Minimizing Beam-On Time in Cancer Radiation Treatment Using Multileaf Collimators

Natashia Boland* Horst W. Hamacher† Frank Lenzen‡

January 4, 2002

Abstract

In this paper the modulation of intensity matrices arising in cancer radiation therapy using multileaf collimators (MLC) is investigated. It is shown that the problem is equivalent to decomposing a given integer matrix into a positive linear combination of $(0,1)$ matrices. These matrices, called *shape matrices*, must have the strict consecutive-1-property, together with another property derived from the technological restrictions of the MLC equipment. Various decompositions can be evaluated by their beam-on-time (time in which radiation is applied to the patient) or the treatment time (beam-on-time plus time for set-ups). We focus on the former, and develop a nonlinear mixed integer programming formulation of the problem. This formulation can be decomposed to yield a column generation formulation: a linear program with a large number of variables that can be priced out by solving a subproblem. We then develop a network model in which paths in the network correspond to feasible shape matrices. As a consequence, we deduce that the column generation subproblem can be solved as a shortest path problem, and so obtain our main theoretical result that the problem is solvable in polynomial time. Furthermore, we are able to develop two alternative models of the problem as side-constrained network flow formulations. Finally, a numerical comparison of our exact solutions with those of well-known heuristic methods shows that the beam-on time can be reduced by a considerable margin.

1 Introduction

In most parts of the world, cancer is one of the major causes of deaths. In order to fight this disease, radiation therapy is used very often, in particular, in cases where the tumor can be localized and metastases have not yet started to form. In such a situation, radiation is

*Department of Mathematics and Statistics, University of Melbourne, Australia
†Fachbereich Mathematik, Universität Kaiserslauten, Germany, hamacher@mathematik.uni-kl.de
‡Anderson Consulting

applied to destroy the tumor (target volume) while maintaining the functionality of organs which are close to the tumor (organs at risk) and thus subject to at least some of the radiation applied to the tumor.

In order to apply radiation from an external source, medical linear accelerators are used. They are part of a gantry which can be rotated about the patient, who is positioned and fixed on a couch (see Figure 1).

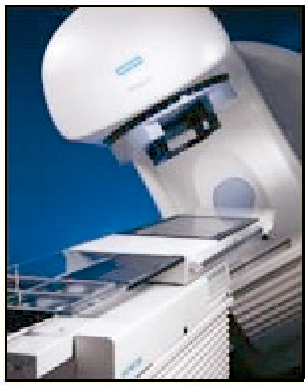


Figure 1: *A medical linear accelarerator with a beam head and a treatment couch*

In the design of the treatment plan information has to be collected and several decisions have to be made individually for each patent, all of which are of crucial importance for the quality of the radiation plan. (see Figure 2.)

(1) The location of the target volume and organs at risk.

(2) A discretization of the radiation beam head into **bixels**.

(3) A discretization of the target volume and risk organs into **voxels**.

(4) A set of positions at which the gantry stops in order to release radiation.

(5) A decision on the **intensity function**, i.e. the amount of radiation released at each stop and in each bixel.

(6) The modulation of the uniform radiation to achieve (5).

The location of target volume and organs at risk is done using computer tomography (CT). Correct three-dimensional images are assumed in all subsequent models. Research in this area is on updating images due to movement of the patient on the couch or due to the impact of previous radiation.

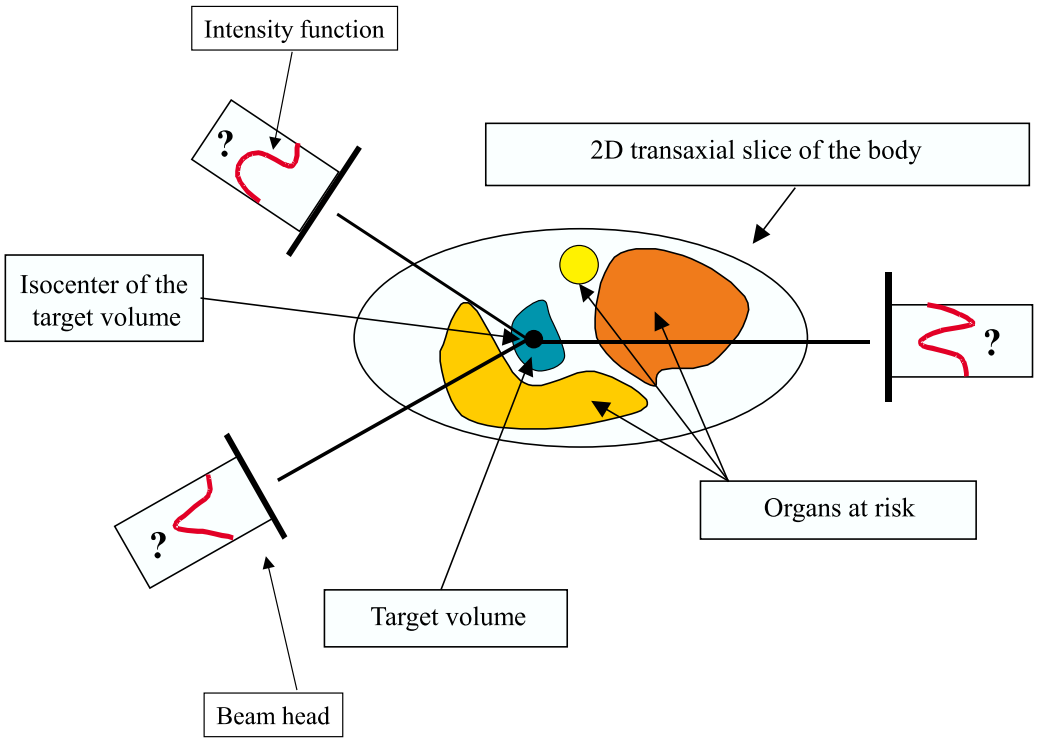


Figure 2: *Radiation planning problems: Stops of the gantry and the two-dimensional intensity functions have to be found. For this purpose each 3-dimensional organ at risk and the target volume are discretized into voxels and the 2-dimensional beam heads into bixels.*

The radiation head is assumed to be a rectangle which is partitioned into equidistant cells. This partitioning will be discussed in more detail in Section 2. Correspondingly, target volume and organs at risk are assumed to be partitioned into cubes. Again, all available models assume that this is done in an equidistant way. Better radiation plans may, however, be achieved by considering non-equidistant partitions, a topic which is currently under research. Radiation gantries are typically designed in such a way that they may stop at 36 positions, equally distributed on the gantry's moving circle around the patient. The number of stops is not fixed, but in the clinical practice a number between 3 and 7 stops is most common. The problem of finding the best positions for stops has been addressed in previous work, such as that on (continuous) inverse approaches to treatment planning, which we discuss further below. However this problem can also be modeled as a discrete location problem, an approach which is currently under investigation.

