# Using Branch-and-Price to Determine Optimal Treatment Plans for Volumetric Modulated Arc Therapy (VMAT)

Pınar Dursun[a], Z. Caner Taşkın[a,*], İ. Kuban Altınel[a]

[a]*Department of Industrial Engineering, Boğaziçi University, 34342, Bebek, İstanbul, Turkey*

**Abstract**

Volumetric Modulated Arc Therapy (VMAT) is the state-of-the-art technique for external radiation therapy treatment. In this method, radiation can be delivered continuously on one or more arcs during the rotation of the gantry of the linear accelerator. This property makes VMAT powerful in obtaining high conformal plans in terms of dose distribution within short treatment times. However, the apertures composed by the leaves of the multileaf collimator (MLC) system that shapes continuously the radiation are interdependent, which makes treatment planning hard. We propose a mixed integer linear programming model for VMAT planning problem and exact branch-and-price algorithms to solve it. The objective of the model is to minimize total radiation that is delivered to the patient, and pricing subproblem is decomposable by rows of the MLC and can be solved as a shortest path problem. We generate a large set of test instances from a real data set and evaluate the performance of the proposed branch-and-price algorithm. Computational results reveal that new algorithms are efficient and capable of finding optimal solutions for large problem instances.

*Keywords:* `integer programming, radiation therapy, VMAT, column generation, branch-and-price; shortest path problem`

## 1. Introduction

External-beam radiation is one of the most common types of radiation therapy, and it is used for treating many types of cancer either alone or together with surgery, chemotherapy, or both. A *linear accelerator*, which is generally equipped with a system called *multileaf collimator* (MLC), generates and directs high-energy beams of radiation from outside the body. An MLC system consists of a number of parallel movable metal leaf pairs that can block some fraction of the radiation beam. The resulting shape, where the beam passes through, is called an *aperture* (see Figure 1). The leaf structures of the MLC systems imply that there can be at most one opening between a pair of leaves. This property is called the *consecutive ones property*. In addition, some MLC systems may not allow *interdigitation*, which means that the overlap of the left and right leaves of adjacent beamlet rows is forbidden. We consider MLC

*Corresponding author. Phone: + 90 (212) 359 6407, Fax: + 90 (212) 265 1800

*Email addresses:* `pinar.dursun@boun.edu.tr` (Pınar Dursun), `caner.taskin@boun.edu.tr` (Z. Caner Taşkın), `altinel@boun.edu.tr` (İ. Kuban Altınel)

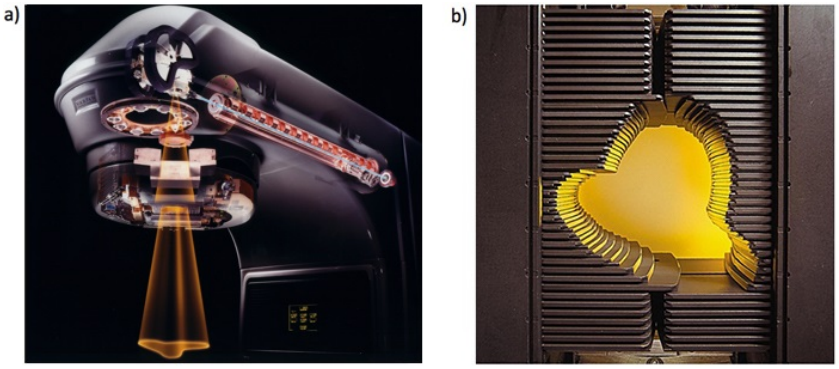


Figure 1: (a) A linear accelerator (Varian, 2018) (b) A multileaf collimator system (CCS Healthcare, 2018).

systems having the consecutive ones property and refer the reader to the study of Gören and Taşkın (2015) for detailed information about other mechanical characteristics that MLC systems can have.

Intensity Modulated Radiation Therapy (IMRT) and Volumetric Modulated Arc Therapy (VMAT) are two commonly applied forms of external-beam radiation therapy. In both methods the linear accelerator rotates around the body and delivers radiation from different angles by keeping the cancer volume on the target. In this way, the conformity of dose distribution to the planning target volume (tumor plus some margin) and normal tissue sparing are much superior compared to earlier techniques (Cambazard et al., 2012). Although IMRT has been used extensively in radiation therapy since 1990s (Ehrgott et al., 2010), VMAT is a more advanced and recent technology. In VMAT high conformal dose distributions may be obtained by delivering less radiation (Teoh et al., 2011), and the delivery time of the treatment is significantly shorter since radiation is delivered continuously through one or more arcs during the rotation of the linear accelerator's gantry (Otto, 2008; Peng et al., 2012). In IMRT, typically only a few discrete angles (5-9) are used (Romeijn et al., 2006). Furthermore, the linear accelerator stops delivering radiation while moving its gantry between different beam angles in both dynamic (*sliding window technique*) and static (*step-and-shoot technique*) types of IMRT (Ehrgott et al., 2010). There are several advantages of having short therapy durations. First of all, the discomfort of patients and risk of negative effects that may result from patient movements decrease. Also, resource utilization becomes more efficient and it is possible to treat more patients (Peng et al., 2012). Moreover, both VMAT and IMRT methods require higher amount of radiation (i.e. more monitor units (MUs)) compared to previous techniques (Palma et al., 2008; Teoh et al., 2011). It is known that high MUs increase the risk of the secondary malignancy since it increases the integral body dose (Hall and Wuu, 2003).

The main contribution of this study is methodological. In particular, we develop efficient branch-and-price (BP) algorithms for the solution of the mixed integer linear programming (MILP) model, which we call VMATP. This model is initially proposed in Dursun et al. (2016). VMATP aims to minimize the total radiation amount given to the patient during the treatment while satisfying prescription doses and mechanical constraints. In our recent work (Dursun et al., 2019) we propose a Benders decomposition algorithm to improve its solvability. Column generation

based heuristics have been applied to the solution of the VMAT planning problem (Men et al., 2010; Peng et al., 2012; Mahnam et al., 2017). However, to the best of our knowledge, there is no algorithm in the literature that combines column generation with branch-and-bound to yield a BP scheme for the exact solution of the VMAT treatment planning problem.

The problem has a natural tendency to decompose into two interacting parts. One of them deals with the geometry of the equipment while the other determines the right amount of radiation dose of the treatment region. Based on this observation we have proposed an exact solution algorithm based on Benders decomposition (Dursun et al., 2019). The idea is to keep binary variables in the master problem and solve a linear programming subproblem to generate cuts. We have recently realized that reversing the order of decomposition and considering a subproblem including the binary variables may have been more advantageous since the problem itself can be decomposed into shortest path subproblems. In fact this gives birth to BP algorithms explained in the sequel. As can be observed from the computational results one of them performs significantly better than the Benders decomposition algorithm, and can compute optimal treatment plans minimizing total radiation for considerably larger instances.

The rest of the paper is organized as follows. Section 2 reviews existing related works. Section 3 defines the VMAT treatment planning problem and briefly introduces the mathematical formulation that is solved by the BP algorithms. Section 4 presents the BP algorithms and necessary derivations for the column generation method. We also discuss implementation issues such as the pricing subproblem solution, branching strategy, and initialization of the restricted model in this section. Section 5 describes the computational experiments and Section 6 concludes the paper.

## 2. Related Works

In this section, we first give a brief overview of the literature on IMRT and VMAT planning problems. Then, we present studies on column generation algorithms applied to IMRT and VMAT planning in detail.

### *2.1. IMRT and VMAT Planning Studies*

There are three main phases in IMRT planning, which can be solved either sequentially or in combination of two phases. The first phase deals with the *beam angle optimization* and the number and orientation of beam angles for irradiation are determined, which is mostly done by a medical physicist in practice based on experience. After determining the beam angles, a *fluence map* is obtained for each one of these angles in the second phase, which is called the *fluence map optimization* (FMO) *problem* (or *intensity problem*). A fluence map denotes the radiation intensity profile to be delivered through a given beam angle and can be represented by a two-dimensional nonnegative integer matrix. It is possible to formulate the FMO as a convex optimization problem; hence, it can be solved efficiently using one of the existing algorithms for convex optimization (Papp and Unkelbach, 2014). The third phase in IMRT planning is MLC *leaf sequencing* (MLS) (or *realization problem*). A given fluence map is decomposed into a number

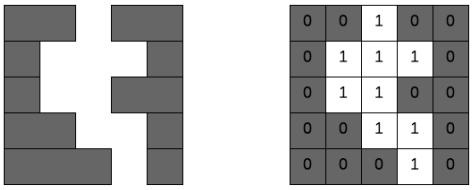


Figure 2: An aperture and its binary matrix representation

of disjoint apertures and corresponding radiation intensities (Guta, 2003; Taşkın et al., 2010; Cambazard et al., 2012). Note that an aperture can be represented by a two-dimensional binary matrix. In Figure 2 an aperture and its binary matrix representation are illustrated for an MLC system that has five leaf pairs (rows), and the leaf openings are decomposed into five columns. Each entry of the matrix is called a *beamlet* and takes value 1 if it is exposed. In IMRT treatment planning these three phases are often solved independently and sequentially. Furthermore, it is also possible to deal with two consecutive phases simultaneously. For instance, the first two phases can be considered together to determine a beam angle and fluence map simultaneously (Lee et al., 2003; Bertsimas et al., 2013). There are also studies that directly optimize a number of apertures with intensities for each one of the determined beam angles. In other words, they solve the second and third phases simultaneously instead of finding a fluence map first and then decomposing it into a number of deliverable apertures (Shepard et al., 2002). This problem is called *direct aperture optimization* (DAO). For more details on IMRT planning, we refer to the comprehensive survey of Ehrgott et al. (2010).

Rotational arc therapy was first introduced in 1995, and named Intensity Modulated Arc Therapy (IMAT) (Yu, 1995). Its use was not widespread until the work of Otto (2008), which introduced a more flexible technique called VMAT with additional degrees of freedom (i.e. gantry speed and dose rate). In VMAT the gantry of the linear accelerator can deliver radiation continuously to the patient's body, and thus in treatment planning it is assumed that there is a large number of equally spaced beam angles (*control points*) to discretize this continuous radiation delivery. At each control point there is only one aperture; however, the apertures at two adjacent control points are interconnected. This is because there are limitations on the motion of the MLC leaves during rotation. Thus, the VMAT planning problem cannot be decomposed into a number of subproblems that can be solved independently. As a result, designing a VMAT treatment plan is significantly harder compared to IMRT treatment planning. Even when the total time to complete a tour is fixed, the resulting problem is a large-scale nonconvex optimization problem. These characteristics make VMAT planning a challenging task, which requires much longer solution time than IMRT planning (Craft et al., 2012).

Studies on VMAT treatment planning can be classified into two groups. The ones in the first group determine an optimal IMRT plan consisting of a number of fluence maps, then convert these fluence maps to a VMAT plan using

an arc-sequencing method. In some of these two-step approaches, the fluence maps are calculated for a small number of evenly spaced control points (Cameron, 2005; Wang et al., 2008; Luan et al., 2008; Shepard et al., 2007; Cao et al., 2009; Bzdusek et al., 2009). Note that the resulting VMAT plan, after conversion, must satisfy the constraints of the MLC systems, hence there may be deviation in dose distribution quality. On the other hand, Craft et al. (2012); Salari et al. (2012); Wala et al. (2012) first obtain an "ideal" IMRT plan that includes a large number of control points, then they coarsen the IMRT plan to reduce delivery time by maintaining dose distribution quality. The studies placed in the second group directly optimize leaf positions and weights of the apertures and are called DAO methods similar to IMRT planning. The solution methods proposed in the literature are generally heuristic algorithms. Earl et al. (2003), Otto (2008), and Zhang et al. (2010) start with a relatively coarse sampling of the control points and use heuristic methods to find the final VMAT treatment plan. Papp and Unkelbach (2014) enforce unidirectional leaf motion over an arc segment, and determine the apertures by solving a sequence of convex optimization problems. Peng et al. (2015) propose a heuristic approach to solve VMAT with constant gantry speed and dose rate. Gozbasi (2010); Akartunalı et al. (2015), and Song et al. (2015) formulate MILP models for the VMAT planning problem in which an aperture and radiation intensity are optimized at each control point subject to a part of the clinical requirements. Gozbasi (2010) and Song et al. (2015) relax some treatment related constraints and aim to satisfy them via the objective function (i.e. minimizing total deviation from the prescribed doses or minimizing the weighted sum of the average dose on critical structures, etc.). On the other hand, in Akartunalı et al. (2015) most of the treatment related requirements are embedded into their MILP model as hard constraints. They only relax the partial volume constraints of target volumes (TVs) and the objective of their model is to maximize the total number of voxels that absorbs the prescribed amount of radiation. However, they all suggest heuristics to solve their problems in order to obtain clinically acceptable feasible treatment plans. Finally, in an earlier work (Dursun et al., 2016) we propose two MILP formulations for VMAT planning. Then we extend one of these formulations in Dursun et al. (2019) and apply Benders decomposition method to solve it efficiently.

### *2.2. Column Generation Based Studies on IMRT and VMAT Planning*

In this section we give more details of the studies on IMRT and VMAT planning that use column generation in their solution methodology. Boland et al. (2004) apply column generation method to find exact optimal solutions for the MLS problem, which is the last phase of IMRT planning. They decompose a given fluence map into a number of weighted apertures by minimizing beam-on time (the time for which the linear accelerator delivers radiation). In each iteration they solve a subproblem and generate a feasible promising aperture with negative reduced cost and then add it to the model. The subproblem considers consecutive ones property and interdigitation constraints. The authors design a network model for the subproblem where there are as many layers as the number of rows (the MLC leaf pairs). In each row there is a finite number of potential positions for the left and the right leaves that satisfy consecutive ones property. They call each of these potential position combinations as a *leaf configuration*, which we use also in the sequel. There are also studies that apply column generation for the solution of DAO in IMRT planning,

where the FMO and MLS are solved together (Romeijn et al., 2005; Men et al., 2007; Carlsson, 2008; Salari and Unkelbach, 2013; Preciado-Walters et al., 2006). The general framework of the algorithms proposed in these studies is to start with an empty set of apertures and add apertures to the plan iteratively. The subproblem yields the most promising feasible aperture. The master problem determines optimum weights of the apertures generated and added to the treatment plan so far. In particular, Romeijn et al. (2005) formulate a large-scale convex programming problem for DAO. They apply column generation method to solve their problem exactly and generate one or more promising apertures in each iteration by solving the network flow formulation similar to the study of Boland et al. (2004). They also modify the network in order to make it possible to solve the DAO problem where the MLC system requires connected apertures. Men et al. (2007) consider the MLC systems that allow only rectangular apertures, and solve the pricing subproblem by a polynomial time algorithm similar to the one in Romeijn et al. (2005).

Some of the studies in VMAT planning literature that directly optimize leaf positions and weights at all control points also use column generation method in their heuristic algorithms. Men et al. (2010) formulate a large-scale convex programming model in which the cost function consists of quadratic one-sided voxel-based penalties and a penalty-based soft constraint for the maximum dose rate variation limitation in VMAT delivery. They generate promising apertures by solving a polynomial-time algorithm and add to the master problem that determines dose rate of all currently generated apertures. They start with an empty set of apertures and in each iteration they generate one aperture for an unoccupied control point that is compatible with the previously generated ones with respect to the maximum leaf motion speed. Also, they solve the master problem by the gradient projection method (Rosen, 1960). Peng et al. (2012) improve this solution approach in their new column generation based greedy heuristic that also takes into account dose rate and gantry speed limit. In a recent study Mahnam et al. (2017) develop a large-scale nonlinear integer programming model that has a quadratic voxel-based least square penalty function as objective function similar to the one in Peng et al. (2012). They also propose a column generation based heuristic that generates a set of sequential apertures as a new column by solving the subproblem and adds to the restricted master problem. They use a graph theoretic approach and formulate the pricing subproblems as shortest path problems. They assume that the MLC system has only consecutive ones property and the apertures that form a partial arc can be decomposed into rows and can be handled independently. Namely, they find as many partial row arcs as the number of rows in the MLC system; then their union yields the aperture set in the partial arc. The proposed algorithms in these studies use column generation method to take advantage of finding the promising columns in a short time within a heuristic framework. However, to the best of our knowledge, our work is the first paper in the literature that proposes a BP algorithm for the exact solution of the VMAT planning problem.

