerators (LINACs) deliver beams of radiation to the tumor in order to kill malignant tumor cells.

A long waiting time between when a patient is ready for RT and when the treatment starts has a negative effect on its outcome due to for example tumor growth, psychological distress of the patient, and progressed symptoms [8,12,14,22,30]. Hence, many cancer institutes worldwide have adopted waiting time targets that determine the maximum waiting time before treatment starts.

The intent of RT treatments is either curative or palliative, where the latter mainly aims to provide pain relief. Furthermore, cancer patients are generally divided into three urgency levels depending on the site of the cancer, treatment intent, and the size and progress of the tumor. The waiting time targets depend on the patient's urgency level.

RT treatments are generally divided into a number of fractions that are delivered once a day and together sum up to the planned radiation dose. The duration of the fractions vary between patients due to for example treatment technique and tumor complexity [16]. There are also many uncertainties in the RT process, including for example patient inflow and unexpected machine failures.

The scheduling of RT patients on LINACs can be divided into block or non-block systems [10]. Block scheduling systems divide days into slots of equal duration, whereas non-block systems allow for different treatment durations. Block systems are more widely used, but have severe drawbacks since there is no way to control the variability of treatment time, which can generate costs related to machine underutilization, staff overtime, and patient waiting time.

Scheduling patients is mostly done manually and is a considerable challenge for RT clinics. Designing more efficient appointment schedules would be of great significance and could potentially save lives. In order to improve the scheduling of radiotherapy patients, this paper makes the following contributions:

- Two CP models and one IP model of the non-block RT patient scheduling problem are introduced that take expected future patients into account. To the best of our knowledge, these are the very first CP models as well as the first IP model to include expected future patient arrivals.
- The treatment time of the day is included in the models which for the first time supports objectives and constraints on treatment time of the day.
- The models capture real-world constraints such as non-consecutive treatment days, different treatment durations, allowed start days, and patient priorities.
- The models are evaluated and compared using a patient arrival model and several experiments based on data from a European cancer clinic.

Plan of the Paper. Section 2 discusses related work. Section 3 presents the setup of the problem. Section 4 describes the models, followed by a description of the search heuristics used for the CP models in Section 5. The models are evaluated in Section 6. Section 7 presents conclusions and potential extensions.

2 Related Work

Related work on optimal scheduling in health care has mainly focused on nurse scheduling (see [5]), outpatient assignment (see [7]), and surgery scheduling (see [21]). The RT patient scheduling problem shares some characteristics with these problems, but has particular attributes that make it difficult to apply the models and methods proposed in the literature.

Scheduling of RT patients is a relatively young field with limited literature. In 2016, a review on the literature using operations research for resource planning in RT was published [31]. The authors found 12 papers addressing the problem of scheduling patients on LINACs, where the first ones are published in 2006.

In [18], the authors show that RT patient scheduling can be seen as a special case of a dynamic job-shop problem. They review different exact and metaheuristic methods suitable for solving the problem. A heuristic that schedules patients forward from the first feasible start date (ASAP) is developed in [25]. A local search heuristic that outperforms the ASAP approach is developed in [26].

The first use of IP for optimization of RT appointments is presented in [9] and [10]. Another IP model for non-block scheduling is presented in [17]. A limitation of these papers is that they do not consider all the constraints present in RT scheduling, such as for example treatments on non-consecutive days and LINAC eligibility. In [6], the authors develop an IP model that includes more realistic constraints, but still using a myopic scheduling policy, i.e., not taking future patient arrivals into account.

Using a block scheduling strategy, [27] presents a method for advance RT patient scheduling, where appointments are scheduled in advance of the service date with future demand still unknown. A Markov decision process (MDP) and approximate dynamic programming are used to solve the problem, and they achieve very good results. In [13], the authors use the same problem setup but also include patient cancellations using simulation-based solution methods. For these approaches, time of the day for the treatments is impossible to include.

A hybrid combining stochastic and online optimization is presented in [19]. The authors use a block-scheduling strategy to schedule curative patients at the same time every day and require that patients leave the center with their appointment, which calls for short computation times. This is different from earlier published methods that all schedule multiple patients in a batch.

CP has been used in RT treatment planning [2] and in chemotherapy patient scheduling [15]. Scheduling is a field where CP has shown to be effective, see for example [3]. A comprehensive review of operations research methods for optimization in radiation oncology is presented in [11], where it is stated that CP has not yet had a significant impact on medical physics.

3 Radiation Therapy Patient Scheduling

This section introduces the RT patient scheduling problem, the assumptions made in this paper, and some fundamental modeling aspects.

Time. Radiation therapy clinics have different routines for scheduling patients on LINACs. Some gather patients into a *batch* and schedule them once or several times a day, while others immediately schedule a single patient. This paper focuses on batch scheduling and assumes that the scheduling is done at the end of each day taking patients from previous days into account. As previously stated, RT clinics can be divided into two categories; those who use *block* and *non-block* scheduling systems. In order to be able to control the variability of treatment time, this paper uses a *non-block* scheduling strategy.

A day is divided into *time windows*. A time window is typically $1.5-4$ hours while a treatment takes $10-45$ minutes. Patients are assigned to windows instead of specific start times as this leads to simpler and more efficient models while maintaining an adequate level of detail from a clinical perspective.

Patients. A physician assigns a *priority* to each patient based on urgency and treatment intent (palliative or curative). It is assumed that there are three priority groups, and therefore three waiting time targets: 2 days for priority A (the highest), 14 days for priority B, and 28 days for priority C patients.

A patient is assigned to a *treatment protocol*, which states the *fractionation scheme* (that is, how many days the patient is to be treated and with which frequency) and the *duration* of each treatment. Different protocols have different allowed *start days*, which enforces that fewer patients are scheduled on weekends. Some protocols also specify that treatment must start on a *certain time of the day*. In this paper, the protocols used are from a large cancer center in Europe.

The scheduled times are communicated to the patient at most one week before the start date or immediately for priority A patients. All fractions are communicated and cannot be re-planned, as this is the collaboration clinic's approach. The schedule can change until being communicated: booking decisions are postponed to the next day if patients are scheduled more than a week away.