The amount of radiation released at each stop and in each bixel can be written as a system of linear equations

$$P\underline{x} = D$$

where $P = (p_{ij})$ is the bixel-voxel unit radiation matrix, i.e. p_{ij} is the amount of radiation

reaching voxel i if one unit of radiation is released at bixel j, and where $\underline{x} = (x_j)$ is the amount of time radiation is sent off at bixel j. Correspondingly, D is the dosage vector representing in each component D_i the radiation of each voxel i obtained as cumulative radiation from all bixels j. The dosage has to satisfy constraints, the most important of which are lower bounds in voxels of the target volume (to destroy the cancer) and upper bounds in voxels of the organs at risk (to maintain functionality). In general, these bounds make the system of linear equations inconsistent and mathematical programming methods are used to minimize the deviation from the bounds ([Bortfeld, 1995], [Burkard et al., 1995]), [Shepard et al., 1999]. For an interesting discussion of issues that arise in modeling the objectives of the treatment planning process, see the work of [Raphael, 1992]. Much previous work on radiation treatment planning has attempted to handle the underlying physics in more details, and has not necessarily assumed discretized models of the target volume and treatment space, nor simple linear relationships between treatment beam intensities and dosages in the target volume. Examples can be found in the work of [Kolmonen et al., 1998] and in [Tervo and Kolmonen, 2000], and references therein. Approaches such as these are based on inverse techniques, which typically lead to difficult nonlinear programming or optimal control problems. [Lee et al., 2000] adopt a conception of the problem similar to that we describe above, and give an integer programming approach. They show that in modest computational time, significant improvements can be obtained in the quality of the treatment plans over those developed by human experts. ([Hamacher and Küfer, 2001]) have recently used multicriteria approaches to tackle this problem. In recent work, [Wu and Zhu, 2001] also take a multicriteria approach, but use genetic algorithms to determine both the intensities and the importance factors of the different criteria. In this paper we will focus on the modulation of the uniform radiation to realize the radiation $\underline{x}$ in each stopping position.

The rest of the paper is organized as follows. In Section 2, the static, or "step-and-shoot", multileaf technology will be introduced and optimization problems resulting from the usage of this technology will be discussed. We focus primarily on the problem of minimizing the total radiation time at each stopping position. The following two sections, Sections 3 and 4 will contain two different models for tackling these problems. The first model is based on mixed integer programming, the second one on a shortest path/network flow formulation. The latter allows us to show that the problem of minimizing total treatment time can be solved in polynomial time, and that column generation approaches yield subproblems which take the form of shortest path problems. This insight is especially important, as the same column generation subproblem is likely to arise within column generation approaches to the overarching dosage problem. In Section 5, we discuss the improvements over existing heuristic methods that result from solving the problem exactly, using the side-constrained network flow model.

2 Modulating Radiation Using Multileaf Collimators

In this section we assume that (1) - (5) from Section 1 have been dealt with. In particular, we know for each stop of the gantry the intensity function represented as a two-dimensional array of the amount of time uniform radiation is released in each bixel, i.e. as intensity matrix $\mathcal{I}$. We assume in the following that $\mathcal{I}$ is an integer valued matrix. If we have chosen a discretization of the beam head into a 6×6 grid,

$$\mathcal{I} = \begin{pmatrix} 0 & 0 & 2 & 2 & 2 & 0 \\ 0 & 1 & 1 & 3 & 1 & 0 \\ 0 & 0 & 2 & 2 & 1 & 0 \\ 1 & 2 & 2 & 2 & 1 & 0 \\ 0 & 1 & 2 & 3 & 2 & 1 \\ 0 & 1 & 2 & 2 & 2 & 2 \end{pmatrix}$$

is such a possible intensity matrix.

In order to generate $\mathcal{I}$ the uniform radiation leaving the linear accelerator has to be modulated by inserting filters between radiation source and patient. The currently most common way to do this is by using metal filters and shaping them in such a way that the intensity matrix $\mathcal{I}$ is generated.

A more advanced way of modulation is achieved by using a multileaf collimator (MLC). Here, each row of $\mathcal{I}$ (often referred to as a **channel**) has an associated pair of leaves - a right leaf and a left leaf. If $\mathcal{I}$ has n columns the left leaf may be positioned in column $0, 1, \ldots, n$, and the right leaf can be placed in columns $1, \ldots, n, n+1$, where columns 0 and $n+1$ are notional columns used to represent the respective leaf's fully retracted position. Radiation can pass in between left and right leaf, so, if the left leaf is in position l and the right leaf is in position r, only the bixels in columns $l+1, \ldots, r-1$ of that channel will transmit radiation. Clearly we require $r > l$. Figure 3 shows an MLC.
Each choice of left/right leaves in all rows is characterized by a $(0,1)$ matrix in the following way. If the left and right leaf is positioned in column l and r, respectively, then the corresponding row has consecutive ones in entries $l+1, \ldots, r-1$ and zeros everywhere else. A $(0,1)$ matrix constructed in this way is called a **shape matrix**. Shape matrices and left/right leaf configurations are in one-to-many correspondence, since for shape matrices with zero rows, more than one left/right leaf configuration can be found.

If $S_1, \ldots, S_K$ are shape matrices and $\alpha_1, \ldots, \alpha_K$ is the time the linear accelerator is opened to release (uniform) radiation when leaf pairs are in the positions indicated by the corresponding shape matrix, an intensity of $\sum_{k=1}^{K} \alpha_k S_k$ is released. Since we know the intensity matrix, the **MLC problem** (basic version) is therefore defined as follows.

Given an integer matrix $\mathcal{I}$, find K, and shape matrices $S_1, \ldots, S_K$ and $\alpha_1, \ldots, \alpha_K > 0$ such

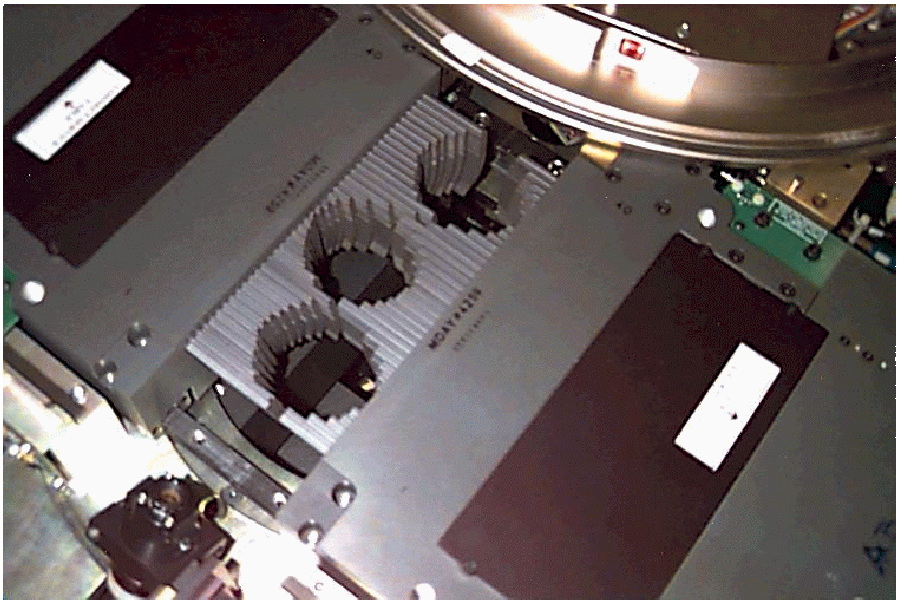


Figure 3: *Leaf pairs of an MLC. Radiation will pass through the openings.*

that

$$\sum_{k=1}^{K} \alpha_k \, S_k = \mathcal{I}.$$

Example 2.1 *Let*

$$\mathcal{I} = \begin{pmatrix} 0 & 0 & 4 & 4 & 3 & 0 \\ 0 & 1 & 1 & 6 & 3 & 0 \\ 0 & 0 & 3 & 4 & 1 & 0 \\ 1 & 3 & 4 & 4 & 3 & 0 \\ 0 & 2 & 3 & 6 & 4 & 3 \\ 0 & 1 & 3 & 3 & 4 & 4 \end{pmatrix}$$

Then $I = 3S_1 + 1S_2 + 2S_3$, *where*

$$S_1 = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}, S_2 = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$S_3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}$$

The leaf configurations corresponding to the shape matrices S_1, S_2 and S_3 are shown in Figure 4.

