Xavier Mellado Esteban

# New methods in intensity-modulated radiation therapy for treatment time reduction

Departamento

Ingeniería Electrónica y Comunicaciones

Director/es

Artacho Terrer, Juan Manuel

ISSN 2254-7606

Prensas de la Universidad
Universidad Zaragoza

Tesis Doctoral

# NEW METHODS IN INTENSITY-MODULATED RADIATION THERAPY FOR TREATMENT TIME REDUCTION

Autor

Xavier Mellado Esteban

Director/es

Artacho Terrer, Juan Manuel

**UNIVERSIDAD DE ZARAGOZA**

Ingeniería Electrónica y Comunicaciones

2011

1542

Escuela de
Ingeniería y Arquitectura
Universidad Zaragoza

Instituto Universitario de Investigación
en Ingeniería de Aragón
Universidad Zaragoza

# NEW METHODS IN
# INTENSITY-MODULATED RADIATION THERAPY
# FOR TREATMENT TIME REDUCTION

XAVIER MELLADO ESTEBAN

Ph.D. Thesis
Official Postgraduate Program On Transversal Engineering
Biomedical Engineering

Supervised by Juan Manuel Artacho Terrer

Communication Technologies Group (GTC)
Aragón Institute for Engineering Research (I3A)

Department of Electronic Engineering and Communications (DIEC)
Escuela de Ingeniería y Arquitectura (EINA)
Universidad de Zaragoza

July 2010

Xavier Mellado Esteban: *New methods in intensity-modulated radiation therapy for treatment time reduction,* © July 2010.

NUEVOS MÉTODOS EN RADIOTERAPIA DE INTENSIDAD MODULADA PARA LA REDUCCIÓN DE LOS TIEMPOS DE ADMINISTRACIÓN DE LOS TRATAMIENTOS

# RESUMEN

El cáncer es hoy en día una de las principales causas de muerte en todo el mundo con 7,6 millones de fallecidos en 2008[1], lo que equivale a un 13 % del total de muertes. Para combatir este grupo de enfermedades, se han desarrollado múltiples técnicas que pueden ser administradas individualmente o de forma conjunta para incrementar su efectividad. La radioterapia de intensidad modulada (abreviatura en inglés IMRT) es una técnica que permite concentrar la dosis de radiación en el volumen del tumor, a la vez que limita la radiación que reciben los tejidos sanos adyacentes. Sin embargo, la complejidad del tratamiento se incrementa frente a otras técnicas de radioterapia convencional como 3DCRT, provocando que el tiempo de administración en cada sesión sea mayor y que la validación clínica resulte más difícil. La presente tesis se centra en la modalidad *step-and-shoot* de IMRT, con el objetivo de desarrollar nuevos métodos que permitan modelar de forma más precisa el problema tratado y reducir los tiempos de administración.

En el modelado del problema, el mapa de intensidad de cada haz y el volumen interior del paciente son discretizados en *beamlets* y *vóxels*, respectivamente, para poder ajustar la dosis a la forma tridimensional del tumor. Las intensidades de los beamlets son generadas mediante un proceso de optimización, que es guiado por los cambios que se producen en la función objetivo obtenida a partir de la distribución de dosis conseguida en el interior del paciente. Por esta razón, el modelo de radiación tiene un papel crucial, ya que es el encargado de obtener la dosis acumulada en cada uno de los vóxels cada vez que las intensidades son modificadas. En la presente tesis, proponemos un método de proyección en el modelo de radiación para el cálculo de relaciones entre beamlets y vóxels, que cambia la relación 1 a $n$ asumida en el método de proyección original por una relación $n$ a $n$ más realista, donde un beamlet puede radiar de forma directa a más de un vóxel, y un vóxel puede ser radiado directamente por más de un beamlet. Esto se consigue proyectando el vóxel completo sobre los beamlets que forman el haz de radiación y buscando cuales han sido activados. Este proceso se realiza utilizando la potencia de las actuales tarjetas gráficas para realizar los cálculos sin que el tiempo de ejecución se incremente excesivamente.