## 3. Problem Formulation

### *3.1. Problem Definition*

For the sake of completeness, we start by briefly explaining VMAT planning problem and providing the related MILP model VMATP, which is introduced in our previous studies (Dursun et al., 2016, 2019), with minor revisions. In VMAT, the gantry of the linear accelerator delivers radiation continuously while rotating around the patient. Also, during rotation, the MLC leaves shape the opening where the radiation passes through for being able to focus on the tumor. However, all tissues exposed to radiation are affected. A VMAT treatment plan must satisfy all prescription radiation doses ensuring that the tumor is killed, healthy tissues are spared and organs at risks (OARs) can repair themselves. The first type of requirements is related to these prescription radiation doses. The second type of requirements that a VMAT plan must also satisfy is related to the mechanical limitations of the linear accelerator, as maximum leaf motion limits and dose rate limits. First, to be able to develop a MILP model for the problem we need to discretize the continuous parts of the problem such as radiation delivery, aperture shape, and body of the patient. It is generally assumed that there is a large number of evenly spaced control points on a co-planar arc. At each one of these control points the leaves of the MLC system form an aperture and radiation is delivered to the patient body only at these control points. This is a reasonable assumption since the effects of the radiation in two consecutive control points are expected to be almost identical, also the aperture shapes are similar due to the limitation on leaf movement.

There is an upper limit on the dose provided by the linear accelerator. We assume that the speed of the gantry is constant and it completes a tour within a certain time. We also assume that the radiation is delivered for the same amount of time at each control point. Thus, it is possible to determine maximum radiation intensity at a control point using dose rate and the tour completion time. Other mechanical limitations are related to the particular properties of the MLC system. We only consider the MLC systems that have consecutive ones property. During the rotation of the gantry the MLC leaf pairs also move and change the shape of the beam. However, there is a limitation in the movement of the leaves in a time unit, thus the treatment plan must also satisfy this maximum leaf motion constraint.

There are typically two types of prescription doses for a structure: in the first type the whole structure must absorb at most (or at least) a specified amount of radiation. These prescriptions are ensured by full volume constraints in the model. In the other type, a partial volume of a structure must absorb at least (or at most) prescribed amount (or tolerance amount) of radiation. For instance, at least 98% of the tumor must absorb at least the prescribed amount of radiation. These requirements are satisfied by partial-volume constraints. Finally, to be able to introduce the dose prescriptions to the model, the body of the patient is discretized into a large number of small cubes called *voxels*. The amount of radiation absorbed by each voxel is calculated and the volume constraints are forced to be satisfied using these amounts. As a result, the MILP model VMATP determines an optimal treatment plan, which consists of the aperture shape and the amount of radiation to be delivered to the patient at each control point, that satisfies these limitations with minimum total radiation intensity. Namely, we minimize the sum of the radiation intensities at control points. Figure 3 illustrates a treatment arc consisting of only three equally spaced control points.

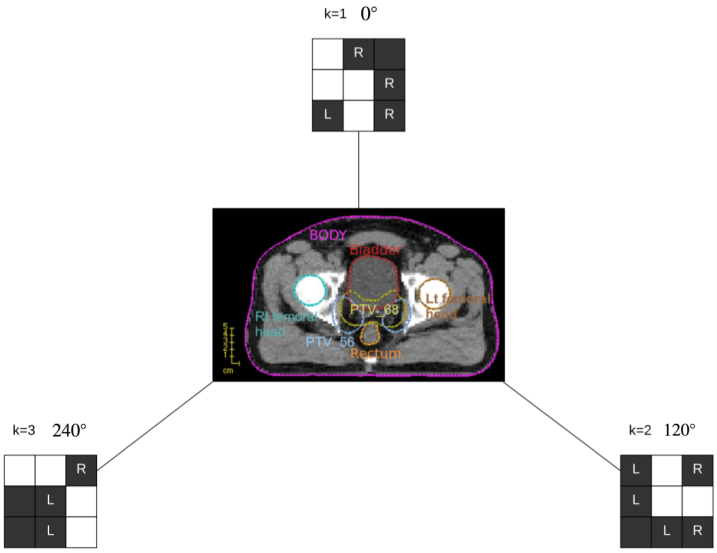


Figure 3: A treatment arc consisting of 3 control points, 3 rows and 3 columns

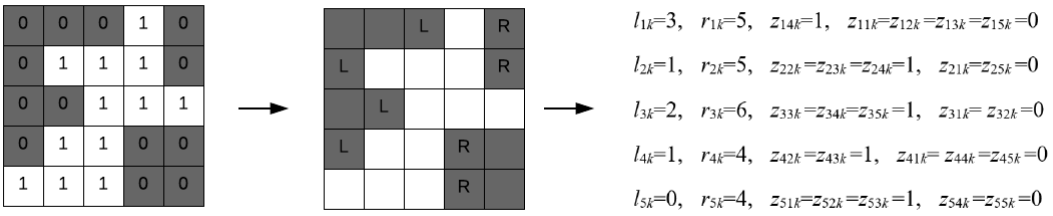


Figure 4: An aperture and its decision variables

### *3.2. Mixed Integer Linear Programming Model*

Let a two-dimensional $m \times n$ matrix represent an aperture at a control point. The number of MLC leaf pairs, and thus the number of rows is $m$ and the number of columns is $n$. We introduce a number of nonnegative integer variables and binary variables to form each one of these matrices. For each row $i$ at control point $k$ two nonnegative integer variables $l_{ik}$ and $r_{ik}$ define positions of the left and right leaves, respectively. There are also $n$ binary variables for each row $i$ at control point $k$, and the binary variable $z_{ijk}$ represents the beamlet $j$ at this row and takes value of 1 if it is open. Only the beamlets between the leaf pairs are open. In Figure 4 an aperture consisting of five leaf pairs ($m = 5$) and five columns ($n = 5$) at control point $k$ is illustrated with corresponding decision variables. Note that at row 3 the right leaf is at its home position and takes value $n + 1 = 6$, and at row 5 the left leaf is at its home position and takes value 0. In Table 1 we list all parameters and decision variables used in the MILP formulation VMATP.

Table 1: Parameters and decision variables

| Parameter | Definition |
|---|---|
| $i$ | Index for an MLC row ($i$=1,...,$m$). |
| $j$ | Index for an MLC column ($j$=0,...,$n$+1), 0 and $n$+1 are home positions of the left and the right leaves, respectively. |
| $k$ | Index for a control point ($k$=1,...,$K$). |
| $t$ | Index for a target volume (TV) ($t$=1,...,$T$). |
| $o$ | Index for an organ at risk (OAR) volume ($o$=1,...,$O$). |
| $v$ | Index for a voxel in a volume. |
| $V_t^{TV}$ | Set of voxels in TV $t$. |
| $V^{TV}$ | Set of all voxels in all TVs, $V^{TV} = \bigcup_{t=1}^{T} V_t^{TV}$. |
| $V_o^{OAR}$ | Set of voxels in OAR volume $o$. |
| $V^{OAR}$ | Set of all voxels in all OAR volumes, $V^{OAR} = \bigcup_{o=1}^{O} V_o^{OAR}$. |
| $V$ | Set of all voxels, $V = V^{TV} \cup V^{OAR}$. |
| $L_t^{TV}$ | Lower bound on the total amount of radiation dose absorbed by a target voxel in TV $t$ (in Gy). |
| $U_t^{TV}$ | Upper bound on the total amount of radiation dose absorbed by a target voxel in TV $t$ (in Gy). |
| $U_o^{OAR}$ | Tolerance radiation dose amount of OAR volume $o$ (in Gy). |
| $\overline{d}_t$ | The prescribed dose for TV $t$ (in Gy). |
| $D_{ijkv}$ | Dose influence matrix (in Gy/MU). |
| $\delta$ | The maximum allowable distance (in beamlet) that a leaf can move between two consecutive control points. |
| $\alpha_t^{TV}$ | The minimum ratio of voxels in TV $t$ that receive radiation at least the prescribed dose $\overline{d}_t$. |
| $\alpha_o^{OAR}$ | The minimum ratio of voxels in OAR volume $o$ that receive radiation at most the tolerance dose $U_o^{OAR}$. |
| $L^{mu}$ | Lower bound on radiation dose intensity at a control point (in MU). |
| $U^{mu}$ | Upper bound on radiation dose intensity at a control point (in MU). |
| **Variable** | **Definition** |
| $l_{ik}$ | Nonnegative integer variable that represents the position of the left leaf (i.e. the rightmost beamlet closed by the left leaf on row $i$ at control point $k$). |
| $r_{ik}$ | Nonnegative integer variable that represents the position of the right leaf (i.e. the leftmost beamlet closed by the right leaf on row $i$ at control point $k$). |
| $z_{ijk}$ | Binary variable, whose value is 1 if the $j$th beamlet of row $i$ at control point $k$ is open, 0 otherwise ($j$=1,...,$n$). |
| $mu_k$ | Nonnegative continuous variable, which stands for radiation dose intensity (in MU) at control point $k$. |
| $d_v$ | Nonnegative continuous variable, which represents the total amount of radiation dose absorbed by voxel $v$ (in Gy). |
| $a_{ijk}$ | Nonnegative continuous variable, which represents radiation dose intensity (in MU) delivered from the $j$th beamlet of row $i$ at control point $k$. |
| $\xi_t^{TV}$ | Continuous variable used in constraint (13), which represents the radiation dose absorbed by the $((1-\alpha_t^{TV})\lvert V_t^{TV}\rvert)$th voxel in TV $t$ receiving the lowest radiation. |
| $\xi_o^{OAR}$ | Continuous variable used in constraint (15), which represents the radiation dose absorbed by the $((1-\alpha_o^{OAR})\lvert V_o^{OAR}\rvert)$th voxel in OAR volume $o$ receiving the highest radiation. |
| $x_{tv}$ | Auxiliary variable for voxel $v$ in TV $t$. |
| $y_{ov}$ | Auxiliary variable for voxel $v$ in OAR volume $o$. |

VMATP:

$$\min \sum_{k=1}^{K} mu_k \tag{1}$$

s.t.

$$r_{ik} - jz_{ijk} \geq 1 \qquad i = 1,\ldots,m;\ j = 1,\ldots,n;\ k = 1,\ldots,K \tag{2}$$

$$(n+1-j)z_{ijk} + l_{ik} \leq n \qquad i = 1,\ldots,m;\ j = 1,\ldots,n;\ k = 1,\ldots,K \tag{3}$$

$$r_{ik} - l_{ik} - \sum_{j=1}^{n} z_{ijk} = 1 \qquad i = 1,\ldots,m;\ k = 1,\ldots,K \tag{4}$$

$$l_{i(k+1)} - l_{ik} \leq \delta \qquad i = 1,\ldots,m;\ k = 1,\ldots,K-1 \tag{5}$$

$$l_{ik} - l_{i(k+1)} \leq \delta \qquad i = 1,\ldots,m;\ k = 1,\ldots,K-1 \tag{6}$$

$$r_{i(k+1)} - r_{ik} \leq \delta \qquad i = 1,\ldots,m;\ k = 1,\ldots,K-1 \tag{7}$$

$$r_{ik} - r_{i(k+1)} \leq \delta \qquad i = 1,\ldots,m;\ k = 1,\ldots,K-1 \tag{8}$$

$$a_{ijk} \leq U^{mu} z_{ijk} \qquad i = 1,\ldots,m;\ j = 1,\ldots,n;\ k = 1,\ldots,K \tag{9}$$

$$a_{ijk} \geq mu_k - U^{mu}(1 - z_{ijk}) \qquad i = 1,\ldots,m;\ j = 1,\ldots,n;\ k = 1,\ldots,K \tag{10}$$

$$a_{ijk} \leq mu_k \qquad i = 1,\ldots,m;\ j = 1,\ldots,n;\ k = 1,\ldots,K \tag{11}$$

$$d_v - \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{k=1}^{K} D_{ijkv} a_{ijk} = 0 \qquad v \in V = V^{TV} \cup V^{OAR} \tag{12}$$

$$\xi_t^{TV} - \frac{1}{(1-\alpha_t^{TV})|V_t^{TV}|} \sum_{v \in V_t^{TV}} x_{tv} \geq \overline{d}_t \qquad t = 1,\ldots,T \tag{13}$$

$$x_{tv} \geq \xi_t^{TV} - d_v \qquad t = 1,\ldots,T;\ v \in V_t^{TV} \tag{14}$$

$$\xi_o^{OAR} + \frac{1}{(1-\alpha_o^{OAR})|V_o^{OAR}|} \sum_{v \in V_o^{OAR}} y_{ov} \leq U_o^{OAR} \qquad o = 1,\ldots,O \tag{15}$$

$$y_{ov} \geq d_v - \xi_o^{OAR} \qquad o = 1,\ldots,O;\ v \in V_o^{OAR} \tag{16}$$

$$d_v \geq L_t^{TV} \qquad t = 1,\ldots,T;\ v \in V_t^{TV} \tag{17}$$

$$d_v \leq U_t^{TV} \qquad t = 1,\ldots,T;\ v \in V_t^{TV} \tag{18}$$

$$mu_k \geq L^{mu} \qquad k = 1,\ldots,K \tag{19}$$

$$mu_k \leq U^{mu} \qquad k = 1,\ldots,K \tag{20}$$

$$\mathbf{l} \in \mathbb{Z}_+^{m\times K};\ \mathbf{r} \in \mathbb{Z}_+^{m\times K};\ \mathbf{z} \in \{0,1\}^{m\times n\times K} \tag{21}$$

$$\mathbf{mu} \in \mathbb{R}_+^{K};\ \mathbf{a} \in \mathbb{R}_+^{m\times n\times K} \tag{22}$$

$$\mathbf{d} \in \mathbb{R}_+^{|V|};\ \mathbf{x} \in \mathbb{R}_+^{|V^{TV}|};\ \mathbf{y} \in \mathbb{R}_+^{|V^{OAR}|} \tag{23}$$

$$\boldsymbol{\xi}^{\boldsymbol{TV}} \in \mathbb{R}^{T};\ \boldsymbol{\xi}^{\boldsymbol{OAR}} \in \mathbb{R}^{O}. \tag{24}$$

There are two main parts of the constraint set: the first part is associated with geometry constraints (2)–(8), which generate a feasible aperture for each one of the control points, and the second part is associated with clinical requirements (9)–(20) (i.e. radiation dose prescriptions). Constraints (2)–(4) satisfy consecutive ones property of the MLC system. They ensure that only $z_{ijk}$ variables between the left and right leaves are 1, and others are 0. Constraints (4) also prevent leaves from overlapping in row $i$ at control point $k$; the right leaf must be on the right side of the left leaf. Constraints (5)–(8) control the leaf movement during the rotation of the gantry. There is an upper limit of the speed of the gantry (this actually limits the maximum distance that a leaf can move in a time unit). Similar to the study of Peng et al. (2015) we assume that the speed of the gantry is constant during rotation but dose rate may change and the maximum allowable distance $\delta$ between two adjacent control points is calculated in terms of beamlets. Note that these constraints force the aperture shapes at two adjacent control points to be similar. Therefore, if the speed of the gantry is high, the amount of $\delta$ is small, which increases the similarity between adjacent apertures.