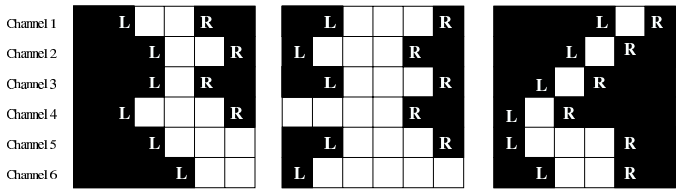


Figure 4: *Leaf configurations of shape matrices S_1, S_2, and S_3 of Example 2.1.*

It is obviously easy to represent any given intensity matrix $\mathcal{I}$ as a positive linear combination of shape matrices S_k. If $\mathcal{I} = (I_{ij})_{\substack{i=1,\ldots,m \\ j=1,\ldots,n}}$ (i.e. the discretization of the beam head is into m rows (or channels) and n columns) we could, for instance, choose $\mathcal{I} = \sum_{i=1}^{m} \sum_{j=1}^{n} I_{ij} S_{ij}$ where S_{ij} is a shape matrix with entries

$$S_{ij}(k,l) := \begin{cases} 1 & \text{if } k = i \text{ and } l = j \\ 0 & \text{otherwise} \end{cases}$$

i.e. a matrix which has just one non-zero entry at position (i,j). For this trivial decomposition of $\mathcal{I}$, the **beam-on-time**, i.e. the time at which the linear accelerator is sending off radiation is

$$\sum_{k=1}^{K} \alpha_k = \sum_{i=1}^{m} \sum_{j=1}^{n} I_{ij}$$

The next example shows that this decomposition is, in general, not a good one.

Example 2.2 *Let*

$$\mathcal{I} = \begin{pmatrix} 2 & 3 \\ 4 & 2 \end{pmatrix}.$$

The trivial decomposition described above yields

$$\mathcal{I} = 2 \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} + 3 \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} + 4 \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} + 2 \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$

with a beam-on time of $\sum_{k=1}^{4} \alpha_k = 2+3+4+2 = 11$.
Alternatively, $\mathcal{I}$ *can be decomposed into*

$$\mathcal{I} = 2\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} + 1\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} + 1\begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$$

giving a beam-on time of $\sum_{k=1}^{3} \alpha'_k = 2+1+1 = 4$.

Since beam-on time is related to the duration of the radiation treatment and since the latter time is to be minimized (in order to avoid discomfort to the patient, and his/her movement which would change the data of all optimization models in the radiation problem) we formulate an MLC optimization problem, the **MLC problem with minimal beam-on time**:

$$\begin{aligned} \min \quad & \sum_{t\in\mathcal{T}} \alpha_t \\ \text{subject to} \quad & \sum_{t\in\mathcal{T}} \alpha_t S_t = \mathcal{I} \\ & \alpha_t \geq 0. \end{aligned} \tag{1}$$

Here, $\mathcal{T}$ is the index set of all possible shape matrices. This is the problem which we will study in more detail in the subsequent sections. Before doing so let us mention additional models which are currently under investigation.

If we assume that the switch between two shape matrices will take a given fixed amount of time (**set-up time**), say T_c seconds, then the objective of the **MLC problem with minimal beam-on time and constant set-up time** is

$$\sum_{k=1}^{K} \alpha_k + (K-1)T_c. \tag{2}$$

When the set-up time is dominant, this objective is minimized by minimizing the number of shape matrices used, K.

In fact, it is likely that the set-up time between shape matrices depends on the form of these matrices. In the **MLC problem with minimal beam-on time and variable set-up time** we would thus consider the objective

$$\sum_{k=1}^{K} (\alpha_k + c(S_k, S_{k+1})) \tag{3}$$

where $c(S_k, S_{k+1})$ is the time it takes to change from S_k to S_{k+1} (and where $c(S_K, S_{K+1}) = 0$)).

A number of authors have considered the problems of minimizing the beam-on time plus constant set-up time, and of minimizing the number of shape matrices used. A very influential paper is that of [Xia and Verhey, 1998], who introduce a heuristic method in which all beam-on times are powers of two. Larger powers of two are "stripped off" first. They compare their method computationally with earlier methods of [Bortfeld et al., 1994] and [Galvin et al., 1993], on randomly generated test problems, and a small number of clinical data sets. Their method was clearly more effective than either of the others for minimizing the number of shape matrices used. This verdict was borne out by [Que, 1999], who also tested several variations of the Xia and Verhey heuristic, as well as the IMFAST algorithm, which is based on the work of [Siochi, 1999], and is implemented in the commercial system CORVUS. [Que, 1999] also used many more clinical data sets for his comparisons. Que's conclusions were that the Xia and Verhey heuristic outperformed the others on a statistical basis. However there were cases in which other algorithms performed better, and the clinical data sets, in particular, showed a lot of variation. In other work, [Yu et al., 1995a] present models and methods for realizing continuous 2D intensity functions by dynamic control of both the leaf velocities and the radiation intensity. [Tervo and Kolmonen, 2000] tackle the overarching treatment planning problem directly in terms of the leaf positions. Their work is more general than that of [Bortfeld et al., 1994, Galvin et al., 1993, Que, 1999, Xia and Verhey, 1998], and of our own, in the sense that they allow collimator leaves to take on any positions; we allow only n positions for each leaf in each channel. However they must, as a consequence, solve rather difficult nonlinear problems. Furthermore, their approach is implicitly dynamic, whereas we focus on the static case. Their idea of working directly with the leaf variables to solve the treatment planning problem is, however, intriguing, and we intend in future work to consider embedding the work we present here in a treatment planning problem, in a similar way.

Less work has been done which considers variable (sequence dependent) set-up time. The work of [Dai and Hu, 1999] is notable. In a two-phase approach, they firstly use the Xia and Verhey heuristic to determine the shape matrices and associated beam-on times, and then seek a sequence which minimizes the leaf-moving times between shape matrices using simulated annealing. However their paper gives very little detail on their method. How leaf-moving times between pairs of shape matrices are determined is not given, nor is any detail of their simulated annealing approach. The simulated annealing achieves roughly a 10% improvement in the total leaf-moving time, over the initial sequence output by the Xia and Verhey heuristic. (It should be noted that the main thrust of Dai and Hu's paper is to examine independent collimators, rather than multi-leaf collimators, and their results on the former are more significant.)

In this paper, we do not consider set-up times, but only consider the MLC problem with minimal beam-on time. In the next section we develop a nonlinear mixed binary programming model, in which we also discuss restrictions on the shape matrices enforced by technological constraints.