When creating a patient schedule in practice, the booking administrator needs to make sure that there is room in the schedule for more urgent future patients. In most cases, this is done by leaving some empty time on each machine. In the models, the *expected future patient arrivals* are included to predict the expected utilization of resources. Only the expected future patients who have a waiting time target shorter than the maximum waiting time target of current patients are included. This is as patients with longer waiting time targets have little or no effect on the current schedule.

An overall *arrival rate* can be extracted from historical data for each clinic, as well as the proportion of arrivals for each priority group. This paper uses the same proportions between the priorities as [19]: 31% are priority A, 19% are priority B, and 50% are priority C. The proportions can easily be adjusted to a particular clinic. In the models, a separate priority group D is created for expected future patients of priority A, since the actual priority A patients should have higher priority than expected future priority A patients. These patients are also treated differently in the search heuristics for the CP models, see Section 5. Each arriving patient is randomly assigned to a treatment protocol.

Machines. The radiation is delivered on LINACs. As a rule, larger centers have multiple machine types used for different sorts of treatment and multiple identical machines to have a redundancy in case of machine failures. In small centers, there may be a few identical machines that only serve some treatment types, while more complex cases are sent to larger centers.

This paper assumes that there are *multiple machines* but only *one machine type*. The machines are exchangeable in that a patient can be scheduled on any machine each day. This scenario is a realistic way of decomposing the multiple machine problem, since the clinics may consider separate scheduling tasks for each machine type. Instead of M machines with W windows each, this is modeled as having one machine with MW windows. Thus, if for example $M = 3$, $W = 4$, then if a patient is scheduled in window $1 - 4$, this corresponds to machine 1, window $5-8$ is machine 2, and window $9-12$ is machine 3. An alternative would be to model this as one machine with W windows and multiply each window length by M, but this would be a relaxation of the actual problem.

Using multiple machines represents a real-world setting and also allows for having a higher arrival rate. If there were only one machine available, the arrival rate would be very low and dividing very few patients into three different priority groups would not give good statistics for the expected future patient arrivals.

Objective. In this paper, the main objective is to minimize a weighted sum of the violations of the target dates, where the weights reflect that it is worse to violate the target date for a patient with higher priority. The secondary objective is to schedule patients at approximately the same time each day. Some patients may still work or study during treatment and hence prefer mornings or late afternoons. More importantly, the biological effects of the radiation is calculated on having 24 hours between each fraction, however, in most cases it is allowed to deviate from this and it is thus an objective rather than a hard constraint. The second objective is the reason why time windows are used in the models; scheduling a patient in the same time window every day ensures that the treatment is delivered at approximately the same time every day.

4 Models

Three models are developed to capture the RT patient scheduling problem; a scheduling-based CP model, a packing-based CP model, and an IP model. These are designed to capture the same real-world constraints and objectives. Using the set $\mathbb{B} := \{0, 1\}$, the inputs to these models are:

$\mathcal{P} = \{1, \dots, P\}$ set of all patients, $P \in \mathbb{N}$
$\mathcal{D} = \{1, \dots, D\}$ set of days in the planning horizon, $D \in \mathbb{N}$
$\mathcal{W} = \{1, \dots, W\}$ set of time windows in a day, $W \in \mathbb{N}$
$w_L = T_s/W$ the window length, where $T_s \in \mathbb{N}$ is the number of time slots during a day with the chosen discretization
$\mathcal{T}_w = \{t_1, t_2, \dots, t_W\}$ set of times when each window starts counting from the beginning of day 1, where $t_1 = 1, \dots, t_i = (i-1)w_L + 1$
$\mathcal{M} = \{1, \dots, M\}$ set of machines, $M \in \mathbb{N}$
$dur_p \in \mathbb{N}$ duration of a fraction for patient p in time slots
$L_p \in \{13, \dots, 47\}$ schedule length for patient p in days
$\mathcal{F} = \{1, \dots, \max(L_p)\}$ set of all treatment days
$FS_p \in \mathbb{B}^{L_p}$ a vector holding the fractionation schedule for patient p, where ones represent treatment and zeros pause days
$S \in \mathbb{B}^D \times \mathbb{B}^{T_s}$ a matrix holding the partially occupied schedule, where $S_{d,t_s} = 1$ iff time slot t_s on day d is occupied
$\mathcal{A}_p \in \{1, \dots, 7\}$ the set of allowed start days for patient p
$c_p \in \mathbb{N}$ penalty for missing the waiting time target for each priority
$d_{L,p} \in \mathbb{N}$ day limit, i.e., the waiting time target for patient p

4.1 Scheduling-based CP Model

Variables. The basic decision variables of the model are as follows:

$start_{p,f} \in \mathcal{T}_w$ the start time for the window patient $p \in \mathcal{P}$ is scheduled in during treatment day $f \in \mathcal{F}$
$window_{p,f} \in \{0, \dots, W\}$ the window patient $p \in \mathcal{P}$ is scheduled in during treatment day $f \in \mathcal{F}$, where window 0 represents no treatment
$fraction_{p,d} \in \{0, \dots, L_p\}$ fraction that is delivered to patient $p \in \mathcal{P}$ on day $d \in \mathcal{D}$, where fraction 0 represents no treatment

There following variables are derived from the basic variables:

$day_{p,f} \in \mathcal{D}$ day patient $p \in \mathcal{P}$ is treated with fraction $f \in \mathcal{F}$
$start\_day_p = day_{p,1}$ start day for patient $p \in \mathcal{P}$

Constraints. In the scheduling-based CP model, the `cumulative` constraint [1] is used to ensure that no two treatments overlap:

$$\texttt{cumulative}([\langle start_{p,f}, w_L, dur_p \rangle | p \in \mathcal{P}, f \in \mathcal{F}], w_L), \tag{1}$$

where w_L is the window length. The partially occupied input schedule is included as patients with fixed window, start time, and duration to ensure that no new patients are scheduled in a window that is already full. The constraint is used "backwards", setting the duration of each treatment equal to the window length w_L and the resource requirement as the duration of the treatment dur_p.