The second part of the constraint set is related to clinical requirements and ensures that the partial and full volume constraints are satisfied. In order to formulate these constraints, (9)–(12) calculate total amount of radiation dose absorbed by voxel $v$ using the dose influence matrix $\mathbf{D}$, which includes the dose contribution of a beamlet at any control point in Gray per MU (Gy/MU) when a voxel is given a unit of radiation from this beamlet. These constraints are equivalent linearization of

$$d_v = \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{K} D_{ijkv} z_{ijk} mu_k \qquad\qquad v \in V = V^{TV} \cup V^{OAR}$$

by means of McCormick's envelope (McCormick, 1976). The constraint includes nonlinear terms created by the product of binary variables $\mathbf{z}$ with continuous variables $\mathbf{mu}$. The nonnegative continuous variable $mu_k$ is the radiation dose intensity at control point $k$. Radiation can pass through only the open beamlets, and thus $a_{ijk}$ value related to an open beamlet equals to $mu_k$, however it is 0 if the beamlet is closed. Partial-volume constraints (13)–(14) for each target volume (TV) $t$ are formulated using Conditional Value At Risk (CVaR) (Rockafellar et al., 2000). There are other studies (Romeijn et al., 2006; Gozbasi, 2010) that use this approach while formulating IMRT and VMAT planning problems. The average radiation dose of (1-$\alpha_t^{TV}$)$|V_t^{TV}|$ voxels absorbing the lowest radiation in TV $t$ (*lower mean tail dose at level* $\alpha_t^{TV}$) is ensured to be at least the prescribed dose $\overline{d}_t$. Similarly, for each OAR $o$, by constraints (15)–(16) the average dose of (1-$\alpha_o^{OAR}$)$|V_o^{OAR}|$ voxels absorbing the highest doses (*upper mean tail dose at level* $\alpha_o^{OAR}$) is forced to be at most its tolerance dose limit $U_o^{OAR}$. There are also full volume constraints (17)–(18) for each TV $t$ and ensure that each voxel in this volume absorbs radiation within its prescribed limits to prevent cold and hot points in tumor. There are also limits for radiation intensity at each control point (19)–(20), since the linear accelerator can deliver a limited amount of radiation in a time unit. Therefore, $L^{mu}$ and $U^{mu}$ depend on these dose rate limits and speed of the gantry. We assume that the speed of the gantry is constant during rotation and total time to complete a tour is determined before treatment planning. Therefore, we calculate these limits of radiation intensity at each

control point in advance. Equations (21)–(24) define the variables. Finally, the objective function (1) minimizes total radiation intensity (in MU) delivered to the patient during a treatment.

## 4. Branch-and-price Algorithms

Branch-and-price is the adaptation of column generation for the exact solution of integer programming problems. At each node of the branch-and-bound tree, column generation is used to solve linear programming relaxation of the reformulation. It is successfully applied to different integer programming problems such as routing, scheduling, and set partitioning problems. Efficiency of the method depends heavily on the problem structure and it is implementation dependent. There is a number of algorithmic issues that occur during implementation, and the proposed algorithms for solving these issues require problem specific solution approaches. In Lübbecke (2011) a general framework of the method and common algorithmic issues that practitioners may encounter are explained in detail. In addition, Vanderbeck (1994) and Desaulniers et al. (2006) present a number of different types of problems that branch-and-price methods have been applied. To the best of our knowledge, we apply branch-and-price method to solve VMAT planning problem for the first time. In the following sections we provide implementation details of the method to VMAT planning as well as solution approaches for the problems that we encounter.

### *4.1. Column Generation Formulations*

Optimal solution of the MILP model VMATP yields a feasible VMAT treatment plan with minimum MUs consisting of a feasible treatment arc (i.e. $K$ sequential apertures, each for one of the control points, satisfying the consecutive ones property and leaf motion limitations) and radiation intensity $mu_k$ delivered to the patient body through the corresponding aperture at control point $k$. We observe that it is possible to reformulate VMATP in such a way that each feasible treatment arc is considered as a column. Let $Z = \{\mathbf{z}^1, \mathbf{z}^2, \ldots, \mathbf{z}^e, \ldots, \mathbf{z}^{|Z|}\}$ be the bounded set of all feasible treatment arcs (i.e. $\mathbf{z}^e = \{z^e_{ijk} \in \{0,1\},\ \ i = 1,\ldots,m; j = 1,\ldots,n; k = 1,\ldots,K\}$). When we only consider the consecutive ones property and if we assume that there are no leaf motion limitations between consecutive control points, then total number of feasible treatment arcs $|Z|$ equals to $\left(\frac{1}{2}(n+1)(n+2)\right)^{Km}$, which is very large. Row $i$ of the MLC system must satisfy consecutive ones property at control point $k$ and $k+1$, and also must satisfy the maximum leaf motion limitations. Thus, row $i$ at control points $k$ and $k+1$ are dependent. However, there is no dependency between row $i$ and other rows at any control point $k$. As a result, rows of a treatment arc are independent. We can decompose a treatment arc into *m treatment row arc*s, and it is possible to consider each feasible treatment row arc for each row $i$ as a new column. Let $Z_i = \{\mathbf{z}^1_i, \mathbf{z}^2_i, \ldots, \mathbf{z}^e_i, \ldots, \mathbf{z}^{|Z_i|}_i\}$ be the bounded set of all feasible treatment row arcs for row $i$ satisfying consecutive ones property and leaf motion limitations ($\mathbf{z}^e_i = \{z^e_{ijk} \in \{0,1\},\ \ j = 1,\ldots,n; k = 1,\ldots,K\}$). As known from integer programming theory it is possible to express a bounded set $Z_i = \{\mathbf{z}_i \in \{0,1\}^{nK}, \mathbf{A} \in \mathbb{R}^{p\times nK}, \mathbf{h} \in \mathbb{R}^p : \mathbf{Az} \leq \mathbf{h}\}$ equivalently as $\{\mathbf{b}_i \in \{0,1\}^{|Z_i|} : \sum_{e=1}^{|Z_i|} b^e_i \mathbf{z}^e_i, \sum_{e=1}^{|Z_i|} b^e_i = 1\}$, where $\mathbf{z}^e_i \in \{0,1\}^{nK}\ e = 1,\ldots|Z_i|$ are the feasible solutions of $Z_i$. Here, binary variable $b^e_i$ indicates whether the feasible row arc $\mathbf{z}^e_i$ is selected ($b^e_i = 1$) or not ($b^e_i = 0$). Thus, it is

possible to represent binary variable $z_{ijk}$ by $\sum_{e=1}^{|Z_i|} b_i^e z_{ijk}^e$. Then, the resulting treatment row arc based reformulation of VMATP, in other words, the master problem (MP) can be written as

MP:

$$\min \sum_{k=1}^{K} mu_k \tag{1}$$

s.t.

$(11) - (20);\ \ (22) - (24),$

$$\sum_{e=1}^{|Z_i|} b_i^e = 1 \qquad i = 1, \ldots, m \tag{25}$$

$$-a_{ijk} + U^{mu} \sum_{e=1}^{|Z_i|} b_i^e z_{ijk}^e \geq 0 \qquad i = 1, \ldots, m;\ \ j = 1, \ldots, n;\ \ k = 1, \ldots, K \tag{26}$$

$$a_{ijk} - mu_k - U^{mu} \sum_{e=1}^{|Z_i|} b_i^e z_{ijk}^e \geq -U^{mu} \quad i = 1, \ldots, m;\ \ j = 1, \ldots, n;\ \ k = 1, \ldots, K \tag{27}$$

$$\mathbf{b} \in \{0, 1\}^{\sum_{i=1}^{m} |Z_i|}, \tag{28}$$

where the convexity constraints (25) ensure that exactly one feasible treatment row arc is selected for each row $i$. Note that Constraints (9) and (10) are replaced with Constraints (26) and (27) as explained above in detail. Therefore, Constraints (26)–(27) and (11) guarantee that the radiation intensity of each beamlet $a_{ijk}$ equals $mu_k$ when this beamlet is open, and 0 if it is closed. Again if we do not consider leaf motion limitations then total number of feasible row arcs for each row is $\left(\frac{1}{2}(n+1)(n+2)\right)^K$. Hence, total number of feasible row arcs is $m\left(\frac{1}{2}(n+1)(n+2)\right)^K$, which is still very large, and the reformulated problem is not tractable due to the exponential number of columns. We can solve the linear programming relaxation of MP (LMP) by column generation starting with a restricted LMP (RLMP) model, which includes a subset of feasible row arcs $Z_i^0$ for each row.

RLMP:

$$\min \sum_{k=1}^{K} mu_k \tag{1}$$

s.t.

$(11) - (20);\ \ (22) - (24),$

$$\sum_{e=1}^{|Z_i^0|} b_i^e = 1 \qquad i = 1, \ldots, m \tag{29}$$

$$-a_{ijk} + U^{mu} \sum_{e=1}^{|Z_i^0|} b_i^e z_{ijk}^e \geq 0 \qquad i = 1, \ldots, m; \;\; j = 1, \ldots, n; \;\; k = 1, \ldots, K \tag{30}$$

$$a_{ijk} - mu_k - U^{mu} \sum_{e=1}^{|Z_i^0|} b_i^e z_{ijk}^e \geq -U^{mu} \qquad i = 1, \ldots, m; \;\; j = 1, \ldots, n; \;\; k = 1, \ldots, K \tag{31}$$

$$\mathbf{b} \in \mathbb{R}_+^{\sum_{i=1}^{m} |Z_i^0|}. \tag{32}$$

Let $\hat{\lambda}_i$, $\hat{\beta}_{ijk}^1$ and $\hat{\beta}_{ijk}^2$ be an optimal dual solution of RLMP associated with constraints (29)–(31). Then we have $m$ subproblems (pricing subproblems (PSPs)) one for each row:

$\text{PSP}_\text{i}$:

$$\min \; -U^{mu} \sum_{k=1}^{K} \sum_{j=1}^{n} (\hat{\beta}_{ijk}^1 - \hat{\beta}_{ijk}^2) z_{ijk} \tag{33}$$

s.t.

$$r_{ik} - j z_{ijk} \geq 1 \qquad j = 1, \ldots, n; \;\; k = 1, \ldots, K \tag{34}$$

$$(n + 1 - j) z_{ijk} + l_{ik} \leq n \qquad j = 1, \ldots, n; \;\; k = 1, \ldots, K \tag{35}$$

$$r_{ik} - l_{ik} - \sum_{j=1}^{n} z_{ijk} = 1 \qquad k = 1, \ldots, K \tag{36}$$

$$l_{i(k+1)} - l_{ik} \leq \delta \qquad k = 1, \ldots, K-1 \tag{37}$$

$$l_{ik} - l_{i(k+1)} \leq \delta \qquad k = 1, \ldots, K-1 \tag{38}$$

$$r_{i(k+1)} - r_{ik} \leq \delta \qquad k = 1, \ldots, K-1 \tag{39}$$

$$r_{ik} - r_{i(k+1)} \leq \delta \qquad k = 1, \ldots, K-1 \tag{40}$$

$$\mathbf{l} \in \mathbb{Z}_+^K; \;\; \mathbf{r} \in \mathbb{Z}_+^K; \;\; \mathbf{z} \in \{0, 1\}^{n \times K}. \tag{41}$$

In our algorithm, we solve the master problem (MP) by branch-and-price method where at each node of the branch-and-bound tree column generation is used to solve LMP. Each time after solving RLMP, we solve $m$ PSPs separately and introduce a new treatment row arc of row $i$ to RLMP only if its reduced cost is negative. We obtain dual optimal price $\hat{\lambda}_i$ associated with constraints (29) for $i = 1, ..., m$ and check whether the optimal value of $\text{PSP}_\text{i}$ is less than $\hat{\lambda}_i$ or not. Being less than $\hat{\lambda}_i$ means that the reduced cost of the corresponding treatment row arc is negative. On the other hand, if optimal objective value of each $\text{PSP}_\text{i}$ yields nonnegative reduced cost, then the column generation iterations stop. Simply, we are at an optimal solution of the LMP. If all $b_i^e$ variables are integer at the optimality then we have also an optimal solution of MP and VMATP. However, if at least one of them is fractional then we continue with branching and solve the modified restricted models by column generation at the new branches.

*4.2. Generating Columns by Solving Shortest Path Problems*

We observe in our preliminary experiments that solving PSPs by using a commercial MIP solver is inefficient. The variation of computation time between iterations is high and it may take too long to generate a column at some iterations. Thus we formulate the pricing subproblems as shortest path problems similar to Mahnam et al. (2017), Gozbasi (2010) and Boland et al. (2004). We explain the shortest path problem formulation of PSP on a small example. Figure 5 illustrates the network representation of $PSP_1$ for the first row of the problem given in Figure 3. There are only three control points ($K = 3$) and three beamlets in a row ($n = 3$). Note that the home positions of the leaves are $j = 0$ and $j = 4$ for the left and right leaves, respectively. The beamlets that are blocked by the leaves are dark gray, and open beamlets are shown as white rectangles. There are two additional nodes in the figure: start and finish nodes. For each one of the control points there are $\frac{(n+1)(n+2)}{2} = 10$ different leaf configurations. For example, the leaf configurations at the top of the figure represent that both leaves are at their home positions and all beamlets are open. Observe that at a control point there are four different combinations of the leaves for closing all beamlets, since the left and the right leaves may be adjacent in four different ways. Moreover, it is assumed that the leaves can move at most one beamlet between consecutive control points (i.e. $\delta = 1$). The leaf configurations at each one of the control points represent the nodes. Also, an arc between two nodes at two adjacent control points indicate that the maximum leaf movement limitations are satisfied, namely these two consecutive leaf configurations are compatible. If the new position of each one of the left and right leaves at the next control point is in its allowable range then there is an arc. Thus, the arcs in the graph represent feasible movements. Observe that the arcs are directed (since the gantry rotates in one direction) and there are no arcs between any two nodes at the same control point. The costs of an arc that connects two nodes at control points $k$ and $k + 1$ is computed as $-U^{mu} \sum_{j=l+1}^{j=r-1} (\hat{\beta}^1_{ijk} - \hat{\beta}^2_{ijk})$. The cost of an arc that connects a node at the last control point $K$ and the finish node is $-U^{mu} \sum_{j=l+1}^{j=r-1} (\hat{\beta}^1_{ijK} - \hat{\beta}^2_{ijK})$, and the cost of an arc between the start node and a node at the first control point is zero. Our aim is to find a path from the start node to the finish node with minimum cost. To sum up, we obtain a directed, acyclic and layered graph consisting of $K$ layers which correspond to $K$ control points. In each layer there are $\frac{(n+1)(n+2)}{2}$ nodes. If we assume that there are no leaf motion limitations between adjacent control points then the total number of arcs in the graph will be $\frac{(K-1)}{4}(n+1)^2(n+2)^2 + (n+1)(n+2)$. We solve this problem using dynamic programming and the upper bound for the complexity of the algorithm is $O(Kn^4)$, which is a polynomial. The optimal solution of this problem yields one of the treatment row arcs with minimum reduced cost and if its reduced cost is negative, then we add this resulting column to RLMP. For example, the shortest path given in Figure 5 with solid lines indicates the treatment row arc illustrated in Figure 6. At the first control point the left leaf is at its home position and the right leaf blocks the second and third beamlets. Only the first beamlet is open. During the movement of the gantry from the first control point to the second one, the left leaf moves one beamlet to the right and blocks the first beamlet. On the other hand, the right leaf moves one beamlet to the right and opens the second beamlet. Finally, during the travel of the gantry from the second control point to the last one, the left leaf returns to its home position and stops blocking the first beamlet. However, the right beamlet does not move, and only the last beamlet is blocked. The reduced cost of this shortest path is equal

to $-\hat{\lambda}_1 - U^{mu}\left((\hat{\beta}^1_{111} - \hat{\beta}^2_{111}) + (\hat{\beta}^1_{122} - \hat{\beta}^2_{122}) + (\hat{\beta}^1_{113} - \hat{\beta}^2_{113} + \hat{\beta}^1_{123} - \hat{\beta}^2_{123})\right)$.