3 A Nonlinear Mixed-Integer Programming Formulation

In this section we will give a formulation of the MLC problem as nonlinear program with continuous and binary variables. As before, we assume that $\mathcal{I}$ is a given, integer-valued $m \times n$ intensity matrix.

For any shape matrix $S = (y_{ij})_{\substack{i=1,\ldots,m \\ j=1,\ldots,n}}$, we introduce $(0,1)$ variables characterizing the positions of left and right leaves in each row $i = 1, \ldots, m$.

$$L_{ij} := \begin{cases} 1 & \text{if the left leaf in row } i \text{ is positioned in column } j \\ 0 & \text{otherwise} \end{cases} \quad \forall j = 0, \ldots, n \tag{4}$$

$$R_{ij} := \begin{cases} 1 & \text{if the right leaf in row } i \text{ is positioned in column } j \\ 0 & \text{otherwise} \end{cases} \quad \forall j = 1, \ldots, n+1 \tag{5}$$

Next, we will characterize shape matrices suitable for modeling MLCs by introducing constraints on the L_{ij} and R_{ij} variables.

- Each row has exactly one left and right leaf, respectively:

$$\sum_{j=0}^{n} L_{ij} = 1 \quad \forall i = 1, \ldots, m \tag{6}$$

$$\sum_{j=1}^{n+1} R_{ij} = 1 \quad \forall i = 1, \ldots, m \tag{7}$$

- "Left" and "right" leaves deserve their name:

$$\sum_{j=0}^{n} jL_{ij} \leq \sum_{j=1}^{n+1} jR_{ij} - 1 \quad \forall i = 1, \ldots, m \tag{8}$$

Conditions (6) - (8) characterize left and right leaf positions. Next, we relate variables L_{ij} and R_{ij} to the entries y_{ij} of the shape matrix. Obviously, this relationship is given by the index condition

$$y_{ij} = 1 \quad \Leftrightarrow \quad L_{ij'} = 1 \text{ for some } j' < j \text{ and } R_{ij''} = 1 \text{ for some } j'' > j$$

where j' and j'' are uniquely defined due to (6) and (7). The following set of conditions translates these constraints on the indices into constraints on the variables:

$$y_{ij} \geq \sum_{l=0}^{j-1} L_{il} - \sum_{r=1}^{j} R_{ir} \tag{9}$$

$$y_{ij} \leq \sum_{l=0}^{j-1} L_{il} \tag{10}$$

$$y_{ij} \leq \sum_{r=j+1}^{n+1} R_{ir} \tag{11}$$

The constraints hold for $i = 1, \ldots, m,\ j = 1, \ldots, n$. It is easy to show that the index condition and constraints (9) - (11) are equivalent. Hence we have shown the following result.

Lemma 3.1 *The $m \times n$ $(0,1)$-matrix $S = (y_{ij})$ is a shape matrix with left and right leaves in positions L_{ij} and R_{ij}, respectively, if and only if L_{ij}, R_{ij} and y_{ij} are $(0,1)$ variables satisfying (6) - (11).*

If we want to distinguish different shape matrices S_t we add an additional index t to the variables characterizing left and right leaf position and radiation, i.e. we use in (6) - (11) the notation L_{ijt}, R_{ijt} and y_{ijt}. We use T to denote an upper bound on the number of shape matrices required.

With this notation $\alpha_t y_{ijt}$ is the amount of time shape matrix $S_t = (y_{ijt})$ is releasing radiation at bixel (i,j). The problem of decomposing a given $m \times n$ intensity matrix $\mathcal{I} = (I_{ij})$ into shape matrices can therefore be written as the problem of finding $\alpha_t \geq 0$ and $y_{ijt} \in \{0,1\}$ satisfying (6) - (11), so that

$$\sum_{t=1}^{T} \alpha_t\, y_{ijt} = I_{ij} \quad \forall i = 1, \ldots, m,\ j = 1, \ldots, n \tag{12}$$

holds. Consequently, the MLC problem (1) can be formulated as the following (nonlinear) mixed integer program

$$\begin{aligned} \min \quad & \sum_{t=1}^{T} \alpha_t && (13)\\ \text{subject to} \quad & (6) - (12) \\ & L_{ijt}, R_{ijt}, y_{ijt} \in \{0,1\} \quad \forall i = 1, \ldots, m,\ j = 1, \ldots, n,\ t = 1, \ldots, T \\ & L_{i0t}, R_{i(n+1)t} \in \{0,1\} \quad \forall i = 1, \ldots, m,\ t = 1, \ldots, T \\ & \alpha_t \geq 0 \quad \forall t = 1, \ldots, T. \end{aligned}$$

As a mathematical programming model, in contrast to approaches considered so far in the literature (for example, those of [Siochi, 1999], [Xia and Verhey, 1998], [Yu et al., 1995b]), (13) is very versatile and can be easily modified and extended to accommodate additional constraints. As an example, we consider the exclusion of **interleaf motion**, i.e. the left leaf in one row should not be to the right of the right leaf in an adjacent row and vice versa (otherwise crashes between the two leaf points or significant radiation leakage in areas which ought to be blocked will occur). Similar to the left-right condition (8), the exclusion can easily be modeled by

$$\sum_{j=0}^{n} jL_{ij} \leq \sum_{j=1}^{n+1} jR_{i-1,j} - 1 \quad \forall i = 2, \ldots, m \tag{14}$$

and

$$\sum_{j=0}^{n} jL_{ij} \leq \sum_{j=1}^{n+1} jR_{i+1,j} - 1 \quad \forall i = 1, \ldots, m-1 \tag{15}$$

An additional example, the exclusion of so-called **tongue-and-groove pairs** has been discussed in Lenzen (2000).

The complexity status of problem (13) has so far not been discussed, nor have its variations with objective functions (2) and (3). Furthermore, although the mixed integer program in (13) has the disadvantage of being nonlinear, it can be approached by column generation, with the y, L and R variables becoming variables in the column generation subproblem, and (13) becoming the (now linear programming) master problem in α (see Section 4 for details of this approach).

In the next section we will show that the MLC problem can be solved in polynomial time by using a reformulation as a network flow problem with side constraints. In doing so, we also show that the column generation subproblems correspond to pure network flow problems, and so the column generation approach can solve the problem in polynomial time. This result will be more widely useful, as the same column generation subproblem is likely to arise within column generation approaches to the dosage problem.

4 A Network Flow Formulation

In this section we will show that shape matrices satisfying the interleaf motion constraints can be represented as paths in a suitably chosen network, the **shape matrix graph** $G_s = (V_s, E_s)$. This network is introduced first.