The variables $day_{p,f}$ and $fraction_{p,d}$ are dual to each other:

$$fraction_{p,d} = f, \text{ where } d = day_{p,f} \quad \forall p \in \mathcal{P}, f \in \mathcal{F}, d \in \mathcal{D}, \tag{2}$$

while $fraction_{p,d} = 0$ if patient p is before the start or after the end of treatment:

$$(d < start\_day_p) \vee (d \geq start\_day_p + L_p) \rightarrow fraction_{p,d} = 0 \quad \forall p \in \mathcal{P}, d \in \mathcal{D}. \tag{3}$$

The day $day_{p,f}$ for fraction $f \in \mathcal{F}$ for patient $p \in \mathcal{P}$ is given by the time when the fraction starts, $start_{p,f}$, divided by the number of time slots T_s. To avoid division, this is expressed as: $(day_{p,f} - 1)T_s + 1 \leq start_{p,f} \leq day_{p,f} T_s$.

The days are connected to each other by the constraint:

$$day_{p,f+1} = day_{p,f} + 1 \quad \forall p \in \mathcal{P}, f \in \mathcal{F},$$

which means that if treatment day f is on day d, then $f + 1$ is on $d + 1$.

Next, connect $window_{p,f}$ to $start_{p,f}$. The vector FS_p is the fractionation schedule for patient p and is input to the problem given by a protocol. $FS_{p,f} = 1$ corresponds to treatment day f being active for patient p, i.e., treatment is delivered that day. Thus, if the input $FS_{p,f}$ is indeed one, this gives

$$start_{p,f} = (day_{p,f} - 1)T_s + (window_{p,f} - 1)w_L + 1, \tag{4}$$

and for all other $p \in \mathcal{P}, f \in \mathcal{F}$, set $window_{p,f} = 0$. This constraint states that the start time for the window patient p is scheduled during fraction f is equal to the start time of that day, plus the start time of the window on that day.

Bounds are also given for when each fraction can start earliest and latest. For example, the patient's second fraction cannot be on the first day, and similarly, the patient's first fraction cannot be on the last day:

$$f \leq day_{p,f} \leq D - (L_p - f) \quad \forall f \in \mathcal{F}. \tag{5}$$

Similar constraints limit the start day for each patient to be at latest L_p days from the end of the planning horizon. Patients have to start treatment on an allowed start day: $start\_day_p \in \mathcal{A}_p \quad \forall p \in \mathcal{P}$.

To break some dominance, patients of the same priority and treatment protocol are sorted by their waiting time target. A constraint enforces that an earlier target patient always starts their treatment before a later target patient.

Objective Function. The first objective is to start each treatment within the waiting time targets. The target violation is measured as the number of days that the patient misses their treatment target date with

$$target\_violation_p = \max(0, start\_day_p - d_{L,p}) \quad \forall p \in \mathcal{P}, \tag{6}$$

where $d_{L,p}$ is the day limit for when treatment should start.

The second objective is to schedule patients in the same window each day. Therefore, a penalty is added each time the window is switched. Since, for example, 3 machines with 4 windows is modeled as 1 machine with 12 windows, there is no penalty for moving from window 1 to 5 or 9, as they are the same but on different machines. An array m_w maps model windows to real machine windows. Only the active treatment days $\mathcal{F}_a$, when $window_{p,f} \neq 0$, are considered:

$$\begin{aligned} window\_diff_p = \sum_{f \in \mathcal{F}_a} (m\_w[window_{p,f}] \neq m\_w[window_{p,f+1}]) \quad \forall p \in \mathcal{P} \\ \text{where } m\_w = [1,2,3,4,1,2,3,4,1,2,3,4] \end{aligned} \tag{7}$$

The two parts of the objective function given by (6) and (7) are combined into a weighted sum. Each entry of $window\_diff_p$ is multiplied by 2 to make it equal to the IP formulation, see Section 4.3. The total objective function is then:

$$\sum_{p \in \mathcal{P}} (100 c_p target\_violation_p + 2 window\_diff_p), \tag{8}$$

where c_p are weights for each priority group used to capture that it is worse to violate the target for higher prioritized patients. The weight 100 reflects that waiting time targets are more important than minimizing window switches.

4.2 Packing-based CP Model

In the packing-based CP model, $fraction_{p,d} = 0$ if patient p has not started or has finished treatment on day d is expressed by the `regular` constraint [23], replacing (3). For example, if the treatment length is 10 days, the regular expression is $r = 0^* \cdot 1 \cdot 2 \cdot \ldots \cdot 8 \cdot 9 \cdot 10 \cdot 0^*$ (where $\cdot$ is concatenation) in:

$$\texttt{regular}([fraction_{p,d} | d \in \mathcal{D}], r) \quad \forall p \in \mathcal{P}. \tag{9}$$

The variables $start_{p,d}$ are removed as the constraints (1) and (4) are removed. Instead, regular expressions on the time windows define the treatment patterns, where window 0 corresponds to no treatment. The example regular expression $r = 0 \cdot ([1-4]^5 \cdot 0^2)^6$ states that $f = 0$ gives $window_{p,0} = 0$, then $window_{p,f} \in \{1, \ldots, 4\}$ for five days, followed by two pause days where $window_{p,f} = 0$, and then repeating this pattern six times, leading to:

$$\texttt{regular}([window_{p,f} | f \in \{0, \ldots, L_p\}], r) \quad \forall p \in \mathcal{P}. \tag{10}$$

The `bin_packing` constraint [29] ensures that the patients fit in each window:

$$\texttt{bin\_packing}([\infty, w_L, w_L, w_L, w_L], [\mathit{window}_{p,f} | p \in \mathcal{P}], [\mathit{dur}_p | p \in \mathcal{P}]) \quad \forall d \in \mathcal{D}, \tag{11}$$

where constraint (2) connects d to f. The capacity of window 0 is infinite and the other windows have the window length w_L as capacity. The partially occupied input schedule is also included as patients with fixed windows.