The study of Boland et al. (2004) is one of the leading papers that uses network models in radiation therapy planning. They use their network model in their column generation approach to solve MLC leaf sequencing (MLS) problem, which is the last phase of IMRT planning. They decompose a given fluence map into a number of feasible apertures with radiation intensities. They design their network in such a way that each layer corresponds to a leaf pair. Therefore, there are as many layers as the number of rows of the fluence map. At each layer the nodes represent the potential positions of the left and right leaves satisfying the consecutive ones property. There are arcs between two nodes at two adjacent layers if the leaf configurations at these rows satisfy interdigitation constraints. Hence, a path in the network corresponds to an aperture, which is feasible with respect to the consecutive ones property and interdigitation constraints. Mahnam et al. (2017) use a similar approach to generate a set of sequential apertures in VMAT planning. They consider that a full treatment arc (i.e. a 360°-arc) consists of a number of sequential partial arcs with the same length (i.e. there are 18 20°-arcs in a full arc). Also, each of these partial arcs includes a number of equally spaced apertures (i.e. 10 apertures with 2° spacing at a 20°-arc). They can generate a partial arc, row by row, using a network model since they consider only the consecutive ones property. In their network model, the number of layers equals to the number of apertures in a partial arc and the nodes represents the leaf configurations, which is similar to the model of Boland et al. (2004). On the other hand, there is an arc between two nodes at two consecutive layers if the corresponding leaf configurations satisfy leaf motion limitations. After solving $m$ subproblems by a shortest path algorithm, they take the union of the resulting partial row arcs to obtain a partial arc. Also, they need to join a number of partial arcs to obtain a full treatment arc, which necessitates a post-optimization (i.e. the intersection points of adjacent partial arcs may be incompatible due to leaf motion limitations). In our study, pricing subproblem generates a full row arc, which is feasible with respect to leaf motion limitations. Hence, we do not need any post-processing operation. Another difference is that the union of $m$ row arcs yields a feasible full treatment arc. Finally, our network design is similar to the one proposed in Gozbasi (2010), which is used to solve VMAT planning problem in a two-stage heuristic approach. They generate a feasible full treatment arc in the first stage where the costs of the arcs are calculated using a beamlet scoring algorithm. In the second stage, they find radiation intensity of each one of the apertures in the treatment arc.

### *4.3. Branching*

At the root node of the branch-and-bound tree if the optimal values of all $b^e_i$ variables are integral then we are at an optimal solution of the problem. Otherwise, we apply branching and solve the resulting restricted linear programming model at each one of the branch-and-bound tree nodes. It is important to find a branching strategy that prevents regenerating columns that are previously prohibited. Also, the columns generated so far must be divided into two groups and it must be possible to modify the PSP so that generating infeasible columns due to the branching constraints is prevented. It is known that applying the ordinary variable branching (dichotomized branching on a $b^e_i$ variable with fractional value) is not efficient (Savelsbergh, 1997). Instead, we branch on the original variables of VMATP; a

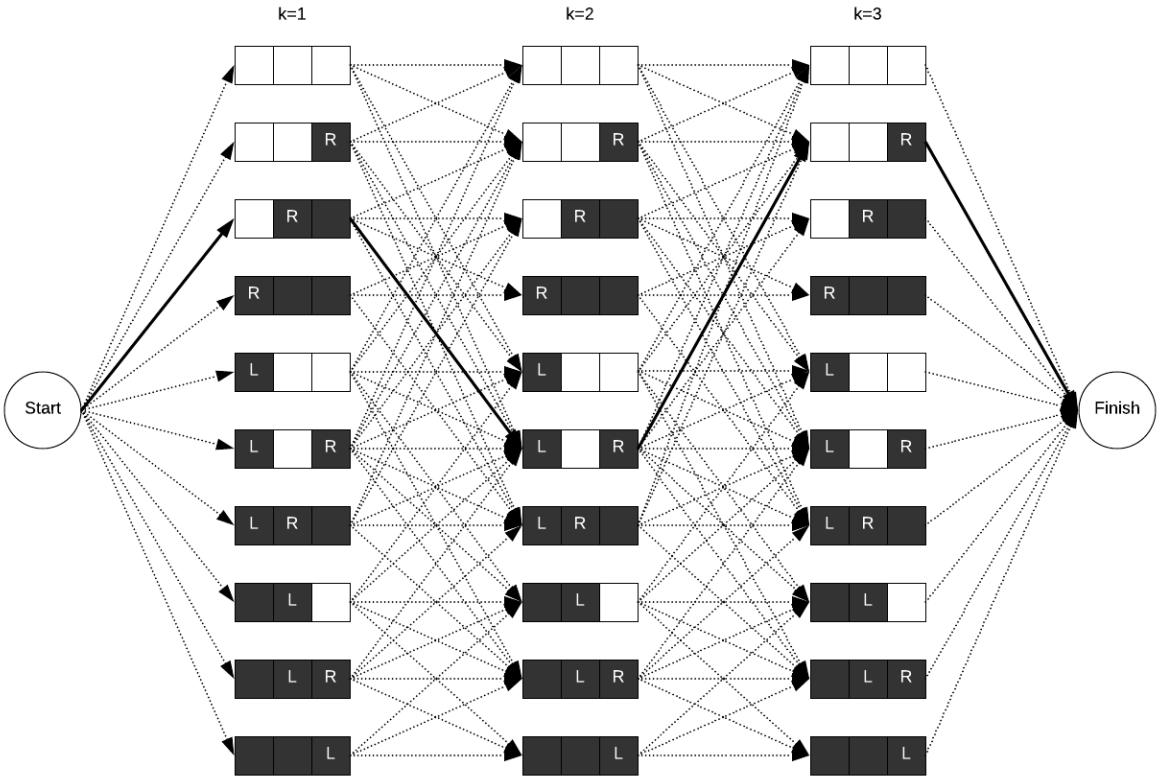


Figure 5: Network representation of $\text{PSP}_1$ for the first row of the treatment arc given in Figure 3 ($K = 3$, $n = 3$)

beamlet with a fractional $\hat{z}_{ijk}$ value. If the optimal solution of LMP at a node is not integer then for at least one row $i$ there must be at least two fractional $b_i^e$ variables. Thus, there must be at least one beamlet with fractional $\hat{z}_{ijk}$ value. Observe that when there are two fractional variables $b_i^1$ and $b_i^2$ in the current solution with columns $\mathbf{z}_i^1$ and $\mathbf{z}_i^2$, respectively, then there must be at least one beamlet $(i, j, k)$ having value 1 in exactly one of these columns. As a result, $\hat{z}_{ijk} = b_i^1 \hat{z}_{ijk}^1 + b_i^2 \hat{z}_{ijk}^2$ becomes fractional. In our branching rule we choose one of these fractional beamlets as the branching variable using the following simple search mechanism: for each control point $k$ we first calculate the ratio $\frac{\sum_{(i,j)} \hat{z}_{ijk}}{\sum_{\substack{(i,j) \\ \hat{z}_{ijk}>0}} 1}$. Observe that if there is at least one fractional beamlet, then this ratio is strictly between 0 and 1, and the corresponding aperture becomes fractional. Note that if all beamlets of an aperture at a control point take only 0 or 1 value, then we do not consider this control point. For each control point $k$ with fractional aperture, we calculate the value $\gamma_k = \hat{mu}_k \left( \frac{\sum_{(i,j)} \hat{z}_{ijk}}{\sum_{\substack{(i,j) \\ \hat{z}_{ijk}>0}} 1} \right)$. Then we select the one with the highest $\gamma_k$ value. We branch on the beamlet at this control point, which is fractional and closest to 1. Thus, we seek a beamlet belonging to an aperture having high radiation intensity and low fractionality. Let us denote the selected beamlet by $z_{ijk}$. We then obtain two child nodes: at one of them beamlet $(i, j, k)$ is open and at the other one it is closed. In the first child node, we remove the respective

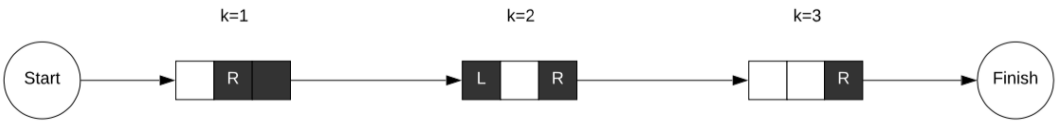


Figure 6: The treatment row arc obtained in Figure 5

arcs in the $PSP_i$'s network model that connect the nodes at control point $k-1$ to the nodes at control point $k$ where the beamlet $(i, j, k)$ is closed. Also, we set $b_i^e$ variables to 0 if the value of beamlet $(i, j, k)$ is 0 in the corresponding column. In the second child node, we remove all arcs that connect the nodes at control point $k-1$ to the nodes at control point $k$ where the beamlet $(i, j, k)$ is open. Also we set all $b_i^e$ variables to 0 if beamlet $(i, j, k)$ takes value 1 in the corresponding columns. Note that, at each branching we use only one beamlet $(i, j, k)$ that belongs to one row ($i$th row), thus we partition only the columns associated with row $i$. Furthermore, at each node the PSPs are modified taking into account all branching decisions leading to the current node.

### *4.4. Initial Columns of RLMP*

We generate $m$ initial columns at the beginning of the BP algorithm by solving the following model

$$\min \sum_{k=1}^{K} mu_k \tag{1}$$

s.t.

$$(2)-(9);\ \ (11);\ \ (19)-(22),$$

$$\sum_{k=1}^{K}\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ijk}D_{ijkv} \geq L_t^{TV} \qquad t = 1,\ldots,T;\ \ v \in V_t^{TV}, \tag{42}$$

which includes all geometry constraints and valid inequalities (9) and (11). We observe that if a treatment arc consists of $K$ apertures unable to deliver enough radiation to satisfy lower limits of voxels in each TV, then this treatment arc cannot yield a feasible solution for VMATP. In other words, a treatment arc must satisfy that each target voxel absorbs radiation at least the prescribed lower limit (i.e. $L_t^{TV}$). Thus, we generate the initial set of $m$ columns by considering this observation and avoiding such infeasible treatment arcs. Optimal solution of the model given above yields $K$ aperture shapes, namely all $z_{ijk}$ values. Hence, for each row $i$ we obtain a treatment row arc and use them to construct initial RLMP model. The initial treatment arc can yield a feasible solution for VMATP, namely the resulting RLMP may provide a solution that satisfies all prescription radiation doses. As a result, we obtain an upper bound for the algorithm since we start with only one column for each row and all $b_i^1$ values are 1 ($i = 1, ..., m$). However, this is not always the case and the initial RLMP may be infeasible. To resolve this problem we add one artificial variable to RLMP for each one of the constraints (13) and (15), and penalize them in the objective function. We keep these artificial variables at all nodes of the branch-and-bound tree and the infeasibility of RLMP is almost completely

removed. Note that there are also full volume constraints in RLMP associated with target voxels (constraints (17) and (18)). The initial columns generated by the formulation given above guarantee that constraint (17) is satisfied at root node. However, RLMP may not satisfy constraint (18), or at successor nodes it may be infeasible due to the branching constraints that cause absence of relevant columns to satisfy constraint (17). This case is rarely encountered, yet we resort to Farkas Pricing as explained by Lübbecke (2011) in detail. Also, the details of initialization using artificial variables is explained by Vanderbeck (1994).

### *4.5. Lower Bounds*

Column generation methods often suffer from *tailing off effect*: at initial iterations a near optimal solution is reached quickly but in the following iterations the improvement in the objective value becomes very small and the algorithm terminates in very long time (Lübbecke and Desrosiers, 2005). We also experience this effect; it takes very long to prove optimality and terminate column generation iterations at each node of branch-and-bound tree. In order to alleviate this problem we strengthen the lower bound using a Lagrangian relaxation approach. We dualize the complicating constraints (9) and (10) in VMATP with nonnegative multipliers $\mathbf{u} \in \mathbb{R}_+^{m\times n\times K}$ and $\mathbf{g} \in \mathbb{R}_+^{m\times n\times K}$ to obtain the *Lagrangian subproblem* (LSP)

LSP($\hat{\mathbf{u}}, \hat{\mathbf{g}}$):

$$\min \sum_{k=1}^{K} mu_k + \sum_{k=1}^{K}\sum_{i=1}^{m}\sum_{j=1}^{n}(\hat{u}_{ijk}(a_{ijk} - U^{mu}z_{ijk}) + \hat{g}_{ijk}(-U^{mu} - a_{ijk} + mu_k + U^{mu}z_{ijk})) \tag{43}$$

s.t.

(2) – (8); (11) – (24).

It defines a valid dual bound on VMATP for given $\hat{\mathbf{u}}$ and $\hat{\mathbf{g}}$ vectors. In general the best dual bound is obtained by solving the *Lagrangian dual problem* (LD): $\max_{\mathbf{u},\mathbf{g}\geq 0}$ LSP($\mathbf{u}, \mathbf{g}$). LD is a max-min problem and one of the most popular method to solve this problem is the subgradient algorithm (Held et al., 1974), in which at each iteration dual multipliers $\mathbf{u}$ and $\mathbf{g}$ are updated and the resulting LSP($\hat{\mathbf{u}}, \hat{\mathbf{g}}$) problem is solved. According to our preliminary analysis LSP($\mathbf{0}, \mathbf{0}$) provides very strong lower bounds. Therefore, we just solve VMATP after relaxing constraints (9) and (10), which is clearly equivalent to LSP($\mathbf{0}, \mathbf{0}$), and use the optimal value as a lower bound. We note that in this case, it is possible to also remove geometry constraints (2)-(8) from LSP($\mathbf{0}, \mathbf{0}$) problem since they do not have any contribution

to the objective function. As a result, we obtain the following relaxation of VMATP:

RVMATP:

$$\min \sum_{k=1}^{K} mu_k \tag{1}$$

s.t.

$$(11) - (20); \;\; (22) - (24).$$

At each node at the beginning we update and solve this linear programming problem and, if it is possible, we also update the lower bound of the current search node. Note that, since we branch on $z_{ijk}$ variables and in each branch we set one of them to 1 or 0, it is possible to adjust RVMATP by setting each one of the $a_{ijk}$ variables associated with the branching decisions that leads to the current node to either $mu_k$ or 0. Moreover, during the column generation iterations, we update the lower bound if the optimal value of RLMP and sum of the optimal values of all PSPs is larger than the current lower bound. Finally, we use depth-first search as node selection strategy. The flow of the resulting algorithm, which we call Branch-and-Price (BP) Algorithm 1, is given in Figure 7.

### *4.6. Algorithmic Improvements*

We modify BP Algorithm 1 and obtain two more algorithms with some minor differences, which we call BP Algorithm 2 and BP Algorithm 3. In BP Algorithm 2, at root node before branching, we solve the resulting restricted MP including columns generated so far as a MILP model. We update UB if the resulting solution is better than the incumbent.