G_s is a layered digraph, consisting of m layers which correspond to the m rows of a shape matrix. In each layer $i = 1, \ldots, m$ there are $\frac{1}{2}(n+1)(n+2)$ vertices, denoted by (i, l, r), which represent potential positions l and r of the left and right leaf in row i, respectively. Here $l \in \{0, 1, \ldots, n\}$, $r \in \{1, \ldots, n, n+1\}$ and $l + 1 \leq r$. Two dummy nodes D and D'

which act as start and end nodes will be added, so

$$V_s := \{(i,l,r) : i = 1,\ldots,m,\ l = 0,1,\ldots,n,\ r = 1,\ldots,n,n+1,\ l+1 \le r\} \cup \{D, D'\}. \tag{16}$$

The arc set E_s contains all edges from D to the first layer 1 and from the last layer m to D', i.e. E_s contains

$$E_+(D) := \{(D,(1,l,r)) : (1,l,r) \in V_s\} \tag{17}$$

$$E_-(D') := \{((m,l,r),D') : (m,l,r) \in V_s\} \tag{18}$$

For $i = 1,\ldots,m-1$ we define the arcs between layers to be

$$E_+(i) := \{((i,l,r),(i+1,l',r')) : (i,l,r,),(i+1,l',r') \in V_s,\ l' \le r-1,\ r' \ge l+1\}. \tag{19}$$

These arcs reflect the exclusion of interleaf motion (see (14) and (15)). Finally we add the return arc (D', D), so the set of all arcs is given by

$$E_s := E_+(D) \cup E_-(D') \cup \{(D', D)\} \cup \bigcup_{i=1}^{m-1} E_+(i). \tag{20}$$

An example of a shape matrix graph G_s is shown in Figure 5

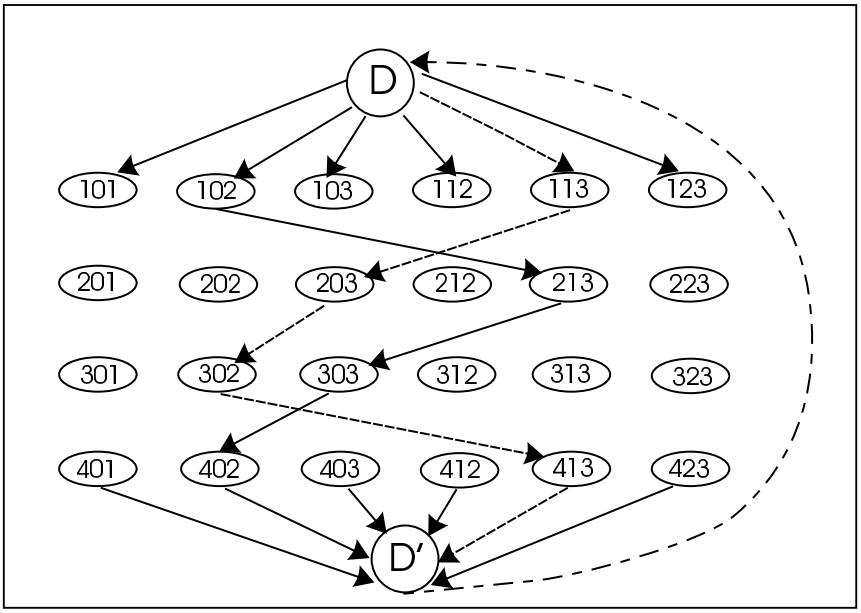


Figure 5: *Shape matrix graph with complete vertex set V_s and some of the arcs of E_s, including two cycles C_1 and C_2 (straight and dotted arcs).*

The next lemma states some properties of G_s. The proof of this lemma is an immediate consequence of the definition of G_s.

Lemma 4.1

(1) $G_s \setminus \{(D', D)\}$ *is an acyclic digraph.*

(2) Every cycle in G_s *corresponds to a shape matrix without interleaf motion and vice versa.*

Figure 6 shows the shape matrix and the leaf configuration corresponding to the cycle

$$C = (D, (102), (213), (303), (402), D', D)$$

in the shape matrix graph of Figure 5.

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{pmatrix}$$

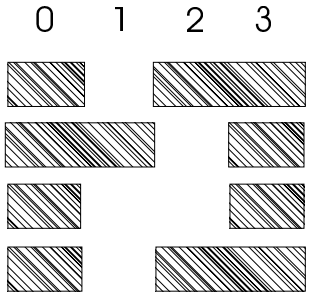


Figure 6: *Shape matrix (left) and leaf configuration (right) corresponding to cycle* $C = (D, (102), (213), (303), (402), D', D)$ *in* G_s *of Figure 5.*

The column generation master problem for the column generation form of the MLC problem (13) is given by the following linear program in α:

$$\begin{aligned} \min \quad & \sum_{t \in \mathcal{T}} \alpha_t \\ \text{subject to} \quad & \sum_{t \in \mathcal{T}} \alpha_t S_t = \mathcal{I} \\ & \alpha_t \geq 0 \quad \forall t \in \mathcal{T} \end{aligned} \tag{21}$$

where $\mathcal{T}$ is the index set of all feasible shape matrices, which we now allow to be generated "on the fly". Consider the column generation subproblem, which seeks the shape matrix of least reduced cost. If we let v_{ij} denote the dual multiplier of the (ij)th intensity constraint, where $S_t = (y_{ijt})$,

$$\sum_{t \in \mathcal{T}} \alpha_t \, y_{ijt} = I_{ij} \quad \forall i = 1, \ldots, m, \; j = 1, \ldots, n, \tag{22}$$

then the column generation subproblem can be expressed as

$$\begin{aligned} \min \quad & 1 - \sum_{i=1}^{m} \sum_{j=1}^{n} v_{ij} \, y_{ij} \\ \text{s.t} \quad & S = (y_{ij}) \text{ is shape matrix.} \end{aligned} \tag{23}$$

Using Lemma 4.1 this subproblem can be solved as a shortest path problem in the shape network $G_s = (V_s, E_s)$ by setting arc **length**

$$l_e := -\sum_{j=l+1}^{r-1} v_{ij} \tag{24}$$

for each arc $e \in E_s$ with end node $(i, l, r) \in V_s$ and $l + 1 < r$. All other arcs have zero arc length. Clearly, the shortest path from D to D' in G_s corresponds to the shape matrix which solves the column generation subproblem (23). Since a shortest path problem in an acyclic network is solvable in polynomial time, and so is a linear program, then by the equivalence between separation and optimization (see e.g. [Nemhauser and Wolsey, 1988]) we have shown the following result.

Theorem 4.2 *The MLC problem with minimal beam-on time is solvable in time polynomial in n and m.*

While Theorem 4.2 is proved by the column generation argument above, we will subsequently give an alternative proof by transforming the MLC problem into a polynomially solvable network flow problem with side constraints.

The key observation is that due to formulation (21) and the equivalence between shape matrices and cycles in G_s, $\sum_{t \in \mathcal{T}} \alpha_t$ can be interpreted as the value $x(D', D)$ of a circulation x which is a composite of flows α_t on cycles C_t corresponding to the shape matrices S_t.

If we consider, for instance, the two cycles

$$C_1 = (D, (102), (213), (303), (402), D', D)$$

and

$$C_2 = (D, (113), (203), (302), (413), D', D)$$

of the shape matrix graph in Figure 5 and $x(C_1) = 3$, $x(C_2) = 2$, (i.e. x is a circulation formed by taking 3 units of flow on cycle C_1 and adding 2 units of flow on cycle C_2), we get a flow corresponding to the intensity matrix

$$3\begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{pmatrix} + 2\begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 3 & 2 \\ 2 & 5 \\ 5 & 3 \\ 3 & 2 \end{pmatrix}.$$

Since by part (1) of Lemma 4.1 all cycles contain arc (D', D) we have that

$$x(D', D) = x(C_1) + x(C_2) = 5$$

represents the flow value which is to be minimized. Since, conversely, every circulation can be decomposed into cycles containing arc (D', D) (see [Ahuja et al., 1993] or

[Hamacher and Klamroth, 2000]), the MLC problem can be solved as a network flow problem in G_s, in fact as a minimum cost circulation problem, with respect to arc costs given by

$$c_e := \begin{cases} 1 & \text{if } e = (D', D) \\ 0 & \text{if } e \in E_s \setminus \{(D', D)\} \end{cases}$$

for each $e \in E_s$.