4.3 IP Model

Variables. The basic variables of the IP model are (again using $\mathbb{B} = \{0, 1\}$):

$s_{p,d} \in \mathbb{B}$ $s_{p,d} = 1$ iff patient $p \in \mathcal{P}$ starts treatment on day $d \in \mathcal{D}$
$q_{p,d,f} \in \mathbb{B}$ $q_{p,d,f} = 1$ iff patient $p \in \mathcal{P}$ has their f-th treatment day on $d \in \mathcal{D}$, $f \in \mathcal{F}$
$x_{p,d,w} \in \mathbb{B}$ $x_{p,d,w} = 1$ iff patient $p \in \mathcal{P}$ is scheduled in window $w \in \mathcal{W}$ on day $d \in \mathcal{D}$

There are also variables that are derived from the basic variables:

$u_{p,d} \in \mathbb{B}$ $u_{p,d} = 1$ iff patient $p \in \mathcal{P}$ has an active treatment day on $d \in \mathcal{D}$
$y_{p,d,w} \in \mathbb{B}$ help variable for the objective function where $y_{p,d,w} = 1$ if patient $p \in \mathcal{P}$ is scheduled in window $w \in \mathcal{W}$ on day $d \in \mathcal{D}$, and if patient p is not scheduled in window w on day d, $y_{p,d,w} \in \mathbb{B}$

Constraints. Patient $p \in \mathcal{P}$ will be treated for L_p days. Thus, the last day to start treatment is $\mathit{daylimit}_p = D - L_p + 1$. The treatment should start exactly one time on an allowed start day given by $\mathcal{A}_p$:

$$\sum_{d \subseteq \mathcal{A}_p}^{\mathit{daylimit}_p} s_{p,d} = 1, \quad \forall p \in \mathcal{P}.$$

For the CP models, constraint (5) states which fractions that can be delivered on which days. In the IP model, this is enforced by setting $q_{p,d,f} = 0$ for all $p \in \mathcal{P}, d \in \mathcal{D}, f \notin \mathcal{F}_{p,d}$ where $\mathcal{F}_{p,d} := \{\max(0, d - (D - L_p)), \ldots, \min(d, L_p)\}$. Using $\hat{\mathcal{F}}_{p,d}$ to denote $\mathcal{F}_{p,d}$ with the last element excluded, the constraint:

$$q_{p,d,f} = q_{p,d+1,f+1} \quad \forall p \in \mathcal{P}, d \in \{1, \ldots, D-1\}, f \in \hat{\mathcal{F}}_{p,d}$$

enforces that all treatment days are scheduled after each other.

The following constraints state that the f-th treatment day can only happen once and that patient p is scheduled at most once every day d:

$$\sum_{d \in \mathcal{D}} q_{p,d,f} \leq 1 \quad \forall p \in \mathcal{P}, f \in \mathcal{F}_{p,d} \qquad \sum_{f \in \mathcal{F}_{p,d}} q_{p,d,f} \leq 1 \qquad \forall p \in \mathcal{P}, d \in \mathcal{D}.$$

In the CP models, this is enforced by constraints (4) or (9).

The first treatment day $f = 1$ is given on the start day for each patient:

$$q_{p,d,1} = s_{p,d} \quad \forall p \in \mathcal{P}, d \in \mathcal{D}.$$

In the fractionation schedule for patient p, an active treatment day f gives $FS_{p,f} = 1$. A variable $u_{p,d}$ is introduced so that $u_{p,d} = 1$ iff patient p is during treatment on day d ($q_{p,d,f} = 1$) and has an active treatment day ($FS_{p,f} = 1$) and zero otherwise, thus, it controls if d is an active day or not for patient p:

$$u_{p,d} = \sum_{f \in \mathcal{F}_{p,d}} (q_{p,d,f} FS_{p,f}) \quad \forall p \in \mathcal{P}, d \in \mathcal{D}.$$

Each patient is scheduled in exactly one time window on active treatment days, and not in any window on off-days:

$$\sum_{w \in \mathcal{W}} x_{p,d,w} = u_{p,d} \quad \forall p \in \mathcal{P}, d \in \mathcal{D}.$$

In the CP models, this constraint is expressed by constraints (4) or (10).

In order to make sure that all treatments fit within each time window, $u_{p,d}$ is used to keep track of if the patient has an active day or not, together with $x_{p,d,w}$, which is one iff patient p is scheduled in window w on day d:

$$\sum_{p \in \mathcal{P}} x_{p,d,w} u_{p,d} dur_p + \sum_{t_s \subseteq w} S_{d,t_s} \leq w_L \quad \forall d \in \mathcal{D}, w \in \mathcal{W}.$$

S is the input schedule, where an element is one iff time slot t_s on day d is occupied. Thus, the sum of the duration of all patients in window w plus the previously occupied slots in that window must be less than or equal the window length w_L. In the CP models, this is enforced by constraints (1) or (11). A major difference is that the number of constraints in the IP model grows with the number of time windows, which is not the case in the CP models.

Objective Function. The objective is the same as in the CP model. A penalty is added for the time by which the target is missed:

$$f_{1,p} = \sum_{d=d_{L,p}}^{D} s_{p,d}(d - d_{L,p}) \quad \forall p \in \mathcal{P},$$

where $d_{L,p}$ corresponds to the waiting target in days for patient p and $s_{p,d} = 1$ on the start day. This constraint corresponds to the CP objective function (6).

The other objective is to schedule the patient in the same time window each day. To do this, a help variable $y_{p,d,w} \in \mathbb{B}$ is introduced so that $y_{p,d,w}$ is one when $x_{p,d,w}$ is one, and the sum of $y_{p,d,w}$'s is one on all days:

$$\begin{aligned} y_{p,d,w} &\geq x_{p,d,w} && \forall p \in \mathcal{P}, d \in \mathcal{D}, w \in \mathcal{W} \\ \sum_{w \in \mathcal{W}} y_{p,d,w} &= 1 && \forall p \in \mathcal{P}, d \in \mathcal{D}. \end{aligned}$$

As for the CP models, the problem with 3 machines with 4 windows each is modeled as having 1 machine with 12 windows, and hence there should be no penalty for switching from window 1 to 5 or 9. Introduce $\mathcal{W}_m = \{1, \ldots, W_m\}$ where W_m is the number of windows on each machine. Each window switch is penalized in the second objective function, where we here assume 3 machines:

$$f_{2,p} = \sum_{d \in \hat{\mathcal{D}}} \sum_{w \in \mathcal{W}_m} \left| \sum_{i=\{w, w+W_m, w+2W_m\}} (y_{p,d,i} - y_{p,d+1,i}) \right| \quad \forall p \in \mathcal{P}. \tag{12}$$

Since $f_{2,p}$ sums both $0-1$ and $1-0$ for each switch, there is a penalty of 2 for every switch. To make the CP objective function equivalent, the factor 2 difference is adjusted for in (8).