According to preliminary experiments we observe that the necessary time to solve the model given in Section 4.4, which generates the initial columns, increases as the size of the problem becomes larger. Moreover, it becomes impossible to solve it optimally within the given time limit. Thus, we simplify this part of the algorithm in BP Algorithm 3. The initial columns are generated by solving a different model that consists of only the geometry constraints and a different objective function:

$$\max \sum_{k=1}^{K} \sum_{i=1}^{m} \sum_{j=1}^{n} z_{ijk} \tag{44}$$

s.t.

$$(2) - (8); \;\; (21).$$

An optimal solution of this model gives a full treatment arc with maximum total number of open beamlets that satisfies geometry constraints. Similar to BP Algorithm 2, a better feasible solution for VMATP is sought at the root node. Instead of generating all promising columns before branching, each subproblem is solved only once and the resulting

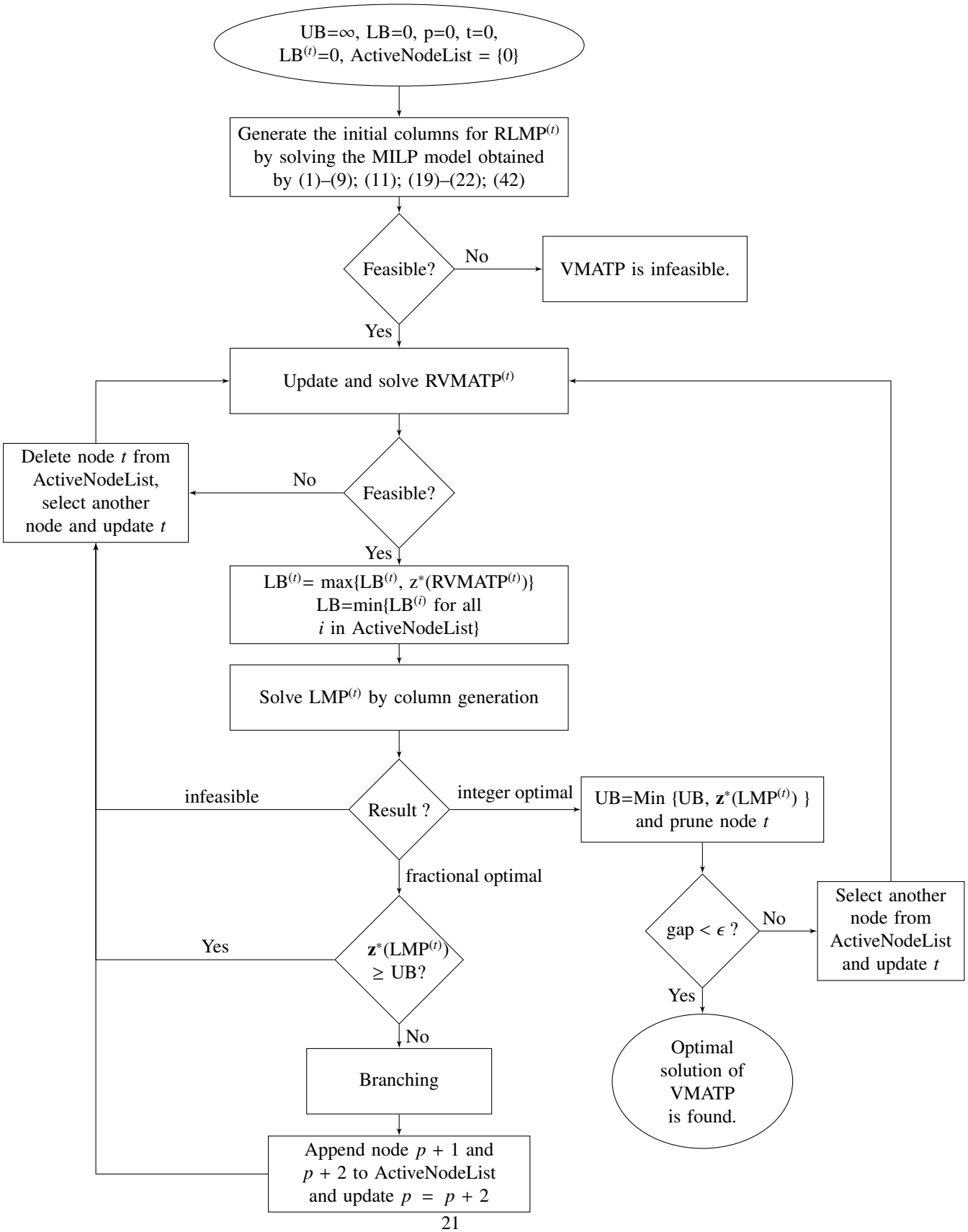




Figure 7: Branch-and-Price Algorithm 1

restricted MP is solved as a MILP model.

## 5. Computational Results

We implement the proposed BP algorithms in Python 2.7 programming language (Python, 2015) and use Gurobi 8.0 as MILP solver (Gurobi, 2018). All tests are carried out on a 64-bit PC with 3.20 GHz Intel(R) Core(TM) i5-6500 CPU and 8 GB of RAM. We first describe the test bed. Then we compare the performance of the proposed algorithms with Gurobi solver and Improved Benders Algorithm 2, which is the best performing algorithm of Dursun et al. (2019).

### *5.1. Test Bed*

In order to test BP algorithms we use a real data set belonging to an anonymous prostate cancer patient provided by Craft et al. (2014a) in common optimization for radiation therapy (CORT) data sets (Craft et al., 2014b). The dose influence matrices in units of Gy per MU are provided in the original prostate data set. There are 690,373 voxels of size 3 $mm^3$ containing 9 defined structures and the remaining part of the body. The defined structures include 2 planning target volumes (PTV 68 and PTV 56) with different prescription doses and 5 OARs (i.e. bladder, left femoral head, right femoral head, penile bulb, and rectum). Since the original problem is intractable due to its size, we reduce the size of the problem by randomly selecting a number of voxels from each one of the structures as listed in Table 2. Note that Table 2 contains the number of voxels in the structures that we consider in the computational experiments. Also, we reduce total number of structures by assuming that there is only one OAR (union of five OARs) and we consider only PTV 68. We derive 18 data sets with different voxel sizes, where each data set has 5 instances generated using a different random number sequence. In total we have 90 instances in the computational experiments. The first 55 instances are used in also Dursun et al. (2019), and in this study we enlarge the test bed by adding 7 new data sets including 35 instances in total. As an example, 600 voxels are selected from PTV68 to form an instance with 1301 voxels, and the remaining voxels are selected from the OARs (240 voxels from bladder, 120 voxels from each one of the femoral heads, 101 voxels from penile bulb, and 120 voxels from rectum). Note that we randomly select voxels from the OARs, and they do not belong to any intersection of the structures. The data sets and a detailed description file are publicly available on our website (Taşkın, 2019).

In the real data set, there are 180 equally spaced control points ($K = 180$) on a coplanar arc, and 25,404 beamlets with size 1 $cm^2$. In VMAT planning, the continuous dose delivery is discretized over a finite number of control points and it is assumed that radiation delivery only occurs at the control points. This assumption is reasonable when there is a large number of control points with typically 2° spacing (Men et al., 2010; Mahnam et al., 2017). Moreover, Otto (2008) indicates that poor sampling of control points and MLC leaf positions can degrade the plan accuracy. Thus, this can result in unacceptable dosimetric error. Hence, beamlet grid and control points are kept in our study as they are in the original data set. There are no dose absorption values for the beamlets that do not belong to beam's-eye-view

(BEV) at the corresponding control point, hence we assume that they are closed during the rotation. Also, we assume that the MLC system has 13 rows and 16 columns ($m = 13, n = 16$), since this size is sufficient to cover all beamlets having dose absorption information.

Table 2: Total number of voxels in the real data set and reduced data sets

| Structure | Voxel | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PTV 68 | 6770 | 10 | 20 | 30 | 40 | 100 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | 1000 | 1100 | 1300 | 1500 | 1650 | 2200 |
| Bladder | 11596 | 4 | 8 | 12 | 16 | 40 | 120 | 160 | 200 | 240 | 280 | 320 | 360 | 400 | 440 | 480 | 520 | 660 | 880 |
| Left f. h. | 5857 | 2 | 4 | 6 | 8 | 20 | 60 | 80 | 100 | 120 | 140 | 160 | 180 | 200 | 220 | 240 | 260 | 330 | 440 |
| Right f. h. | 5974 | 2 | 4 | 6 | 8 | 20 | 60 | 80 | 100 | 120 | 140 | 160 | 180 | 200 | 220 | 240 | 260 | 330 | 440 |
| Penile b. | 101 | 2 | 4 | 6 | 8 | 20 | 60 | 80 | 100 | 101 | 101 | 101 | 101 | 101 | 101 | 101 | 101 | 101 | 101 |
| Rectum | 1764 | 2 | 4 | 6 | 8 | 20 | 60 | 80 | 100 | 120 | 140 | 160 | 180 | 200 | 220 | 240 | 260 | 330 | 440 |
| TOTAL | 32062 | 22 | 44 | 66 | 88 | 220 | 660 | 880 | 1100 | 1301 | 1501 | 1701 | 1901 | 2101 | 2301 | 2601 | 2901 | 3401 | 4501 |

We assume that there is only one arc in VMAT treatment and the gantry completes a tour in 3 minutes with constant speed. The maximum dose rate of the linear accelerator is approximately 600 MUs per minute, which we also use in our experiments. There are 180 control points, and thus at each one of them the gantry delivers radiation for at most one second. As a result the maximum radiation dose intensity $U^{mu}$ is set to 10 MUs. We also assume that the value of $\delta$ is 2, namely a leaf can move at most two beamlets between consecutive control points. $\alpha_1^{OAR}$ and $\alpha_1^{TV}$ are set to 0.40 and 0.95, respectively. There are 34 fractions in the treatment and in each one $\overline{d}_1$ is 2 Gy. The remaining prescribed doses are $U_1^{TV}$, $L_1^{TV}$, and $U_1^{OAR}$ are set to 2.14 Gy, 1.9 Gy, and 1.47 Gy, respectively.

### *5.2. Computational Experiments*

We solve all instances by Gurobi solver, Improved Benders Algorithm 2, and the three new BP algorithms. We set the CPU time limit to 3600 seconds in all experiments and execute all algorithms on one thread in order to keep the conditions the same and to be able to compare the performances of them. In the BP algorithms, RVMATP model is solved using Barrier method at the root node and then its method is changed to dual simplex in the descendant search nodes. We solve RLMP using primal simplex in order to warm start from the last basis after adding a new column. Also, there is a threshold on the reduced cost for new generated columns. If the reduced cost is not below $-0.05$ we do not add the corresponding column to RLMP. We do not perform any other parameter tuning for the Gurobi solver and keep parameters at their default settings. In Table 3 we give the summary of the computational results that includes average optimality gaps (%), CPU times (seconds), total number of instances that the corresponding method can find a feasible solution (S/T) and can solve optimally (O/T) out of total instances. Note that 0 is a valid lower bound for the objective function, total MUs of the treatment, since the amount of radiation intensity at each control point is nonnegative. Similarly, 1800 MUs is a valid upper bound since the radiation intensity at a control point can be at most 10 MUs. Whenever a method can find neither a feasible solution nor a lower bound for an instance, we calculate the optimality gap as 100% using these bounds. Also, when only a lower bound is provided then we calculate the optimality gap using this valid upper bound, 1800 MUs. In Table 4 the computational results that include lower and upper bounds for each one of the instances are provided.

We divide data sets into four groups: small (with 22-220 voxels), medium (with 660-1701 voxels), large (with 1901-2901 voxels), and very large (with 3401 and 4501 voxels). The results of small data sets (with at most 220 voxels) show that Gurobi, Improved Benders Algorithm 2, BP Algorithm 2 and BP Algorithm 3 can solve almost all of the instances optimally in short CPU times. BP Algorithm 1 cannot find a feasible solution for one instance having size 220 within time limit. Gurobi performs better than all of the BP algorithms and Improved Benders Algorithm 2 with respect to the average optimality gaps in all small data sets and also with respect to the average CPU times in data sets with 22, 44, and 66 voxels. In particular, Gurobi can solve all instances optimally except 66-5 and 220-5. BP algorithms and Improved Benders Algorithm 2 cannot solve these instances and also some other instances optimally.

As the size of the problem increases, Gurobi starts failing to solve some instances within time limit. It cannot provide neither an upper bound nor a lower bound for 9 out of 30 instances of medium size (having total number of voxels between 660-1701), and can only provide a lower bound for 2 out of 30 instances. It solves 18 out of the remaining 19 medium size instances optimally in relatively long CPU times. On the other hand, all of the new BP algorithms perform better than Gurobi in both performance measures (only the average optimality gaps of data sets with 660 and 1100 voxels are worse in BP Algorithm 1, and also the average optimality gap of data set with 660 voxels is slightly worse in BP Algorithm 2). They can find a feasible solution for all of the medium size instances. Moreover, BP Algorithm 1, BP Algorithm 2, and BP Algorithm 3 can respectively solve 20, 26, and 29 instances optimally. BP Algorithm 2 cannot solve only 4 instances (660-1, 660-2, 1301-1 and 1301-2), and BP Algorithm 3 cannot solve only one instance (1301-2) optimally. Also, the resulting optimality gaps of these instances are very small (at most 0.02%). Improved Benders Algorithm 2 can also find a feasible solution for all of the medium size instances except 1701-2. The optimality gaps are below 1% in almost all cases (except 660-3, 880-1 and 1100-5), however the number of instances that it can solve optimally is only three. In particular, the smallest average CPU times and optimality gaps are obtained by BP Algorithm 3. As a result, BP Algorithm 3 outperforms the other ones for medium size instances.

The results of large problems (with 1901-2901 voxels) are also similar to the results of medium size problems. Gurobi can solve only 2 out of 25 instances optimally within time limit, and for 22 instances it can neither provide an upper bound nor a lower bound, for the remaining one instance it can only provide a lower bound. On the other hand, BP Algorithm 3 solves all of the instances optimally within less than half of the time limit. The other two BP algorithms can also provide an upper bound for almost all of the instances. BP Algorithm 1 and BP Algorithm 2 cannot provide a feasible solution respectively for 3 and 1 instance within time limit. The number of instances that these algorithms can solve optimally is 17 and 20, respectively. Also, the optimality gaps of the problems, which are not solved optimally, are below 1% in almost all instances. Improved Benders Algorithm 2 can find a feasible solution for 20 instances, but it can solve only one of them optimally. Finally, BP Algorithm 3 outperforms other algorithms in both performance measures with significant differences.

In addition to these three groups of data sets, we generate and solve two larger data sets with 3401 and 4501 voxels to be able to make the difference between algorithms more clear and observe the limits of BP Algorithm 3. Gurobi fails to find lower and upper bounds for all instances. BP Algorithm 1 and BP Algorithm 2 both find a feasible solution

of 5 and 4 out of 10 instances, and they can solve 3 and 2 instances optimally, respectively. On the other hand, BP Algorithm 3 solves 8 instances to optimally. It can only provide a lower bound for each one of the remaining two instances. Improved Benders Algorithm 2 can solve 4 out of 10 instances with small optimality gaps (below 1%), but it cannot solve any of the instances optimally.

In overall BP Algorithm 3 can solve 83 out of 90 instances optimally, and for the remaining 5 instances it can provide very small optimality gaps. (i.e. below 0.1% except instance 22-5). However, Gurobi fails to provide upper and lower bounds for almost half of the instances (i.e. 41 out of 90) within time limit. Also, it cannot provide an upper bound for other 3 instances. It only solves 43 instances optimally and 3 instances with small optimality gaps. If we check the average CPU times and optimality gaps, BP Algorithm 3 outperforms all other methods in both performance measures in almost all data sets (except with 22, 44 and 66 voxels). The last row of Table 3 shows that the minimum average optimality gap of all instances is 1.93%, the minimum average CPU time is 1021.6 seconds, the maximum total number of instances with a feasible solution is 88, and the maximum number of instances that are solved optimally is 83, which are all obtained by BP Algorithm 3.

Finally, we investigate convergence characteristics of BP Algorithm 3 and compare it with the other algorithms. We note that BP Algorithm 3 finds a provably optimal solution for almost all instances that it can solve to optimality (81 out of 83) at the root node. The remaining 2 out of 83 instances, (instances 22-1 and 66-4) are solved to optimality in deeper nodes. Our investigation of gap vs time curves revealed that BP Algorithm 3 is consistently able to reduce optimality gap faster than Gurobi and Improved Benders 2 algorithms, especially for large and very large instances. We also experimented with setting optimality gap stopping criterion to 0.01%, 2% and 5% for all algorithms. As expected, increasing optimality gap limit decreased CPU time spent by all algorithms, but BP Algorithm 3 still spent the smallest amount of CPU time. Results of these additional analyses imply that performance of BP Algorithm 3 is robust with respect to time and optimality gap limits compared to other algorithms.