We have, however, to enforce the side constraint that the values I_{ij} of the intensity matrix $\mathcal{I}$ are generated by the circulation, i.e. we have the side constraints

$$\sum_{l=0}^{j-1} \sum_{r=j+1}^{n+1} \sum_{e \in E_-(i,l,r)} x_e = I_{ij} \tag{25}$$

for all $i = 1, \ldots, m$ and all $j = 1, \ldots, n$, where we write $E_-(q)$ to denote the set of edges in E_s entering node q, i.e. $\{(p, q) \in E_s\}$.

The network circulation formulation, together with the side constraints (25), provide a linear programming formulation for the MLC problem with minimal beam-on time. Note that its size, in terms of numbers of variables and constraints, is polynomial in n and m, as opposed to the column generation formulation, which has exponentially many variables.

However, we do not rest there: it is possible to get a linear programming formulation which is even closer to a pure network flow formulation. Rather than stating (25) as an algebraic constraint, the shape matrix graphs is expanded to a network $\hat{G}_s = (\hat{V}_s, \hat{E}_s)$ defined as follows (see Figure 7).

Each node $(i, l, r) \in \mathcal{N}_s := V_s \setminus \{D, D'\}$ is split into two nodes $(i, l, r)^1$ and $(i, l, r)^2$. The idea is that flow will always enter row i via a node of the form $(i, l, r)^1$ and may only leave row i via a node of the form $(i, l, r)^2$. In between, we will have the flow go through arcs representing cells which are irradiated in row i if leaf positions (l, r) are used. Thus we introduce new nodes of the form (i, j), $i = 1, \ldots, m$, $j = 0, \ldots, n$. Hence

$$\begin{aligned} \hat{V}_s \;:=\; & \left\{(i, l, r)^1, (i, l, r)^2 : (i, l, r) \in \mathcal{N}_s\right\} \\ & \cup \{(i, j) : i = 1, \ldots, m,\ j = 0, \ldots, n\} \cup \{D, D'\} \end{aligned} \tag{26}$$

Note that

$$|\hat{V}_s| = m(n+1)(n+3) + 2$$

nodes, since $|N_s| = \frac{1}{2}m(n+1)(n+2)$, and $|\hat{V}_s| = 2|N_s| + m(n+1) + 2 = m(n+1)(n+2) + m(n+1) + 2$. Obviously this is polynomial in n and m; it is $O(mn^2)$.

The arc set $\hat{E}_s$ consists of the following subsets

$$\begin{aligned}\hat{E}_s^{\text{old}} &:= \left\{\left((i,l,r)^2,(i+1,l',r')^1\right) : ((i,l,r),(i+1,l',r')) \in E_s\right\} \\ &\cup \left\{\left(D,(1,l,r)^1\right) : (1,l,r)^1 \in \hat{V}_s\right\} \\ &\cup \left\{\left((m,l,r)^2,D'\right) : (m,l,r)^2 \in \hat{V}_s\right\} \\ &\cup \{(D',D)\} \end{aligned} \tag{27}$$

are copies of the edges in E_s connected with appropriate nodes in $\hat{V}_s$. In addition we have new arcs of two types.

$$\begin{aligned}\hat{E}_s^1 &= \left\{\left((i,l,r)^1,(i,l)\right) : (i,l,r)^1 \in \hat{V}_s\right\} \\ &\cup \left\{\left((i,r-1),(i,l,r)^2\right) : (i,l,r)^2 \in \hat{V}_s\right\}\end{aligned}$$

All arcs $e \in \hat{E}_s^{\text{old}} \cup \hat{E}_s^1$ considered so far have

$$\text{lower capacities } \underline{u}_e = 0 \text{ and upper capacities } \overline{u}_e = \infty \tag{28}$$

The second set of new arcs is given by

$$\hat{E}_s^2 := \{((i,j-1),(i,j)) : i = 1,\ldots,m,\ j = 1,\ldots,n\}\,.$$

These are called the **intensity arcs**, since the flow on these arcs has to be equal to the intensity. Therefore, we set lower and upper capacity on these arcs to be

$$\underline{u}_e = \overline{u}_e = I_{ij} \quad \forall e = ((i,j-1),(i,j)) \in \hat{E}_s^2 \tag{29}$$

The idea is that where previously flow went into a node (i,l,r), and implicitly this meant the flow would contribute to the intensity constraint for cells $(i,j), j = l+1,\ldots,r-1$, now we force the flow to go into node $(i,l,r)^1$, then through arcs $((i,j-1),(i,j))$ for $j = l+1,\ldots,r-1$, thus contributing to the requirement for lower = upper capacity (= intensity) in those arcs, and then back to node $(i,l,r)^2$ before going on to the next row $i+1$. However, to ensure this flow sequence occurs in the flow decomposition, we must still enforce a side constraint, albeit a simpler one than (25), namely that the flow in arc $((i,l,r)^1,(i,l))$ must equal the flow in arc $((i,r-1),(i,l,r)^2)$ for all $(i,l,r) \in V_s$.

It is straightforward to show that $|\hat{E}_s|$ is $O(mn^4)$, (for each of the m rows, there are $O(n^2)$ nodes, so there are $O(n^4)$ arcs of the form $((i,l,r)^2,(i+1,l'r')^1)$, passing from one row to the next, and this is the dominant term), which is polynomial in n and m.

We have therefore established the following result.

Theorem 4.3 *The MLC problem with minimal beam-on-time is equivalent to the network flow problem*