New variables avoid absolute values, as they would render the model non-linear: $z_{p,d,w} = \sum_{i=\{w,w+W_m,w+2W_m\}}(y_{p,d,i} - y_{p,d+1,i})$ for $w \in \mathcal{W}_m$. $z_{p,d,w}$ is divided into a positive and negative part; $z_{p,d,w} = z^+_{p,d,w} - z^-_{p,d,w}$:

$$\begin{aligned} f_{2,p} &= \sum_{d \in \hat{\mathcal{D}}} \sum_{w \in \mathcal{W}_m} z^+_{p,d,w} + z^-_{p,d,w} && \forall p \in \mathcal{P} \\ z^+_{p,d,w} &\geq 0, z^-_{p,d,w} \geq 0 && \forall p \in \mathcal{P}, d \in \mathcal{D}, w \in \mathcal{W}_m. \end{aligned} \tag{13}$$

The two formulations (12) and (13) are equivalent in a minimization setting.

In total, the objective function is equivalent to the CP objective function (8):

$$\sum_{p \in \mathcal{P}} (100 c_p f_{1,p} + f_{2,p}).$$

5 CP Search

When solving the CP models, the search is conducted in the following order:

1. Assign priority A patients randomly to a start day as early as possible.
2. Assign priority C and D patients to a start day randomly by choosing the $start\_day_p$ variable with smallest domain size over weighted degree [4].
3. Assign priority B patients to their earliest possible start day.
4. Assign the number of window switches $window\_diff_p$ as small as possible for all patients sorted by their duration, with longest duration assigned first.

For easy instances, with few patients to schedule, it is possible to construct deterministic search heuristics for the CP models that perform much better than the random search strategies described above. However, for more difficult instances these heuristics fail to even find a solution within reasonable time. This is the reason for including randomization.

When solving a minimization problem in CP, branch-and-bound tree search is used, which follows the same branch until it has failed. However, using a restart strategy causes search to restart from the top node whenever search finds a solution or after a specific number of failures defined by the restart strategy. In this case, using a restart strategy yields better results because the problem is somewhat under-constrained; it is easy to find feasible solutions and relatively few failures occur. Therefore, the Luby restart strategy [20] multiplied by a factor of 100 is used. The interval is a result of testing many different intervals on different problem setups. Doing restarts this often can be compared to approximating a Large Neighborhood Search (LNS) [28] (see also Section 7).

6 Experiments and Results

Multiple experiments are run to evaluate and compare the three models. The experiments are run on a Windows 10 machine with an Intel® Core™ i9-7940X X-series processor and 64 GB of RAM. The IP model is solved using the MIP solver of CPLEX 12.8 in the Python API with default parameters. The CP models use MiniZinc 2.2.2 and are solved with Gecode 6.1.0. Other solvers have been tested, such as the lazy clause solver Chuffed, but Gecode gave the best overall results on the tested problem instances.

A simulation engine is built with Python 3.6. In this engine, the first day starts from an empty schedule and patients to be scheduled are assumed to arrive according to a Poisson process. For each simulated day, a patient schedule is created using the previously described models, and the patients are fixed to the schedule if they have a start day within a week (since this is the limit assumed to communicate the schedules to the patients). The schedule from the previous day is used as input for the next day in the simulation, together with the backlog of yet unscheduled patients. For these patients, the waiting time target is adjusted by one day since it is counted from the day the patient is ready for treatment.

The simulation engine is used to generate problem benchmarks, that each have a partially occupied schedule, a patient backlog, and the number of expected patients arriving per day (as discussed in Section 3). Occupation in a schedule is measured as the average occupation of the first two weeks in that schedule. This is not a perfect measure; if the first week is completely booked and the second is completely free, some urgent patients will not be able to meet their target dates, although occupancy is 50% in total for these weeks.

Scheduling patients on three different machines, 16 different benchmarks that are grouped into three categories are summarized in Table 1. The categories capture the following aspects:

a) The average number of patients arriving each day.
w) Which weekday the schedule is created.
l) The load in the input, i.e., the amount of partial occupation in the input schedule and the size of the backlog. These are closely related, since an almost empty schedule does not come with a large backlog, and vice versa. Benchmarks l-3 and l-4 examine scalability and do not represent realistic scenarios, as patients would be transferred to other clinics.

The weights in the objective function are chosen as $c_1 = 10$, $c_2 = 3$, $c_3 = 1$, corresponding to priority group A to C, and are the same as in [24]. Priority D is for expected future patients of priority A and has weight $c_4 = 5$. The day is divided into 4 or 6 time windows. The timeout is set to 6 hours.