Table 3: Summary of the computational results (GAP %, CPU in seconds)

| | Gurobi | | | | BP Algorithm 1 | | | | BP Algorithm 2 | | | | BP Algorithm 3 | | | | Impr. Benders Algorithm 2 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SAMPLE | GAP | CPU | S/T | O/T | GAP | CPU | S/T | O/T | GAP | CPU | S/T | O/T | GAP | CPU | S/T | O/T | GAP | CPU | S/T | O/T |
| **22** | 0.00 | 24.2 | 5/5 | 5/5 | 0.00 | 60.6 | 5/5 | 5/5 | 0.00 | 40.7 | 5/5 | 5/5 | 0.03 | 1110.8 | 5/5 | 4/5 | 0.00 | 90.4 | 5/5 | 5/5 |
| **44** | 0.00 | 23.2 | 5/5 | 5/5 | 0.00 | 60.1 | 5/5 | 5/5 | 0.00 | 74.6 | 5/5 | 5/5 | 0.00 | 36.6 | 5/5 | 5/5 | 0.00 | 138.0 | 5/5 | 5/5 |
| **66** | 0.01 | 753.8 | 5/5 | 4/5 | 0.39 | 2176.2 | 5/5 | 2/5 | 0.03 | 2180.6 | 5/5 | 2/5 | 0.03 | 2010.4 | 5/5 | 3/5 | 0.16 | 1610.2 | 5/5 | 3/5 |
| **88** | 0.00 | 79.2 | 5/5 | 5/5 | 0.00 | 64.2 | 5/5 | 5/5 | 0.00 | 76.1 | 5/5 | 5/5 | 0.00 | 44.9 | 5/5 | 5/5 | 0.00 | 130.5 | 5/5 | 5/5 |
| **220** | 0.00 | 1233.7 | 5/5 | 4/5 | 17.36 | 800.9 | 4/5 | 4/5 | 0.01 | 810.9 | 5/5 | 4/5 | 0.00 | 792.4 | 5/5 | 4/5 | 0.03 | 1983.7 | 5/5 | 3/5 |
| **660** | 0.00 | 1914.4 | 5/5 | 5/5 | 1.20 | 1897.4 | 5/5 | 2/5 | 0.01 | 1759.6 | 5/5 | 3/5 | 0.00 | 222.7 | 5/5 | 5/5 | 2.04 | 3151.8 | 5/5 | 1/5 |
| **880** | 20.00 | 2931.7 | 4/5 | 3/5 | 0.00 | 927.8 | 5/5 | 5/5 | 0.00 | 649.5 | 5/5 | 5/5 | 0.00 | 187.1 | 5/5 | 5/5 | 3.07 | 3093.8 | 5/5 | 1/5 |
| **1100** | 0.00 | 2536.6 | 5/5 | 5/5 | 4.48 | 1352.7 | 5/5 | 4/5 | 0.00 | 824.1 | 5/5 | 5/5 | 0.00 | 381.9 | 5/5 | 5/5 | 0.34 | 3259.4 | 5/5 | 1/5 |
| **1301** | 37.35 | 2908.2 | 3/5 | 3/5 | 0.53 | 1907.3 | 5/5 | 3/5 | 0.01 | 2035.3 | 5/5 | 3/5 | 0.00 | 1200.8 | 5/5 | 4/5 | 0.26 | 3600 | 5/5 | 0/5 |
| **1501** | 97.36 | 3600 | 0/5 | 0/5 | 0.00 | 889.0 | 5/5 | 5/5 | 0.00 | 952.1 | 5/5 | 5/5 | 0.00 | 613.1 | 5/5 | 5/5 | 0.26 | 3600 | 5/5 | 0/5 |
| **1701** | 60.00 | 3205.8 | 2/5 | 2/5 | 0.67 | 3044.7 | 5/5 | 1/5 | 0.00 | 1666.4 | 5/5 | 5/5 | 0.00 | 938.1 | 5/5 | 5/5 | 17.55 | 3600 | 4/5 | 0/5 |
| **1901** | 77.36 | 3390.5 | 1/5 | 1/5 | 0.00 | 1867.5 | 5/5 | 5/5 | 0.00 | 1959.3 | 5/5 | 5/5 | 0.00 | 910.1 | 5/5 | 5/5 | 34.88 | 3600 | 3/5 | 0/5 |
| **2101** | 80.00 | 3587.2 | 1/5 | 1/5 | 0.17 | 2729.2 | 5/5 | 3/5 | 0.00 | 2615.1 | 5/5 | 5/5 | 0.00 | 1132.1 | 5/5 | 5/5 | 0.30 | 3600 | 5/5 | 0/5 |
| **2301** | 100 | 3600 | 0/5 | 0/5 | 34.72 | 2643.5 | 3/5 | 3/5 | 0.01 | 2476.0 | 5/5 | 3/5 | 0.00 | 800.0 | 5/5 | 5/5 | 34.87 | 3600 | 3/5 | 0/5 |
| **2601** | 100 | 3600 | 0/5 | 0/5 | 0.16 | 2551.5 | 5/5 | 4/5 | 0.00 | 2705.5 | 5/5 | 5/5 | 0.00 | 1209.0 | 5/5 | 5/5 | 1.17 | 3404.6 | 5/5 | 1/5 |
| **2901** | 100 | 3600 | 0/5 | 0/5 | 17.78 | 3172.6 | 4/5 | 2/5 | 20.43 | 3356.6 | 4/5 | 2/5 | 0.00 | 1441.3 | 5/5 | 5/5 | 17.62 | 3600 | 4/5 | 0/5 |
| **3401** | 100 | 3600 | 0/5 | 0/5 | 0.75 | 3011.8 | 5/5 | 3/5 | 20.75 | 3179.9 | 4/5 | 2/5 | 0.00 | 2041.7 | 5/5 | 5/5 | 17.73 | 3600 | 4/5 | 0/5 |
| **4501** | 100 | 3600 | 0/5 | 0/5 | 100 | 3600 | 0/5 | 0/5 | 100 | 3600 | 0/5 | 0/5 | 34.71 | 3316.9 | 3/5 | 3/5 | 86.78 | 3600 | 0/5 | 0/5 |
| **Avg/Sum** | **48.45** | **2454.9** | **46/90** | **43/90** | **9.90** | **1819.8** | **81/90** | **61/90** | **7.85** | **1720.1** | **83/90** | **69/90** | **1.93** | **1021.6** | **88/90** | **83/90** | **12.06** | **2736.8** | **78/90** | **25/90** |

Table 4: Detailed computational results (GAP %, CPU in seconds)

| | Gurobi | | | | BP Algorithm 1 | | | | BP Algorithm 2 | | | | BP Algorithm 3 | | | | Impr. Benders Algorithm 2 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| INST. | LB | UB | GAP | CPU | LB | UB | GAP | CPU | LB | UB | GAP | CPU | LB | UB | GAP | CPU | LB | UB | GAP | CPU |
| **22-1** | 236.550 | 236.550 | 0.00 | 34.0 | 236.550 | 236.550 | 0.00 | 35.6 | 236.550 | 236.550 | 0.00 | 36.8 | 236.550 | 236.550 | 0.00 | 1875.4 | 236.550 | 236.572 | 0.00 | 130.8 |
| **22-2** | 231.148 | 231.148 | 0.00 | 16.3 | 231.148 | 231.148 | 0.00 | 32.4 | 231.148 | 231.148 | 0.00 | 35.3 | 231.148 | 231.148 | 0.00 | 30.5 | 231.148 | 231.152 | 0.00 | 129.2 |
| **22-3** | 232.605 | 232.606 | 0.00 | 15.4 | 232.605 | 232.622 | 0.00 | 40.8 | 232.605 | 232.627 | 0.00 | 51.0 | 232.605 | 232.626 | 0.00 | 24.9 | 232.605 | 232.626 | 0.00 | 94.0 |
| **22-4** | 233.700 | 233.702 | 0.00 | 26.5 | 233.700 | 233.700 | 0.00 | 177.0 | 233.700 | 233.700 | 0.00 | 60.7 | 233.700 | 233.700 | 0.00 | 23.0 | 233.700 | 233.709 | 0.00 | 69.7 |
| **22-5** | 234.742 | 234.760 | 0.00 | 28.6 | 234.742 | 234.751 | 0.00 | 17.2 | 234.742 | 234.751 | 0.00 | 19.5 | 234.742 | 235.107 | 0.16 | 3600 | 234.742 | 234.753 | 0.00 | 28.4 |
| **44-1** | 232.912 | 232.913 | 0.00 | 14.0 | 232.912 | 232.916 | 0.00 | 37.5 | 232.912 | 232.916 | 0.00 | 48.4 | 232.912 | 232.912 | 0.00 | 32.2 | 232.912 | 232.929 | 0.00 | 172.1 |
| **44-2** | 236.830 | 236.847 | 0.00 | 38.4 | 236.830 | 236.830 | 0.00 | 39.2 | 236.830 | 236.830 | 0.00 | 50.0 | 236.830 | 236.830 | 0.00 | 47.4 | 236.830 | 236.840 | 0.00 | 144.0 |
| **44-3** | 232.903 | 232.903 | 0.00 | 19.6 | 232.903 | 232.903 | 0.00 | 34.8 | 232.903 | 232.903 | 0.00 | 38.8 | 232.903 | 232.903 | 0.00 | 29.5 | 232.903 | 232.911 | 0.00 | 61.1 |
| **44-4** | 236.838 | 236.855 | 0.00 | 19.4 | 236.838 | 236.838 | 0.00 | 143.6 | 236.838 | 236.838 | 0.00 | 175.0 | 236.838 | 236.852 | 0.00 | 45.7 | 236.838 | 236.846 | 0.00 | 91.7 |
| **44-5** | 236.429 | 236.429 | 0.00 | 24.7 | 236.429 | 236.429 | 0.00 | 45.5 | 236.429 | 236.429 | 0.00 | 60.8 | 236.429 | 236.429 | 0.00 | 28.4 | 236.429 | 236.432 | 0.00 | 221.3 |
| **66-1** | 236.830 | 236.830 | 0.00 | 29.0 | 236.830 | 236.831 | 0.00 | 33.4 | 236.830 | 236.831 | 0.00 | 41.4 | 236.830 | 236.830 | 0.00 | 53.1 | 236.830 | 236.831 | 0.00 | 666.6 |
| **66-2** | 237.597 | 237.614 | 0.00 | 45.2 | 237.597 | 237.634 | 0.02 | 3600 | 237.597 | 237.631 | 0.01 | 3600 | 237.597 | 237.688 | 0.04 | 3600 | 237.597 | 237.617 | 0.00 | 142.7 |
| **66-3** | 236.451 | 236.451 | 0.00 | 13.5 | 236.451 | 236.451 | 0.00 | 47.9 | 236.451 | 236.456 | 0.00 | 61.3 | 236.451 | 236.451 | 0.00 | 50.3 | 236.451 | 236.465 | 0.00 | 41.7 |
| **66-4** | 234.940 | 234.956 | 0.00 | 81.1 | 234.940 | 234.970 | 0.01 | 3600 | 234.940 | 234.970 | 0.01 | 3600 | 234.940 | 234.940 | 0.00 | 2748.5 | 234.940 | 236.322 | 0.58 | 3600 |
| **66-5** | 237.732 | 237.874 | 0.06 | 3600 | 237.730 | 242.330 | 1.90 | 3600 | 237.730 | 237.993 | 0.11 | 3600 | 237.730 | 237.949 | 0.09 | 3600 | 237.730 | 238.228 | 0.21 | 3600 |
| **88-1** | 236.830 | 236.830 | 0.00 | 46.4 | 236.830 | 236.834 | 0.00 | 89.3 | 236.830 | 236.837 | 0.00 | 110.0 | 236.830 | 236.830 | 0.00 | 35.7 | 236.830 | 236.845 | 0.00 | 157.1 |
| **88-2** | 237.050 | 237.067 | 0.00 | 43.9 | 237.050 | 237.050 | 0.00 | 89.1 | 237.050 | 237.050 | 0.00 | 103.8 | 237.050 | 237.050 | 0.00 | 60.7 | 237.050 | 237.058 | 0.00 | 207.3 |
| **88-3** | 236.645 | 236.647 | 0.00 | 75.2 | 236.645 | 236.645 | 0.00 | 61.9 | 236.645 | 236.656 | 0.00 | 86.1 | 236.645 | 236.645 | 0.00 | 41.0 | 236.645 | 236.655 | 0.00 | 52.4 |
| **88-4** | 237.048 | 237.048 | 0.00 | 158.8 | 237.048 | 237.048 | 0.00 | 46.8 | 237.048 | 237.048 | 0.00 | 46.9 | 237.048 | 237.048 | 0.00 | 42.4 | 237.048 | 237.055 | 0.00 | 199.3 |
| **88-5** | 235.866 | 235.866 | 0.00 | 71.9 | 235.866 | 235.867 | 0.00 | 33.8 | 235.866 | 235.867 | 0.00 | 33.6 | 235.866 | 235.866 | 0.00 | 44.7 | 235.866 | 235.867 | 0.00 | 36.5 |
| **220-1** | 236.864 | 236.874 | 0.00 | 335.5 | 236.864 | 236.864 | 0.00 | 81.2 | 236.864 | 236.864 | 0.00 | 81.1 | 236.864 | 236.864 | 0.00 | 59.9 | 236.864 | 236.951 | 0.04 | 3600 |
| **220-2** | 237.067 | 237.073 | 0.00 | 550.1 | 237.067 | 237.067 | 0.00 | 83.0 | 237.067 | 237.067 | 0.00 | 82.9 | 237.067 | 237.067 | 0.00 | 145.7 | 237.067 | 237.090 | 0.00 | 184.5 |
| **220-3** | 237.924 | 237.924 | 0.00 | 536.8 | 237.924 | 237.924 | 0.00 | 160.0 | 237.924 | 237.924 | 0.00 | 210.2 | 237.924 | 237.924 | 0.00 | 81.0 | 237.924 | 237.935 | 0.00 | 2267.1 |
| **220-4** | 237.028 | 237.028 | 0.00 | 1146.3 | 237.028 | 237.028 | 0.00 | 80.3 | 237.028 | 237.028 | 0.00 | 80.4 | 237.028 | 237.028 | 0.00 | 75.3 | 237.028 | 237.049 | 0.00 | 267.0 |
| **220-5** | 237.872 | 237.898 | 0.01 | 3600 | 237.870 | N/A | N/A** | 3600 | 237.870 | 238.001 | 0.06 | 3600 | 237.870 | 237.926 | 0.02 | 3600 | 237.870 | 238.105 | 0.10 | 3600 |
| **660-1** | 237.666 | 237.678 | 0.00 | 2341.4 | 237.666 | 237.701 | 0.01 | 1442.2 | 237.666 | 237.701 | 0.01 | 3600 | 237.666 | 237.666 | 0.00 | 267.3 | 237.666 | 237.683 | 0.00 | 1359.0 |
| **660-2** | 237.847 | 237.866 | 0.00 | 1842.8 | 237.847 | 241.372 | 1.46 | 3600 | 237.847 | 237.900 | 0.02 | 3600 | 237.847 | 237.847 | 0.00 | 135.8 | 237.847 | 238.532 | 0.29 | 3600 |
| **660-3** | 237.349 | 237.349 | 0.00 | 2262.6 | 237.349 | 237.349 | 0.00 | 462.3 | 237.349 | 237.349 | 0.00 | 462.4 | 237.349 | 237.349 | 0.00 | 139.6 | 237.349 | 262.662 | 9.64 | 3600 |
| **660-4** | 238.339 | 238.345 | 0.00 | 1585.2 | 238.339 | 249.641 | 4.53 | 3600 | 238.339 | 238.361 | 0.00 | 753.4 | 238.339 | 238.339 | 0.00 | 196.4 | 238.339 | 238.409 | 0.03 | 3600 |
| **660-5** | 237.869 | 237.869 | 0.00 | 1539.9 | 237.869 | 237.869 | 0.00 | 382.4 | 237.869 | 237.869 | 0.00 | 382.0 | 237.869 | 237.869 | 0.00 | 374.1 | 237.869 | 238.410 | 0.23 | 3600 |
| **880-1** | 238.177 | 238.190 | 0.00 | 2457.7 | 238.177 | 238.179 | 0.00 | 869.1 | 238.177 | 238.186 | 0.00 | 1011.2 | 238.177 | 238.177 | 0.00 | 174.0 | 238.177 | 279.394 | 14.75 | 3600 |
| **880-2** | N/A | N/A | N/A* | 3600 | 237.412 | 237.412 | 0.00 | 2383.6 | 237.412 | 237.412 | 0.00 | 723.1 | 237.412 | 237.412 | 0.00 | 199.8 | 237.412 | 238.093 | 0.29 | 3600 |
| **880-3** | 238.357 | 238.385 | 0.01 | 3600 | 238.357 | 238.358 | 0.00 | 433.6 | 238.357 | 238.358 | 0.00 | 561.1 | 238.357 | 238.358 | 0.00 | 210.9 | 238.357 | 238.806 | 0.19 | 3600 |
| **880-4** | 237.214 | 237.238 | 0.00 | 2551.7 | 237.214 | 237.214 | 0.00 | 388.3 | 237.214 | 237.214 | 0.00 | 388.7 | 237.214 | 237.214 | 0.00 | 202.3 | 237.214 | 237.227 | 0.00 | 1069.2 |