Una vez que los mapas de intensidad han sido optimizados, todavía es preciso descomponerlos en una serie de aperturas con un tiempo de exposición asociado para que puedan ser administradas por un acelerador lineal equipado con un MLC. La descomposición tiene multitud

[1] Globocan 2008, IARC, 2010

de soluciones posibles, pero se debe buscar la más simple y eficiente para no prolongar los tiempos de administración y hacer más difícil la validación clínica. Con este objetivo, proponemos reducir los tiempos de administración por medio de tres métodos diferentes pertenecientes a la tarea de descomposición o segmentación MLC. El primero de ellos es un algoritmo de segmentación unidireccional, que mediante la sincronización de las láminas obtiene segmentos con formas más suaves y, a la vez, tiende a evitar la aparición de más de una componente conexa, es decir, procura generar una única apertura por segmento. El segundo método consigue reducciones de hasta un 25 % en el número de segmentos a partir del estudio de una representación tridimensional de la descomposición obtenida, buscando aquellos puntos clave en la matriz de fluencia que dificultan la descomposición y que con una ligera modificación en su fluencia pueden ser eliminados. Por último, se presenta un método que permite post-procesar segmentaciones y reducir el número de segmentos obtenidos a una cantidad prefijada por el usuario. Esta reducción se consigue utilizando la similitud entre aperturas adyacentes en segmentaciones unidireccionales para agruparlas en tantos grupos como segmentos desee el usuario. Posteriormente, una apertura y un peso equivalentes son creados para cada grupo teniendo en cuenta el mapa de fluencia original, a fin de preservar las zonas de alta intensidad y con ello permanecer lo más cerca posible de la fluencia originalmente planificada.

# NEW METHODS IN INTENSITY-MODULATED RADIATION THERAPY FOR TREATMENT TIME REDUCTION

## ABSTRACT

Cancer is a leading cause of death worldwide accounting for 7.6 million deaths (around 13% of all deaths) in 2008[2]. In order to fight this group of diseases, multiple techniques have been developed that can be applied individually or together for increasing their effectiveness. Intensity-modulated radiation therapy (IMRT) is a technique that allows the radiation dose to conform precisely to the three-dimensional shape of the tumour while sparing any critical structure. However, the complexity of the treatment is increased, if it is compared with other conventional radiotherapy techniques like 3DCRT. As a consequence, treatment times are extended and the clinical validation becomes more challenging. This thesis is focused on the development of new methods for the *step-and-shoot* mode of IMRT, aiming at improving the modelling of the problems solved and reducing the treatment times.

For modelling the problem and conforming the radiation dose to the shape of the tumour, the intensity map for each beam and the patient volume are discretized into beamlets and voxels, respectively. Beamlet intensities are obtained by optimization, and the objective function that drives the optimization process is computed from the dose distribution achieved inside the patient. For this reason, the radiation model has a crucial role, since it is used to obtain the accumulated dose in every voxel after a change is performed in beamlet intensities. In this thesis, we propose a projection method for the radiation model to compute the relationship between beamlets and voxels, where the 1-to-n relationship assumed by original projection method is exchanged for a more realistic n-to-n relationship, so that one beamlet can directly radiate many voxels, and one voxel can be directly radiated by many beamlets. The solution is found by projecting the whole voxel on the beamlet grid and searching for the activated beamlets. The method uses the computer graphic card inside a compute-by-drawing approach that performs the calculations without increasing in excess computation times.

Once the intensity maps have been optimized, it is still needed to decompose them in a set of segments with an associated beam-on-time, for being delivered with a linear accelerator equipped with an MLC. The decomposition has multiple solutions, but the most simple and efficient has to be found in order to do not extend treatment times or make the clinical validation more difficult. To this end, we propose reducing delivery times with three methods for the MLC segmentation step. The first method is a unidirectional segmentation method, which synchronizes the leaves motion for obtaining segments with smoother

[2] Globocan 2008, IARC, 2010

contours and being prone to avoid more than one connected component, i.e. it tends to generate segments with only one aperture. The second method achieves reductions up to 25% in the number of segments. The original segmentation is represented in a three-dimensional structure, where the algorithm can search for the key points of the fluence map that make the decomposition more difficult and that can be erased with a small change in its fluence. Finally, it is presented a method for post-processing unidirectional segmentations and reducing the number of segments to a user-specified quantity. The method takes advantage of the similarity between adjacent segments in unidirectional segmentations. This similarity allows the algorithm to cluster the original segments into the same number of groups as the desired number of segments. Then, an equivalent aperture and weight is computed for each group, taking into account the original fluence map in order to preserve the highest intensity zones and stay as close as possible to the original planned fluence.

# ACKNOWLEDGEMENTS

I would like to thank Guillermo Tobías for his previous work on the MLC segmentation field, which left tons of routines that helped me and made the development of this thesis easier. I also thank my supervisor Juan Manuel Artacho for always having time to discuss with me, Mónica Hernández for her invaluable hints and advices, Técnicas Radiofísicas S.L. for allowing the use of the PCRT 3D$^{\circledR}$ software, and the Physics and Radiological Protection Service staff at the Hospital Clínico Universitario Lozano Blesa for its collaboration on the experiments and its kindness; especially to Miguel Canellas and Esther Millán, whose positive attitude and willingness to work made the clinical tests possible.

# CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# ACRONYMS

2D two-dimensional

3D three-dimensional

AMRT arc-modulated radiation therapy

BC branch-and-cut

CPU central processing unit

CT computed tomography

CTV clinical target volume

CTVgr clinical target volume gross disease region

CTVel clinical target volume elective nodal region

DAO direct aperture optimization

DMLC dynamic MLC

DMPO direct machine parameter optimization

DSS direct step-and-shot

DVH dose-volume histogram

EUD equivalent uniform dose

FBO frame buffer object

FMO fluence map optimization

GPU graphics processing unit

ICC interleaf collision constraint

IMRT intensity-modulated radiation therapy

MI modulation index

MLC multileaf collimator

MU monitor units

NS number of segments

OAR organ at risk

ONS optimizing the number of segments

OTNMU optimizing the total number of monitor units

PTV planning target volume

ROI region of interest

RP rod pushing

SMLC static MLC

TGC tongue-and-groove constraint

TNMU total number of monitor units

V&R verification and recording

# 1 INTRODUCTION

## Contents

## 1.1 THESIS SCOPE

Radiation therapy (in North America), or radiotherapy (in the UK and Australia) is the medical use of ionizing radiation for the treatment of malignant tumours (cancers). Most common cancer types can be treated with radiotherapy in some way and it can be used as the primary therapy or combined with surgery, chemotherapy, hormone therapy or some mixture of them. The precise treatment intent (curative, neoadjuvant or palliative) will depend on the tumour type, location, and stage, as well as the general health of the patient.

*Neoadjuvant therapy refers to radiation therapy given to people with cancer prior to surgery. The aim is to reduce the size or extent of the cancer before receiving surgery, thus making procedures easier and more likely to be successful.*

This thesis is focused on an external radiotherapy technique called intensity-modulated radiation therapy (IMRT), which is increasingly used for cancer treatment in complicated body sites such as head and neck, prostate, breast and lung. IMRT allows radiation doses to conform the three-dimensional (3D) shape of the tumour while minimizing the dose to surrounding critical structures and normal tissue. By contrast, the treatment complexity is increased and it is crucial to generate treatment plans that can be delivered efficiently and accurately while meeting the treatment goals. The aim of the research described in this thesis was to develop, evaluate and clinically validate methods that either improve the modelling of the problem solved in IMRT or reduce the delivery times.

The content of this thesis is divided into six chapters. The fist chapter is an introduction to the IMRT technique, describing the steps for generating a treatment plan, together with the problems involved in each step and the most common approaches to deal with them. Second chapter is focused on the modelling of the relationship between beamlets and voxels used for computing the dose distribution achieved inside the patient while the intensity maps are optimized. The third, forth and fifth chapters deal with the decomposition of the intensity maps in order to increase the delivery efficiency. Finally, the sixth chapter contains the conclusions and future work.

## 1.2 INTENSITY-MODULATED RADIATION THERAPY

IMRT is an external radiotherapy technique which directs photon (megavoltage X-ray) or electron beams to a *target volume*. The beam collide with the different tissues, then electrons and free radicals are detached and scattered from molecules, and eventually they collide with the DNA molecules of the cells. These collisions lead to a chain of physical and chemical events that eventually produce a biological damage, since the DNA molecule is broken by ionizing or exciting its atoms. This DNA damage is then passed on through cell division, accumulating damage to the cancer cell's DNA, causing them to die or reproduce more slowly. Cancer cells have a diminished ability to repair sub-lethal damage compared to most healthy cells. Consequently, the treatment is conveniently fractionated (spread out over time) for allowing normal cells to recover and repopulate, while tumour cells that are generally less efficient in repair between fractions will be more affected by the radiation. For a extended introduction to the radiobiological processes see [32].