Table 1. Setup of the benchmarks, 3 machines

Benchmark	Occupation (%) (4/6 windows)	Expected number of arriving patients	Number of patients in backlog (4/6 windows)	Number of patients including future arrivals (4/6 windows)	Planning day
1. a-1	18.5 / 10.2	4	5 / 8	49 / 52	Sunday
2. a-2	18.5 / 10.2	6	5 / 8	77 / 80	Sunday
3. a-3	18.5 / 10.2	8	5 / 8	93 / 96	Sunday
4. a-4	53.9 / 55.0	4	8 / 8	52 / 52	Tuesday
5. a-5	53.9 / 55.0	6	8 / 8	80 / 80	Tuesday
6. a-6	53.9 / 55.0	8	8 / 8	96 / 96	Tuesday
7. w-1	53.9 / 55.0	5	8 / 8	79 / 79	Monday
8. w-2	53.9 / 55.0	5	8 / 8	79 / 79	Tuesday
9. w-3	53.9 / 55.0	5	8 / 8	79 / 79	Wednesday
10. w-4	53.9 / 55.0	5	8 / 8	79 / 79	Thursday
11. w-5	53.9 / 55.0	5	8 / 8	79 / 79	Friday
12. w-6	53.9 / 55.0	5	8 / 8	79 / 79	Saturday
13. l-1	53.9 / 55.0	5	8 / 8	79 / 79	Sunday
14. l-2	60.9 / 61.8	5	16 / 14	87 / 85	Sunday
15. l-3	69.8 / 64.8	5	24 / 34	90 / 100	Friday
16. l-4	73.3 / 69.5	5	45 / 48	111 / 114	Thursday

The performance of the models is measured as the objective function value as a function of runtime for the benchmarks, for both 4 and 6 windows.

Average Patient Arrival. The results for benchmarks a-1 to a-6 are shown in Figure 1. They show that except for the 6 window case in benchmark a-4, the IP model reaches optimality considerably faster, and the packing-based CP model outperforms the scheduling-based CP model on all instances. When the arrival rate is low, as in benchmarks a-1 and a-4, the IP and CP models have similar performance. The CP models are however more sensitive to an increase in the average number of patients arriving each day, while the time to reach optimality in the 4 window IP model does not change significantly. Increasing the number of time windows from 4 to 6 makes the IP model slower in all cases. When the partial occupation is low (benchmarks a-1 to a-3), the packing-based CP model is not slower with more windows, which is not the case when the partial occupation is higher (benchmarks a-4 to a-6).

Weekday. The results when varying the weekday the schedule is created are presented in Figure 2. Some treatment protocols state that treatment can be initiated only on certain days, but this does not have a large effect on runtime. An observation is that for the 6-window case, although the IP model reaches optimality faster, the packing-based CP model has a better objective value initially. Benchmarks w-1 to w-6 show that if the time limit is short, the packing-based CP model performs better than the IP model (the same holds for a benchmark w-7 starting on a Sunday which is not included due to space limitations).

Patient Load. Altering the load, the results can be seen in Figure 3. Again, the IP model outperforms the CP models in finding optimality. In benchmarks l-3 and l-4, in 6 hours of runtime the scheduling-based CP model does not find any