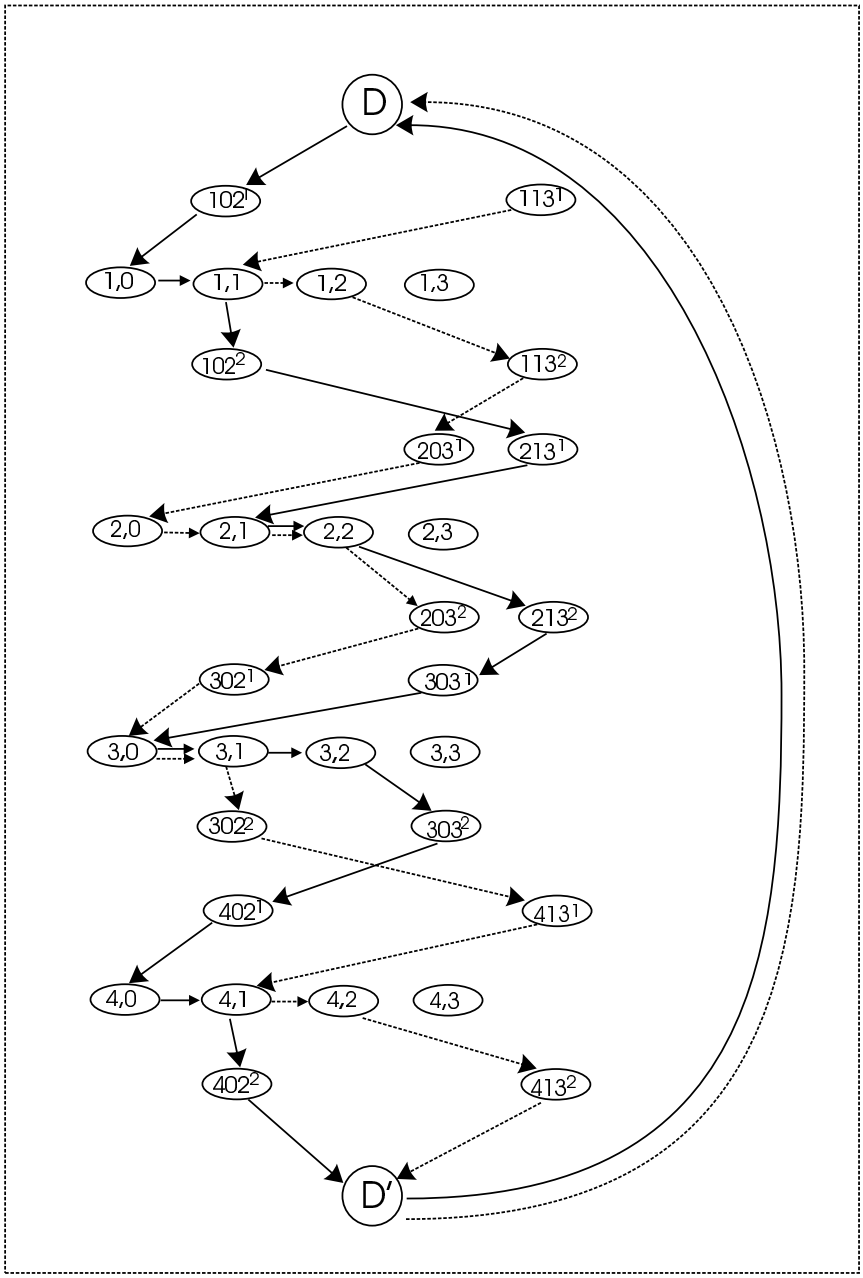


Figure 7: *Extended shape matrix graph. Only nodes and arcs of extended cycles C_1 and C_2 of Figure 5 are shown.*

$$\min x(D', D)$$

subject to x *a circulation in* $\hat{G}_s$ *lying between lower and upper capacity limits* $\underline{u}$ *and* $\overline{u}$*, defined by (28) and (29), and satisfying*

$$x\left((i,l,r)^1,(i,l)\right) = x\left((i,r-1),(i,l,r)^2\right) \quad \forall (i,l,r) \in V_s. \tag{30}$$

Theorem 4.3 also yields an alternative proof of the polynomial solvability of the MLC problem with minimal beam-on time.

The two side-constrained network flow models we have presented here may be of some interest in their own right. For example, we will address the question of integrality of solutions in future work. Combinatorial algorithms for solving them, such as generalization of negative dicycle and shortest augmenting dipath algorithms, (see [Ahuja et al., 1993]), which consider side constraint (30), are currently under investigation ([Görg, 2001]).

5 Numerical Experience

In this section, we compare the results of solving the MLC problem with minimal beam-on time using the network flow approach with those of the well-known heuristic methods of [Siochi, 1999] and [Xia and Verhey, 1998]. Note that we showed in Section 4 that the network flow algorithms solve this problem up to optimiality in polynomial time while the latter two heuristics usually compute suboptimal solutions.

We implemented both network flow approaches introduced in the preceding sections using the package AMPL (see [Fourer et al, 1993]), and found that the one on the graph G_s required less CPU time on our test problems than the model on the expanded graph. (Of course, the solutions given by both were the same.) However in both cases, the time required by the CPLEX solver in AMPL to solve the linear programs is negligible.

We refer to our method as **NetFlow** in Table 1 and Figure 8. The methods of [Siochi, 1999] and [Xia and Verhey, 1998] (denoted **Siochi** and **XV**, respectively) were implemented in C++. Details of the methods as implemented by us can be found in [Lenzen, 2000].

Our test set consists of fifteen 10×10 randomly generated intensity matrices. Each entry in the matrix is an integer value between 1 and 15, with each integer having equal probability of being selected. All methods were run on the same machine (1 GHz PC). The results, summarized in Table 1 and Figure 8, show that the network flow approach produces, as expected, a radiation plan with the smallest beam-on time in every single example. The difference in the beam-on times computed by the three methods in in some examples quite large. We note, however, that the comparison here is slightly unfair, in that the emphasis of the [Xia and Verhey, 1998] heuristic is on minimizing the number of shape matrices required. [Siochi, 1999] was concerned with both the number of shape matrices and beam-on time.

Table 1: Total beam-on time used by the three methods

Data Set	**XV**	**Siochi**	**NetFlow**
1	87	50	39
2	79	40	33
3	95	42	37
4	87	39	37
5	81	58	46
6	77	51	45
7	87	52	47
8	73	48	41
9	87	40	33
10	89	47	40
11	71	47	41
12	93	43	37
13	69	40	32
14	93	47	43
15	93	42	37

The same dominance of the network flow approach can, however, also be observed if the set-up time is taken into consideration as well (for instance by multiplying the number of different shape matrices with a fixed set-up time and adding it to the beam-on time). In every example we have tested so far, the resulting treatment time was shorter than the treatment times obtained by the algorithms of [Siochi, 1999] and [Xia and Verhey, 1998]. These results are not completely understood at this point of time and will be discussed in a forthcoming paper.

6 Conclusion and Further Research

We have presented new formulations for the multileaf collimator problem: a mixed integer, nonlinear model, a decomposition of this model leading to a column generation formulation, and two alternative network flow models with side constraints. The latter is derived using a network in which paths from a designated source to a designated sink node correspond to feasible shape matrices. This network is used to show that the column generation subproblem can be modelled as a shortest path problem. This, and more directly the side-constrained network flow formulations, show that the MLC problem with minimum beam-on time is polynomially solvable. We illustrated this result by solving a set of randomly generated test problems, and comparing the solutions with those of well known heuristics. We found that the beam-on time is, indeed, reduced by a considerable amount of time using our new approach.

Our future research efforts go into the following directions.

Currently, we use AMPL to solve the network flow problem with side constraints. This approach is reasonable for the MLC problem with minimal beam-on-time since the CPU time for a single problem is neglectably small. Nevertheless, streamlining of the network flow algorithm is investigated in a diploma thesis (Görg 2002). We expect an additional speed up by taking more advantage of the network structure of the problem. This speed up can be of use in solving the other versions of the MLC problem. In particular, we will benefit from this by tackling the MLC problem with minimal beam-on time and constant set-up time.

Moreover, variable set-up times ought to be considered. As a measure of switching from one shape matrix S_p to another S_q one could, for instance, investigate

$$c(S_p, S_q) = \beta \max_{i=1,\ldots,m} \max \left\{ |l_i^p - l_i^q| \, , |r_i^p - v_i^q| \right\},$$

the largest distance a left or right leaf will have to move (multiplied with some time factor β). As briefly discussed at the end of Section 3, the resulting MLC problems will have a TSP like constraint. Although the variable set-up time MLC model is more realistic than the ones described in this paper, its solution will be a very big challenge.