Table 4: Detailed computational results (GAP %, CPU in seconds)

| | Gurobi | | | | BP Algorithm 1 | | | | BP Algorithm 2 | | | | BP Algorithm 3 | | | | Impr. Benders Algorithm 2 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| INST. | LB | UB | GAP | CPU | LB | UB | GAP | CPU | LB | UB | GAP | CPU | LB | UB | GAP | CPU | LB | UB | GAP | CPU |
| **880-5** | 238.006 | 238.022 | 0.00 | 2449.0 | 238.006 | 238.017 | 0.00 | 564.5 | 238.006 | 238.017 | 0.00 | 563.4 | 238.006 | 238.006 | 0.00 | 148.4 | 238.006 | 238.302 | 0.12 | 3600 |
| **1100-1** | 238.082 | 238.101 | 0.00 | 2290.0 | 238.082 | 306.808 | 22.40 | 3600 | 238.082 | 238.082 | 0.00 | 1132.6 | 238.082 | 238.082 | 0.00 | 286.7 | 238.082 | 239.057 | 0.41 | 3600 |
| **1100-2** | 237.714 | 237.717 | 0.00 | 1599.2 | 237.714 | 237.714 | 0.00 | 1123.2 | 237.714 | 237.714 | 0.00 | 952.6 | 237.714 | 237.714 | 0.00 | 604.3 | 237.714 | 237.730 | 0.00 | 1897.2 |
| **1100-3** | 237.165 | 237.170 | 0.00 | 2623.5 | 237.165 | 237.165 | 0.00 | 851.3 | 237.165 | 237.165 | 0.00 | 640.6 | 237.165 | 237.165 | 0.00 | 431.4 | 237.165 | 237.493 | 0.14 | 3600 |
| **1100-4** | 237.321 | 237.321 | 0.00 | 2795.7 | 237.321 | 237.321 | 0.00 | 555.4 | 237.321 | 237.321 | 0.00 | 551.5 | 237.321 | 237.321 | 0.00 | 344.8 | 237.321 | 237.447 | 0.05 | 3600 |
| **1100-5** | 237.801 | 237.802 | 0.00 | 3374.6 | 237.801 | 237.801 | 0.00 | 633.7 | 237.801 | 237.801 | 0.00 | 843.3 | 237.801 | 237.801 | 0.00 | 242.1 | 237.801 | 240.459 | 1.11 | 3600 |
| **1301-1** | 237.785 | 237.796 | 0.00 | 1964.5 | 237.785 | 240.423 | 1.10 | 3600 | 237.785 | 237.824 | 0.02 | 3600 | 237.785 | 237.785 | 0.00 | 1159.8 | 237.785 | 239.284 | 0.63 | 3600 |
| **1301-2** | N/A | N/A | N/A* | 3600 | 238.405 | 242.142 | 1.54 | 3600 | 238.405 | 238.437 | 0.01 | 3600 | 238.405 | 238.430 | 0.01 | 3600 | 238.405 | 238.792 | 0.16 | 3600 |
| **1301-3** | 237.901 | 237.917 | 0.00 | 2700.2 | 237.901 | 237.916 | 0.00 | 878.2 | 237.901 | 237.903 | 0.00 | 1040.2 | 237.901 | 237.903 | 0.00 | 400.9 | 237.901 | 238.464 | 0.24 | 3600 |
| **1301-4** | 238.138 | N/A | N/A** | 3600 | 238.137 | 238.151 | 0.00 | 629.3 | 238.137 | 238.151 | 0.00 | 897.5 | 238.137 | 238.151 | 0.00 | 607.7 | 238.137 | 238.534 | 0.17 | 3600 |
| **1301-5** | 237.386 | 237.386 | 0.00 | 2676.3 | 237.386 | 237.386 | 0.00 | 829.1 | 237.386 | 237.386 | 0.00 | 1038.5 | 237.386 | 237.386 | 0.00 | 235.4 | 237.386 | 237.654 | 0.11 | 3600 |
| **1501-1** | N/A | N/A | N/A* | 3600 | 237.770 | 237.777 | 0.00 | 938.1 | 237.770 | 237.777 | 0.00 | 839.4 | 237.770 | 237.770 | 0.00 | 736.5 | 237.770 | 238.253 | 0.20 | 3600 |
| **1501-2** | N/A | N/A | N/A* | 3600 | 237.459 | 237.459 | 0.00 | 715.9 | 237.459 | 237.459 | 0.00 | 1161.3 | 237.459 | 237.459 | 0.00 | 472.9 | 237.459 | 238.068 | 0.26 | 3600 |
| **1501-3** | N/A | N/A | N/A* | 3600 | 237.925 | 237.925 | 0.00 | 1325.6 | 237.925 | 237.925 | 0.00 | 1446.2 | 237.925 | 237.925 | 0.00 | 438.4 | 237.925 | 238.263 | 0.14 | 3600 |
| **1501-4** | N/A | N/A | N/A* | 3600 | 237.925 | 237.925 | 0.00 | 701.2 | 237.925 | 237.925 | 0.00 | 636.3 | 237.925 | 237.925 | 0.00 | 751.4 | 237.925 | 238.347 | 0.18 | 3600 |
| **1501-5** | 237.665 | N/A | N/A** | 3600 | 237.665 | 237.665 | 0.00 | 764.2 | 237.665 | 237.665 | 0.00 | 677.3 | 237.665 | 237.665 | 0.00 | 666.0 | 237.665 | 238.894 | 0.51 | 3600 |
| **1701-1** | N/A | N/A | N/A* | 3600 | 238.008 | 241.879 | 1.60 | 3600 | 238.008 | 238.025 | 0.00 | 2815.2 | 238.008 | 238.008 | 0.00 | 604.1 | 238.008 | 238.239 | 0.10 | 3600 |
| **1701-2** | N/A | N/A | N/A* | 3600 | 237.823 | 237.842 | 0.00 | 823.3 | 237.823 | 237.842 | 0.00 | 1335.5 | 237.823 | 237.823 | 0.00 | 814.2 | 237.823 | N/A | N/A** | 3600 |
| **1701-3** | 238.105 | 238.105 | 0.00 | 2348.4 | 238.105 | 239.348 | 0.52 | 3600 | 238.105 | 238.105 | 0.00 | 1142.3 | 238.105 | 238.105 | 0.00 | 732.2 | 238.105 | 238.239 | 0.06 | 3600 |
| **1701-4** | 237.896 | 237.899 | 0.00 | 2880.3 | 237.896 | 239.334 | 0.60 | 3600 | 237.896 | 237.896 | 0.00 | 1246.2 | 237.896 | 237.896 | 0.00 | 1096.1 | 237.896 | 238.715 | 0.34 | 3600 |
| **1701-5** | N/A | N/A | N/A* | 3600 | 237.677 | 239.182 | 0.63 | 3600 | 237.677 | 237.677 | 0.00 | 1792.8 | 237.677 | 237.677 | 0.00 | 1443.9 | 237.677 | 238.823 | 0.48 | 3600 |
| **1901-1** | N/A | N/A | N/A* | 3600 | 237.693 | 237.706 | 0.00 | 1522.6 | 237.693 | 237.706 | 0.00 | 2632.7 | 237.693 | 237.701 | 0.00 | 1242.6 | 237.693 | N/A | N/A** | 3600 |
| **1901-2** | 237.709 | N/A | N/A** | 3600 | 237.709 | 237.709 | 0.00 | 890.5 | 237.709 | 237.709 | 0.00 | 888.5 | 237.709 | 237.709 | 0.00 | 599.6 | 237.709 | 239.087 | 0.58 | 3600 |
| **1901-3** | 237.829 | 237.830 | 0.00 | 2552.3 | 237.829 | 237.836 | 0.00 | 2022.6 | 237.829 | 237.836 | 0.00 | 2635.2 | 237.829 | 237.829 | 0.00 | 1265.9 | 237.829 | 238.211 | 0.16 | 3600 |
| **1901-4** | N/A | N/A | N/A* | 3600 | 237.662 | 237.678 | 0.00 | 2577.5 | 237.662 | 237.678 | 0.00 | 1938.8 | 237.662 | 237.662 | 0.00 | 712.0 | 237.662 | N/A | N/A** | 3600 |
| **1901-5** | N/A | N/A | N/A* | 3600 | 238.216 | 238.216 | 0.00 | 2324.4 | 238.216 | 238.216 | 0.00 | 1701.1 | 238.216 | 238.216 | 0.00 | 730.5 | 238.216 | 238.360 | 0.06 | 3600 |
| **2101-1** | N/A | N/A | N/A* | 3600 | 237.786 | 237.787 | 0.00 | 3475.3 | 237.786 | 237.787 | 0.00 | 1846.6 | 237.786 | 237.787 | 0.00 | 1150.4 | 237.786 | 238.496 | 0.30 | 3600 |
| **2101-2** | N/A | N/A | N/A* | 3600 | 237.622 | 237.622 | 0.00 | 1763.4 | 237.622 | 237.622 | 0.00 | 2507.2 | 237.622 | 237.622 | 0.00 | 944.0 | 237.622 | 238.534 | 0.38 | 3600 |
| **2101-3** | N/A | N/A | N/A* | 3600 | 237.770 | 237.775 | 0.00 | 1207.4 | 237.770 | 237.775 | 0.00 | 2383.0 | 237.770 | 237.770 | 0.00 | 1069.2 | 237.770 | 239.031 | 0.53 | 3600 |
| **2101-4** | N/A | N/A | N/A* | 3600 | 237.812 | 239.124 | 0.55 | 3600 | 237.812 | 237.828 | 0.00 | 2739.7 | 237.812 | 238.812 | 0.00 | 1238.4 | 237.812 | 238.371 | 0.23 | 3600 |
| **2101-5** | 237.858 | 237.858 | 0.00 | 3536.2 | 237.858 | 238.544 | 0.29 | 3600 | 237.858 | 237.860 | 0.00 | 3599 | 237.858 | 237.860 | 0.00 | 1258.3 | 237.858 | 237.969 | 0.05 | 3600 |
| **2301-1** | N/A | N/A | N/A* | 3600 | 237.970 | N/A | N/A** | 3600 | 237.970 | 237.985 | 0.00 | 2818.1 | 237.970 | 237.970 | 0.00 | 1100.1 | 237.970 | N/A | N/A** | 3600 |
| **2301-2** | N/A | N/A | N/A* | 3600 | 238.049 | 238.053 | 0.00 | 2814.4 | 238.049 | 238.114 | 0.03 | 3600 | 238.049 | 238.049 | 0.00 | 658.0 | 238.049 | N/A | N/A** | 3600 |
| **2301-3** | N/A | N/A | N/A* | 3600 | 237.679 | N/A | N/A** | 3600 | 237.679 | 237.716 | 0.02 | 3600 | 237.679 | 237.679 | 0.00 | 770.5 | 237.679 | 238.814 | 0.48 | 3600 |

Table 4: Detailed computational results (GAP %, CPU in seconds)

| | Gurobi | | | | BP Algorithm 1 | | | | BP Algorithm 2 | | | | BP Algorithm 3 | | | | Impr. Benders Algorithm 2 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| INST. | LB | UB | GAP | CPU | LB | UB | GAP | CPU | LB | UB | GAP | CPU | LB | UB | GAP | CPU | LB | UB | GAP | CPU |
| **2301-4** | N/A | N/A | N/A* | 3600 | 237.593 | 237.593 | 0.00 | 2161.4 | 237.593 | 237.593 | 0.00 | 1323.2 | 237.593 | 237.593 | 0.00 | 660.2 | 237.593 | 238.072 | 0.20 | 3600 |
| **2301-5** | N/A | N/A | N/A* | 3600 | 238.164 | 238.164 | 0.00 | 1041.9 | 238.164 | 238.164 | 0.00 | 1038.9 | 238.164 | 238.164 | 0.00 | 811.1 | 238.164 | 238.471 | 0.13 | 3600 |
| **2601-1** | N/A | N/A | N/A* | 3600 | 237.826 | 239.782 | 0.82 | 3600 | 237.826 | 237.827 | 0.00 | 3175.5 | 237.826 | 237.827 | 0.00 | 1124.2 | 237.826 | 238.532 | 0.30 | 3600 |
| **2601-2** | N/A | N/A | N/A* | 3600 | 238.026 | 238.026 | 0.00 | 1695.7 | 238.026 | 238.026 | 0.00 | 1789.2 | 238.026 | 238.026 | 0.00 | 1463.3 | 238.026 | 238.797 | 0.32 | 3600 |
| **2601-3** | N/A | N/A | N/A* | 3600 | 237.783 | 237.783 | 0.00 | 3104.5 | 237.783 | 237.783 | 0.00 | 2498.2 | 237.783 | 237.783 | 0.00 | 1027.2 | 237.783 | 237.803 | 0.00 | 2623.2 |
| **2601-4** | N/A | N/A | N/A* | 3600 | 237.980 | 237.980 | 0.00 | 1414.6 | 237.980 | 237.980 | 0.00 | 2775.9 | 237.980 | 237.980 | 0.00 | 1020.6 | 237.980 | 248.841 | 4.37 | 3600 |
| **2601-5** | N/A | N/A | N/A* | 3600 | 237.942 | 237.942 | 0.00 | 2942.8 | 237.942 | 237.942 | 0.00 | 3288.7 | 237.942 | 237.942 | 0.00 | 1409.6 | 237.942 | 239.973 | 0.85 | 3600 |
| **2901-1** | N/A | N/A | N/A* | 3600 | 237.766 | 241.732 | 1.64 | 3600 | 237.766 | 241.732 | 1.64 | 3600 | 237.766 | 237.766 | 0.00 | 1319.4 | 237.766 | 238.497 | 0.31 | 3600 |
| **2901-2** | N/A | N/A | N/A* | 3600 | 238.103 | 238.103 | 0.00 | 3301.2 | 238.103 | 238.103 | 0.00 | 2699.7 | 238.103 | 238.103 | 0.00 | 1212.9 | 238.103 | 238.829 | 0.30 | 3600 |
| **2901-3** | N/A | N/A | N/A* | 3600 | 237.758 | 237.758 | 0.00 | 1762.0 | 237.758 | 237.758 | 0.00 | 3283.4 | 237.758 | 237.758 | 0.00 | 1547.2 | 237.758 | 238.662 | 0.38 | 3600 |
| **2901-4** | N/A | N/A | N/A* | 3600 | 237.959 | N/A | N/A** | 3600 | 237.959 | N/A | N/A** | 3600 | 237.959 | 237.959 | 0.00 | 1500.1 | 237.959 | N/A | N/A** | 3600 |
| **2901-5** | N/A | N/A | N/A* | 3600 | 237.819 | 238.980 | 0.49 | 3600 | 237.819 | 238.980 | 0.49 | 3600 | 237.819 | 237.823 | 0.00 | 1626.7 | 237.819 | 238.570 | 0.31 | 3600 |
| **3401-1** | N/A | N/A | N/A* | 3600 | 237.703 | 244.550 | 2.80 | 3600 | 237.703 | 244.550 | 2.80 | 3600 | 237.703 | 237.703 | 0.00 | 1837.2 | 237.703 | 238.801 | 0.46 | 3600 |
| **3401-2** | N/A | N/A | N/A* | 3600 | 238.198 | 238.198 | 0.00 | 2432.2 | 238.198 | 238.198 | 0.00 | 2433.3 | 238.198 | 238.198 | 0.00 | 2479.5 | 238.198 | 239.270 | 0.45 | 3600 |
| **3401-3** | N/A | N/A | N/A* | 3600 | 238.048 | 238.048 | 0.00 | 2633.1 | 238.048 | 238.048 | 0.00 | 2666.2 | 238.048 | 238.048 | 0.00 | 1920.0 | 238.048 | 238.767 | 0.30 | 3600 |
| **3401-4** | N/A | N/A | N/A* | 3600 | 237.789 | 237.794 | 0.00 | 2793.6 | 237.789 | N/A | N/A** | 3600 | 237.789 | 237.789 | 0.00 | 1903.9 | 237.789 | N/A | N/A** | 3600 |
| **3401-5** | N/A | N/A | N/A* | 3600 | 237.969 | 240.198 | 0.93 | 3600 | 237.969 | 240.198 | 0.93 | 3600 | 237.969 | 237.969 | 0.00 | 2067.7 | 237.969 | 239.490 | 0.64 | 3600 |
| **4501-1** | N/A | N/A | N/A* | 3600 | N/A | N/A | N/A* | 3600 | N/A | N/A | N/A* | 3600 | 237.861 | 237.861 | 0.00 | 3206.4 | 237.861 | N/A | N/A** | 3600 |
| **4501-2** | N/A | N/A | N/A* | 3600 | N/A | N/A | N/A* | 3600 | N/A | N/A | N/A* | 3600 | 238.001 | 238.001 | 0.00 | 3020.4 | 238.001 | N/A | N/A** | 3600 |
| **4501-3** | N/A | N/A | N/A* | 3600 | N/A | N/A | N/A* | 3600 | 237.943 | N/A | N/A** | 3600 | 237.943 | N/A | N/A** | 3600 | 237.943 | N/A | N/A** | 3600 |
| **4501-4** | N/A | N/A | N/A* | 3600 | N/A | N/A | N/A* | 3600 | N/A | N/A | N/A* | 3600 | 237.888 | N/A | N/A** | 3600 | 237.888 | N/A | N/A** | 3600 |
| **4501-5** | N/A | N/A | N/A* | 3600 | N/A | N/A | N/A* | 3600 | N/A | N/A | N/A* | 3600 | 238.039 | 238.040 | 0.00 | 3157.7 | 238.039 | N/A | N/A** | 3600 |