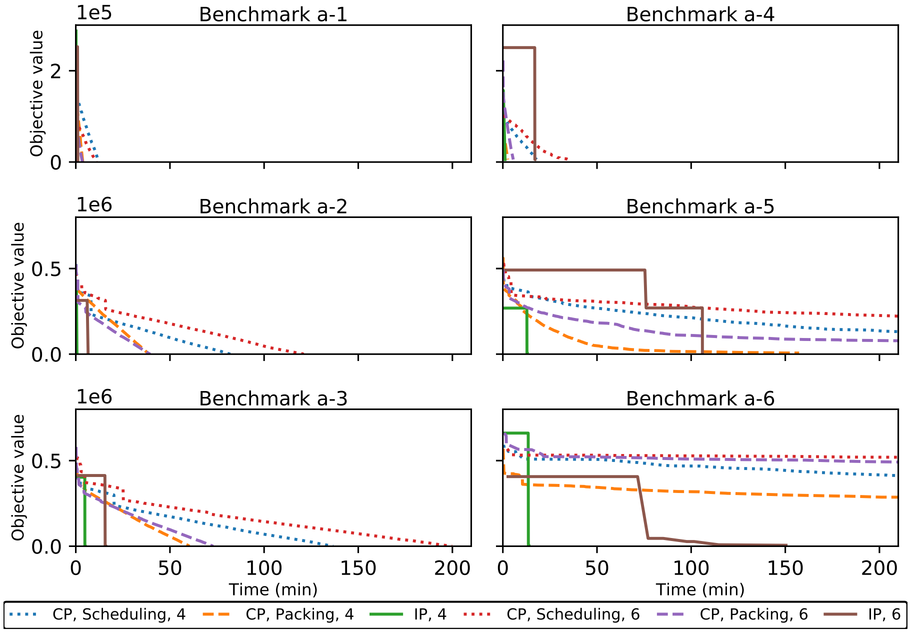

Fig. 1. Results with varying arrival rate, 4 windows and 6 windows

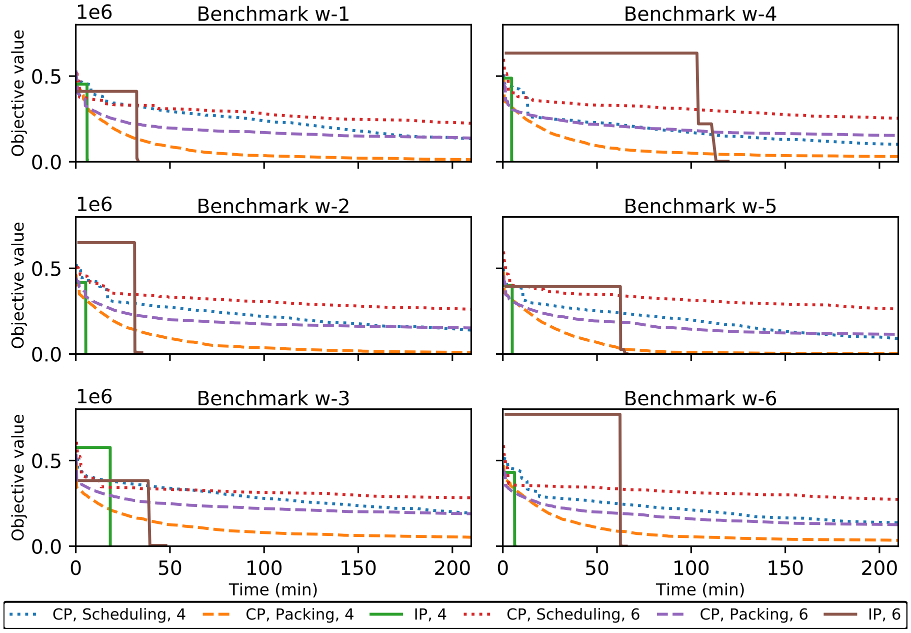

Fig. 2. Results with varying planning day, 4 windows and 6 windows

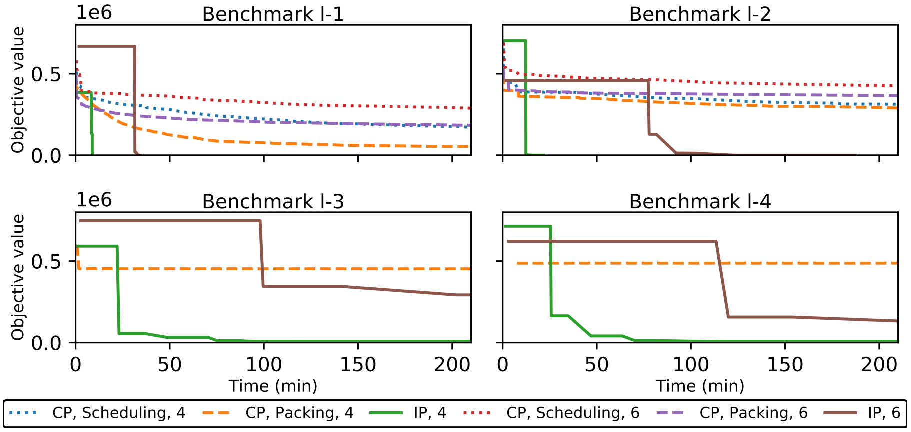

Fig. 3. Results when varying the load, 4 windows and 6 windows

solutions, the packing-based CP model does not find a solution in the 6-window case, and the IP model does not reach optimality in the 6-window case. However, these cases represent too heavy a load to be realistic scenarios.

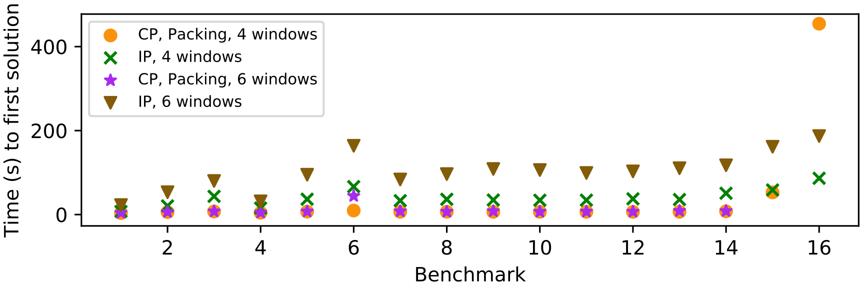

Fig. 4. Time to find the first feasible solution for each of the benchmarks

Time to Feasible Solution. Figures 1 to 3 show that the IP model outperforms the CP models in finding an optimal solution. However, the packing-based CP model is in almost all cases faster to find a feasible solution, see Figure 4.

7 Conclusions and Future Work

Radiation therapy is one of the most commonly used cancer therapies worldwide. The waiting times can be reduced by the optimization of the use of LINACs for treatment.

This paper introduces three different models that capture the non-block radiotherapy patient scheduling problem; one IP model, a scheduling-based CP model, and a packing-based CP model. The patients have different priority levels, treatment patterns, treatment durations, and start days for treatment. The expected future patient arrivals are included in the models to predict future resource utilization.

The models are evaluated for multiple different scenarios. The results show that the packing-based CP model outperforms the scheduling-based CP model on all problem instances. In general, the CP models find feasible solutions faster than the IP model, however, the IP model reaches optimality considerably faster. All models are sensitive to an increase in the number of time windows per day.

Future Work. To make the models more realistic, overtime on the machines and cancellation of treatments should be included in the models. Cancellation of treatments is common in RT, and allowing for overtime on the machines is important since it is often used in practice to reduce waiting times. Both aspects are bound to increase the complexity of the models, and overtime is likely to make the IP model nonlinear. Another future direction is to extend the models to include multiple machine types, which also increases the models' complexity.

For the models to be able to handle these complexities, a potential future extension is to take advantage of the strengths of each model in an IP/CP hybrid. Another option would be to use some decomposition method on the IP model, for example Benders decomposition. To better explore the search space, an option is to use Large Neighborhood Search (LNS) when solving the CP models.

A potential improvement of the stochastic aspect of the models is to use scenario-based probabilities instead of expected values when accounting for patients arriving in the future.

The models have so far been compared with each other. The next step would be to see how the models perform over time, using the simulation engine and comparing to both a myopic scheduling strategy (not taking future patient arrivals into account) and to historical data from a clinic.

Acknowledgments. The authors thank Per Enqvist at KTH and Kjell Eriksson at RaySearch Laboratories for a fruitful collaboration. The authors are grateful for insightful discussions about the CP models with Mats Carlsson and Peter J. Stuckey and constructive comments from the anonymous reviewers.

References

1. Aggoun, A., Beldiceanu, N.: Extending Chip in order to solve complex scheduling and placement problems. Mathematical and Computer Modelling **17**(7), 57 – 73 (1993)
2. Baatar, D., Boland, N., Brand, S., Stuckey, P.J.: CP and IP approaches to cancer radiotherapy delivery optimization. Constraints **16**, 173–194 (2011)