7 Acknowledgment

Natashia Boland acknowledges partial support from the Fachbereich Mathematik, University of Kaiserslautern which allowed her to stay in Kaiserslautern from October through November 2000. We thank Davaa Bataar, Florentine Bunke, Andre Görg , and Berhanu Guta, University of Kaiserslautern, for several discussions on the subject of the paper.

References

[Ahuja et al., 1993] Ahuja, R. K., Magnanti, T., and Orlin, J. (1993). *Networks Flows.* Prentice Hall.

[Bortfeld, 1995] Bortfeld, T. (1995). Dosiskonformation in der Tumortherapie mit externer ionisierender Strahlung: Physikalische Möglichkeiten und Grenzen. Habilitationsschrift, Deutsches Krebsforschungszentrum, Heidelberg.

[Bortfeld et al., 1994] Bortfeld, T., Kahler, D., Waldron, T., and Boyer, A. (1994). X-ray field compensation with multileaf collimators. *International Journal of Radiation Oncology, Biology and Physics*, 28:723–730.

[Burkard et al., 1995] Burkard, R., Leitner, H., R. Rudolf, R., Siegl, T., and Tabbert, E. (1995). Discrete Optimization Models for Treatment Planning in Radiation Therapy.

In Hutten, H., editor, *Science and Technology for Medicine, Biomedical Engineering in Graz*. Pabst, Lengerich.

[Dai and Hu, 1999] Dai, J.-R. and Hu, Y.-M. (1999). Intensity-modulation radiotherapy using independent collimators: an algorithm study. *Medical Physics*, 26:2562–2570.

[Fourer et al, 1993] Fourer, R., Gay, D.M. and Kernighan, B.W. (1993). *AMPL: A modeling language for mathematical programming*. The Scientifc Press.

[Galvin et al., 1993] Galvin, J., Chen, X., and Smith, R. (1993). Combining multileaf field to modulate fluence distributions. *International Journal of Radiation Oncology, Biology and Physics*, 27:697–705.

[Goldberg and Tarjan, 1989] Goldberg, A.V. and Tarjan, R.E. (1989). Finding minimum cost circulations by cancelling negative cycles. *Journal of the ACM*, 36:873–886.

[Görg, 2001] Görg, A. (2001). Anwendungen von Netzwerkflüssen bei der Realisierung von Krebsbestrahlung durch Multileaf Kollimatoren. Master's thesis, University of Kaiserslautern.

[Hamacher and Klamroth, 2000] Hamacher, H. and Klamroth, K. (2000). *Linear and Network Optimization*. Vieweg Verlag.

[Hamacher and Küfer, 2001] Hamacher, H. and Küfer, K.-H. (2001). Inverse Radiation Therapy Planning - A Multiple Objective Optimization Approach. *Discrete Applied Mathematics*. To appear.

[Kolmonen et al., 1998] Kolmonen, P., Tervo, J., and Lahtinen, T. (1998). Use of the cimmino algorithm and continuous approximation for the dose deposition kernel in the inverse problem of radiation treatment planning. *Phys. Med. Biol.*, 43:2539–2554.

[Lee et al., 2000] Lee, E., Fox, T., and Crocker, I. (2000). Optimization of radiosurgery treatment planning via mixed integer programming. *Medical Physics*, 27:995–1004.

[Lenzen, 2000] Lenzen, F. (June 2000). An Integer Programming Approach to the Multileaf Collimator Problem. Master's thesis, University of Kaiserslautern.

[Nemhauser and Wolsey, 1988] Nemhauser, G. and Wolsey, L. (1988). *Integer and Combinatorial Optimization*. John Wiley & Sons.

[Que, 1999] Que, W. (1999). Comparison of algorithms for multileaf collimator field segmentation. *Medical Physics*, 26:2390–2396.

[Raphael, 1992] Raphael, C. (1992). Mathematical models of objectives in radiation therapy treatment planning. *Phys. Med. Biol.*, 1992:1293–1311.

[Shepard et al., 1999] Shepard, D., Ferris, M., Olivera, G., and Mackie, T. (1999). Optimizing the Delivery of Radiation Therapy to Cancer Patients. *SIAM Review*, 41(1):721–744.

[Siochi, 1999] Siochi, R. (1999). Minimizing static intensity modulation delivery time using an intensity solid pradigm. *International Journal of Radiation Oncology Biology Physics*, 43:671–680.

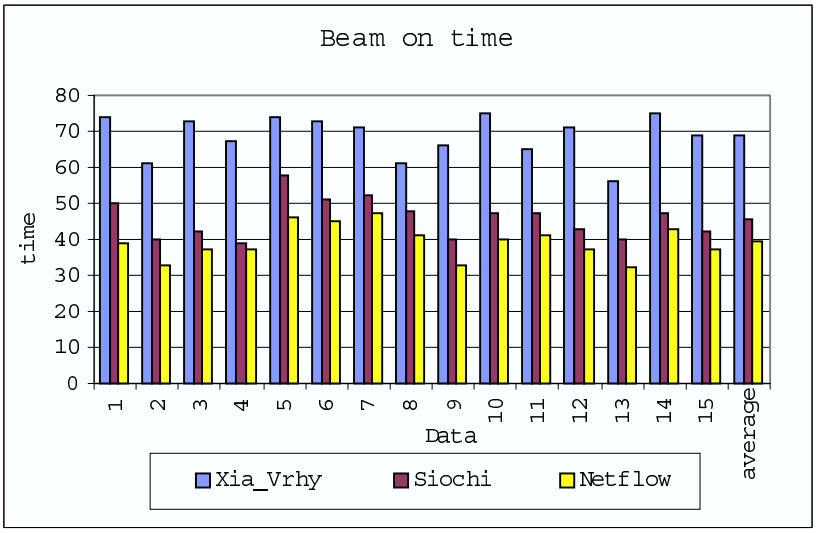
[Tervo and Kolmonen, 2000] Tervo, J. and Kolmonen, P. (2000). A model for the control of a multileaf collimator in radiation therapy treatment planning. *Inverse Problems*, 16:1875–1895.

[Wu and Zhu, 2001] Wu, X. and Zhu, Y. (2001). An optimization method for importance factors and beam weights based on genetic algorithms for radiotherapy treatment planning. *Phys. Med. Biol.*, 46:1085–1099.

[Xia and Verhey, 1998] Xia, P. and Verhey, L. (1998). Multileaf collimator leaf sequencing algorithm for intensity modulated beams with multiple segments. *Medical Physics*, 25:1424 – 1434.

[Yu et al., 1995a] Yu, C., Symons, M., Du, M., Martinez, A., and Wong, J. (1995a). A method for implementing dynamic photon beam intensity modulation using independent jaws and a multileaf collimator. *Physcs in Medicine and Biology*, 40:769–787.

[Yu et al., 1995b] Yu, C., Yan, D., Du, M., Zhou, S., and Verhey, L. (1995b). Optimization of leaf-positions when shaping a radiation field with a multileaf collimator. *Physics in Medicine and Biology*, 40:305–308.


Beam on time
time
80
70
60
50
40
30
20
10
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
average
Data
Xia_Vrhy
Siochi
Netflow