Note: cells marked with * are accepted as 100%, and ** are calculated as 100(1800-LB)/1800% in Table 3

## 6. Conclusions

VMAT is known as the state-of-the-art delivery method in radiation therapy. However, finding high quality VMAT plans is a challenging issue. The most important reason is that radiation is delivered to the patient's body continuously and the MLC system shapes the beam while the gantry rotates around the patient. This property makes the control points interdependent, since the apertures at two adjacent control points must be compatible. As a result, the VMAT planning problem cannot be decomposed into independent small problems and must be considered as a whole in contrast with the preceding technology IMRT. This makes the problem harder to solve. We develop a MILP model for the VMAT planning problem in which all radiation prescription doses are hard constraints and the objective is to minimize total MUs of treatment. In our proposed branch-and-price (BP) algorithms we reformulate the VMAT treatment planning problem and introduce each feasible treatment row arc for each MLC row as a variable of the reformulated model. We solve the linear programming relaxation of the reformulated model using column generation at each node of the branch-and-bound tree. For each pricing subproblem, a network model is developed and solved using dynamic programming in polynomial time. Our BP implementation provides significantly better results compared to Gurobi solver and also Improved Benders Algorithm 2, which is the most efficient algorithm of our previous study (Dursun et al., 2019). Moreover, it is possible to solve pricing subproblems in parallel on multiple threads because they are independent from each other. Thus, it is possible to speed-up the BP Algorithms. However, it is possible to solve real size problems including only one target volume and one OAR with the current version of our algorithms. Even though our proposed BP algorithms significantly outperform other approaches, improving them further in order to make it possible to obtain applicable treatment plans with low MUs for clinical cases remains as a challenging research direction. As a future study, BP algorithms can be improved by combining them with some heuristic approaches proposed in the literature in order to obtain better upper bounds for the problem.

## Acknowledgments

This research has been partially supported by the Boğaziçi University Research Fund with grant number 11520-16A03D1 and by the Turkish Directorate of Strategy and Budget under the TAM Project number DPT2007K120610. The authors would like to thank very kindly to Prof. Hatice Bilge, Asst. Prof. Murat Okutan, Prof. Ethem Nezih Oral from Istanbul University Oncology Institute for their guidance and support.

## References


Akartunalı, K., Mak-Hau, V., Tran, T., 2015. A unified mixed-integer programming model for simultaneous fluence weight and aperture optimization in VMAT, Tomotherapy, and Cyberknife. Computers & Operations Research 56, 134–150.

Bertsimas, D., Cacchiani, V., Craft, D., Nohadani, O., 2013. A hybrid approach to beam angle optimization in intensity-modulated radiation therapy. Computers & Operations Research 40 (9), 2187–2197.

Boland, N., Hamacher, H. W., Lenzen, F., 2004. Minimizing beam-on time in cancer radiation treatment using multileaf collimators. Networks 43 (4), 226–240.

Bzdusek, K., Friberger, H., Eriksson, K., Hårdemark, B., Robinson, D., Kaus, M., 2009. Development and evaluation of an efficient approach to volumetric arc therapy planning. Medical Physics 36 (6), 2328–2339.

Cambazard, H., O'Mahony, E., O'Sullivan, B., 2012. A shortest path-based approach to the multileaf collimator sequencing problem. Discrete Applied Mathematics 160 (1), 81–99.

Cameron, C., 2005. Sweeping-window arc therapy: an implementation of rotational IMRT with automatic beam-weight calculation. Physics in Medicine and Biology 50 (18), 4317–4336.

Cao, D., Afghan, M. K., Ye, J., Chen, F., Shepard, D. M., 2009. A generalized inverse planning tool for volumetric-modulated arc therapy. Physics in Medicine and Biology 54 (21), 6725–6738.

Carlsson, F., 2008. Combining segment generation with direct step-and-shoot optimization in intensity-modulated radiation therapy. Medical Physics 35 (9), 3828–3838.

CCS Healthcare, 2018. Accessed: 2018-03-27.
URL `http://www.ccshealthcare.com/services/radiation-oncology/`

Craft, D., Bangert, M., Long, T., Papp, D., Unkelbach, J., 2014a. Shared data for intensity modulated radiation therapy (IMRT) optimization research: the CORT dataset. GigaScience 3 (1), 37.

Craft, D., Bangert, M., Long, T., Papp, D., Unkelbach, J., 2014b. Supporting material for: "Shared data for IMRT optimization research: the CORT dataset".

Craft, D., McQuaid, D., Wala, J., Chen, W., Salari, E., Bortfeld, T., 2012. Multicriteria VMAT optimization. Medical Physics 39 (2), 686–696.

Desaulniers, G., Desrosiers, J., Solomon, M. M., 2006. Column Generation. Vol. 5. Springer Science & Business Media.

Dursun, P., Taşkın, Z. C., Altınel, İ. K., 2016. Mathematical models for optimal volumetric modulated arc therapy (VMAT) treatment planning. In: Procedia Computer Science (Proceedings of International Conference on Health and Social Care Information Systems and Technologies, HCist, Porto). Vol. 100. pp. 644–651.

Dursun, P., Taşkın, Z. C., Altınel, İ. K., 2019. The determination of optimal treatment plans for Volumetric Modulated Arc Therapy (VMAT). European Journal of Operational Research 272 (1), 372–388.

Earl, M., Shepard, D., Naqvi, S., Li, X., Yu, C., 2003. Inverse planning for intensity-modulated arc therapy using direct aperture optimization. Physics in Medicine and Biology 48 (8), 1075–1089.

Ehrgott, M., Güler, Ç., Hamacher, H. W., Shao, L., 2010. Mathematical optimization in intensity modulated radiation therapy. Annals of Operations Research 175 (1), 309–365.

Gören, M., Taşkın, Z. C., 2015. A column generation approach for evaluating delivery efficiencies of collimator technologies in IMRT treatment planning. Physics in Medicine and Biology 60 (5), 1989.

Gozbasi, H. O., 2010. Optimization approaches for planning external beam radiotherapy. Ph.D. thesis, Georgia Institute of Technology.

Gurobi, O., 2018. Gurobi optimizer reference manual version 8.0. Accessed: 2018-07-10.
URL `http://www.gurobi.com/documentation/8.0/refman.pdf`

Guta, B., 2003. Subgradient optimization methods in integer programming with an application to a radiation therapy problem. Ph.D. thesis, Teknishe Universitat Kaiserlautern, Kaiserlautern.

Hall, E. J., Wuu, C.-S., 2003. Radiation-induced second cancers: the impact of 3D-CRT and IMRT. International Journal of Radiation Oncology* Biology* Physics 56 (1), 83–88.

Held, M., Wolfe, P., Crowder, H. P., 1974. Validation of subgradient optimization. Mathematical Programming 6 (1), 62–88.

Lee, E. K., Fox, T., Crocker, I., 2003. Integer programming applied to intensity-modulated radiation therapy treatment planning. Annals of Operations Research 119 (1-4), 165–181.

Luan, S., Wang, C., Cao, D., Chen, D. Z., Shepard, D. M., Cedric, X. Y., 2008. Leaf-sequencing for intensity-modulated arc therapy using graph algorithms. Medical Physics 35 (1), 61–69.

Lübbecke, M. E., 2011. Column generation. Wiley Encyclopedia of Operations Research and Management Science.

Lübbecke, M. E., Desrosiers, J., 2005. Selected topics in column generation. Operations Research 53 (6), 1007–1023.

Mahnam, M., Gendreau, M., Lahrichi, N., Rousseau, L.-M., 2017. Simultaneous delivery time and aperture shape optimization for the volumetric-modulated arc therapy (VMAT) treatment planning problem. Physics in Medicine and Biology 62 (14), 5589–5611.

McCormick, G. P., 1976. Computability of global solutions to factorable nonconvex programs: Part I- Convex underestimating problems. Mathematical Programming 10 (1), 147–175.

Men, C., Romeijn, H. E., Jia, X., Jiang, S. B., 2010. Ultrafast treatment plan optimization for volumetric modulated arc therapy (VMAT). Medical Physics 37 (11), 5787–5791.

Men, C., Romeijn, H. E., Taşkın, Z. C., Dempsey, J. F., 2007. An exact approach to direct aperture optimization in IMRT treatment planning. Physics in Medicine and Biology 52 (24), 7333–7352.

Otto, K., 2008. Volumetric modulated arc therapy: IMRT in a single gantry arc. Medical Physics 35 (1), 310–317.

Palma, D., Vollans, E., James, K., Nakano, S., Moiseenko, V., Shaffer, R., McKenzie, M., Morris, J., Otto, K., 2008. Volumetric modulated arc therapy for delivery of prostate radiotherapy: comparison with intensity-modulated radiotherapy and three-dimensional conformal radiotherapy. International Journal of Radiation Oncology* Biology* Physics 72 (4), 996–1001.

Papp, D., Unkelbach, J., 2014. Direct leaf trajectory optimization for volumetric modulated arc therapy planning with sliding window delivery. Medical Physics 41 (1), 011701.

Peng, F., Jia, X., Gu, X., Epelman, M. A., Romeijn, H. E., Jiang, S. B., 2012. A new column-generation-based algorithm for VMAT treatment plan optimization. Physics in Medicine and Biology 57 (14), 4569–4588.

Peng, F., Jiang, S. B., Romeijn, H. E., Epelman, M. A., 2015. VMATc: VMAT with constant gantry speed and dose rate. Physics in Medicine and Biology 60 (7), 2955.

Preciado-Walters, F., Langer, M. P., Rardin, R. L., Thai, V., 2006. Column generation for IMRT cancer therapy optimization with implementable segments. Annals of Operations Research 148 (1), 65–79.

Python, 2015. Python 2.7.11 documentation. Accessed: 2018-03-27.
URL `https://docs.python.org/release/2.7.11/`

Rockafellar, R. T., Uryasev, S., et al., 2000. Optimization of conditional value-at-risk. Journal of Risk 2, 21–42.

Romeijn, H. E., Ahuja, R. K., Dempsey, J. F., Kumar, A., 2005. A column generation approach to radiation therapy treatment planning using aperture modulation. SIAM Journal on Optimization 15 (3), 838–862.

Romeijn, H. E., Ahuja, R. K., Dempsey, J. F., Kumar, A., 2006. A new linear programming approach to radiation therapy treatment planning problems. Operations Research 54 (2), 201–216.

Rosen, J. B., 1960. The gradient projection method for nonlinear programming. Part I. Linear constraints. Journal of the society for industrial and applied mathematics 8 (1), 181–217.

Salari, E., Unkelbach, J., 2013. A column-generation-based method for multi-criteria direct aperture optimization. Physics in Medicine and Biology 58 (3), 621–639.

Salari, E., Wala, J., Craft, D., 2012. Exploring trade-offs between VMAT dose quality and delivery efficiency using a network optimization approach. Physics in Medicine and Biology 57 (17), 5587–5600.

Savelsbergh, M., 1997. A branch-and-price algorithm for the generalized assignment problem. Operations Research 45 (6), 831–841.

Shepard, D., Cao, D., Afghan, M., Earl, M., 2007. An arc-sequencing algorithm for intensity modulated arc therapy. Medical Physics 34 (2), 464–470.

Shepard, D., Earl, M., Li, X., Naqvi, S., Yu, C., 2002. Direct aperture optimization: a turnkey solution for step-and-shoot IMRT. Medical Physics 29 (6), 1007–1018.

Song, J., Shi, Z., Sun, B., Shi, L., 2015. Treatment planning for volumetric-modulated arc therapy: Model and heuristic algorithms. IEEE Transactions on Automation Science and Engineering 12 (1), 116–126.

Taşkın, Z. C., 2019. VMAT Data Sets.
URL http://www.ie.boun.edu.tr/~taskin/data/vmat

Taşkın, Z. C., Smith, J. C., Romeijn, H. E., Dempsey, J. F., 2010. Optimal multileaf collimator leaf sequencing in IMRT treatment planning. Operations Research 58 (3), 674–690.

Teoh, M., Clark, C. H., Wood, K., Whitaker, S., Nisbet, A., 2011. Volumetric modulated arc therapy: a review of current literature and clinical use in practice. The British Journal of Radiology 84, 967–996.

Vanderbeck, F., 1994. Decomposition and column generation for integer programs. Ph.D. thesis, Université catholique de Louvain.

Varian, 2018. Accessed: 2018-03-27.
URL https://www.varian.com/about-varian/newsroom/image-gallery?title_field_value=linear+&field_image_category_tid=All

Wala, J., Salari, E., Chen, W., Craft, D., 2012. Optimal partial-arcs in VMAT treatment planning. Physics in Medicine and Biology 57 (18), 5861–5874.

Wang, C., Luan, S., Tang, G., Chen, D. Z., Earl, M. A., Cedric, X. Y., 2008. Arc-modulated radiation therapy (AMRT): a single-arc form of intensity-modulated arc therapy. Physics in Medicine and Biology 53 (22), 6291–6303.

Yu, C. X., 1995. Intensity-modulated arc therapy with dynamic multileaf collimation: an alternative to tomotherapy. Physics in Medicine and Biology 40 (9), 1435–1449.

Zhang, P., Happersett, L., Yang, Y., Yamada, Y., Mageras, G., Hunt, M., 2010. Optimization of collimator trajectory in volumetric modulated arc therapy: development and evaluation for paraspinal SBRT. International Journal of Radiation Oncology* Biology* Physics 77 (2), 591–599.