3. Barták, R., Salido, M., Rossi, F.: New trends in constraint satisfaction, planning, and scheduling: A survey. The Knowledge Engineering Review **25**, 249–279 (Sep 2010)
4. Boussemart, F., Hemery, F., Lecoutre, C., Sais, L.: Boosting systematic search by weighting constraints. In: de Mántaras, R.L., Saitta, L. (eds.) Sixteenth European Conference on Artificial Intelligence. pp. 146–150. IOS Press, Valencia, Spain (Aug 2004)
5. Burke, E.K., De Causmaecker, P., Berghe, G.V., Van Landeghem, H.: The State of the Art of Nurse Rostering. Journal of Scheduling **7**(6), 441–499 (Nov 2004)
6. Burke, E.K., Leite-Rocha, P., Petrovic, S.: An integer linear programming model for the radiotherapy treatment scheduling problem. arXiv e-prints arXiv:1103.3391 (2011)
7. Cayirli, T., Veral, E.: Outpatient Scheduling in Health Care: a Review of Literature. Production and Operations Management **12**(4), 519–549 (2009)
8. Chen, Z., King, W., Pearcey, R., Kerba, M., Mackillop, W.J.: The relationship between waiting time for radiotherapy and clinical outcomes: A systematic review of the literature. Radiotherapy and Oncology **87**(1), 3 – 16 (2008)
9. Conforti, D., Guerriero, F., Guido, R.: Optimization models for radiotherapy patient scheduling. 4OR **6**(3), 263–278 (Sep 2008)
10. Conforti, D., Guerriero, F., Guido, R.: Non-block scheduling with priority for radiotherapy treatments. European Journal of Operational Research **201**(1), 289–296 (2010)
11. Ehrgott, M., Holder, A.: Operations Research Methods for Optimization in Radiation Oncology. Journal of Radiation Oncology Informatics **6**(1), 1–41 (2014)
12. Fortin, A., Bairati, I., Albert, M., Moore, L., Allard, J., Couture, C.: Effect of treatment delay on outcome of patients with early-stage head-and-neck carcinoma receiving radical radiotherapy. International Journal of Radiation Oncology Biology Physics **52**(4), 929–936 (2002)
13. Gocgun, Y.: Simulation-based approximate policy iteration for dynamic patient scheduling for radiation therapy. Health Care Management Science **21**(3), 317–325 (2018)
14. Gomez, D.R., Liao, K.P., Swisher, S.G., Blumenschein, G.R., Erasmus, J.J., Buchholz, T.A., Giordano, S.H., Smith, B.D.: Time to treatment as a quality metric in lung cancer: Staging studies, time to treatment, and patient survival. Radiotherapy and Oncology **115**(2), 257–263 (2015)
15. Hahn-Goldberg, S., Beck, J.C., Carter, M.W., Trudeau, M., Sousa, P., Beattie, K.: Solving the chemotherapy outpatient scheduling problem with constraint programming. Journal of Applied Operational Research **6**(3), 135–144 (2014)
16. Halperin, E.C., Wazer, D.E., Brady, L.W., Perez, C.A.: Perez and Brady's principles and practice of radiation oncology. Lippincott Williams and Wilkins, sixth edn. (2013)
17. Jacquemin, Y., Marcon, E., Pommier, P.: A pattern-based approach of radiotherapy scheduling. In: IFAC Proceedings Volumes. vol. 44, pp. 6945–6950 (Jan 2011)
18. Kapamara, T., Sheibani, K., OCL, H., Reeves, C., Petrovic, D.: A review of scheduling problems in radiotherapy. In: Proceedings of the International Control Systems Engineering Conference on Systems Engineering (ICSE2006). pp. 201–207 (2006)
19. Legrain, A., Fortin, M.A., Lahrichi, N., Rousseau, L.M.: Online stochastic optimization of radiotherapy patient scheduling. Health Care Management Sciences **18**, 110–123 (2015)
20. Luby, M., Sinclair, A., Zuckerman, D.: Optimal speedup of Las Vegas algorithms. Information Processing Letters **47**, 173–180 (1993)

21. May, J.H., Spangler, W.E., Strum, D.P., Vargas, L.G.: The Surgical Scheduling Problem: Current Research and Future Opportunities. Production and Operations Management **20**, 392–405 (2011)
22. O'Rourke, N., Edwards, R.: Lung cancer treatment waiting times and tumour growth. Clinical Oncology **12**(3), 141–144 (2000)
23. Pesant, G.: A regular language membership constraint for finite sequences of variables. In: Wallace [32], pp. 482–495
24. Petrovic, D., Castro, E., Petrovic, S., Kapamara, T.: Radiotherapy scheduling. In: Uyar, A., Ozcan, E., Urquhart, N. (eds.) Automated Scheduling and Planning, Studies in Computational Intelligence, vol. 505, pp. 155–189. Springer, Berlin, Heidelberg (2013)
25. Petrovic, S., Leung, W., Song, X., Sundar, S.: Algorithms for radiotherapy treatment booking. In: Qu, R. (ed.) 25th Workshop of the UK Planning and Scheduling Special Interest Group. pp. 105–112 (Apr 2006)
26. Riff, M.C., Cares, J.P., Neveu, B.: RASON: A new approach to the scheduling radiotherapy problem that considers the current waiting times. Expert Systems with Applications **64**, 287–295 (Dec 2016)
27. Sauré, A., Patrick, J., Tyldesley, S., Puterman, M.L.: Dynamic multi-appointment patient scheduling for radiation therapy. European Journal of Operational Research **223**(2), 573 – 584 (2012)
28. Shaw, P.: Using constraint programming and local search methods to solve vehicle routing problems. In: Maher, M., Puget, J.F. (eds.) Fourth International Conference on Principles and Practice of Constraint Programming. LNCS, vol. 1520, pp. 417–431. Springer Berlin Heidelberg, Berlin, Heidelberg (1998)
29. Shaw, P.: A constraint for Bin Packing. In: Wallace [32], pp. 648–662
30. Van Harten, M.C., Hoebers, F.J., Kross, K.W., Van Werkhoven, E.D., Van Den Brekel, M.W., Van Dijk, B.A.: Determinants of treatment waiting times for head and neck cancer in the netherlands and their relation to survival. Oral Oncology **51**(3), 272–278 (2015)
31. Vieira, B., Hans, E.W., Van Vliet-Vroegindeweij, C., Van De Kamer, J., Van Harten, W.: Operations research for resource planning and -use in radiotherapy: a literature review. BMC Medical Informatics and Decision Making **16**(149) (2016)
32. Wallace, M. (ed.): Tenth International Conference on Principles and Practice of Constraint Programming, LNCS, vol. 3258. Springer Berlin Heidelberg, Toronto, Canada (Sep 2004)