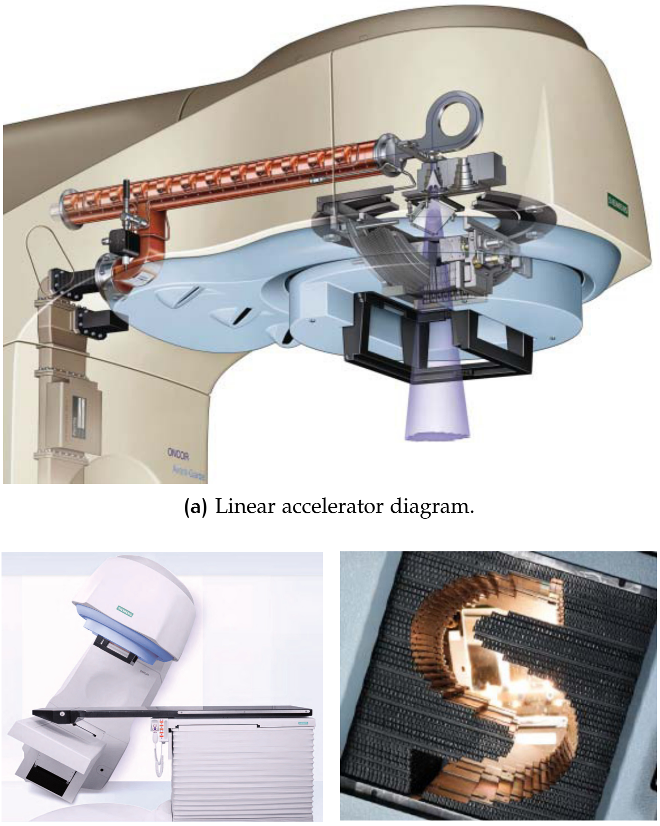
A computer-controlled linear accelerator equipped with a multileaf collimator (MLC) (figure 1) is used in IMRT to deliver the radiation and achieve inside the patient a specific level of radiation dose, the so-called *prescribed dose*. The dose is defined as the amount of radiation absorbed by the tissue and it is measured in grays (Gy), where one gray is the absorption of one joule of energy, in the form of ionizing radiation, by one kilogram of matter. The linear accelerator output is measured in monitor units (MU), which are calibrated for a specific energy such that one MU gives an absorbed dose of one gray in a water phantom at a specific depth of $D_{max}$ (the maximum dose along the beam central axis) for a field size of 10x10 cm with a source-to-axis (or alternatively source-to-surface) distance of 100 cm.

*The linear accelerator arm and the couch can be rotated around a given fixed point, called the isocenter, in such a way that the radiation can be delivered from any direction to this point.*

IMRT allows the dose to conform precisely to the 3D shape of the tumour and to spare any critical structure, usually referred as organ at risk (OAR), and normal tissue by modulating the intensity (fluence) of the beams that radiate the target volume from several spatial positions, see figure 2, and controlling the dose achieved in multiple small volumes inside the patient. A 3D computed tomography (CT) image of the patient is used for defining several regions of interest (ROIs) that are discretized into voxels. Then, the treatment planing system uses computerized dose calculations to determine for each beam the non-uniform intensity map or *fluence map* (which corresponds to the beam field of view discretization into beamlets, sometimes called *bixels*). Beamlet intensities are obtained by optimization in such a way that the intersection of the radiation beams at the isocenter will best conform the radiation to the target volume voxels, while achieving the prescribed dose. Finally, a linear accelerator can only provide uniform radiation, so the planned intensity maps need to be decomposed with a MLC segmentation method in a linear combination of uniform maps, that consist of a shaped aperture (or segment) created by the MLC (figures 1(c) and 3(d)) and an associated weight accounting for a relative beam-on time.

*The word voxel comes from volumetric pixel.*

An MLC contains two opposite banks of metal leaves. For each row, there is one leaf located to the left and another leaf located to the right.



(a) Linear accelerator diagram.

(b) Couch and arm rotated.

(c) Collimator aperture.

Figure 1: Pictures of the Siemens ONCOR™ Impression Plus linear accelerator used in some of the experiments of this thesis. Images courtesy of Siemens.

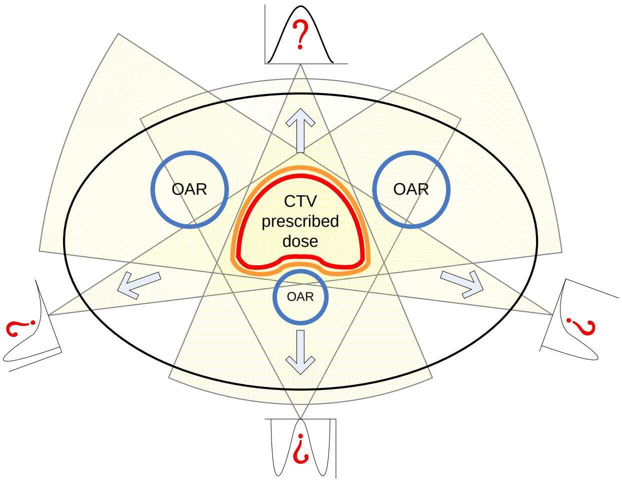


Figure 2: A 2D schematic diagram of an inverse planning. The blue and red contours represent the OARs and the CTV, respectively, and the orange contour is the safety margin added to the CTV in order to account for setup errors and involuntary movement.

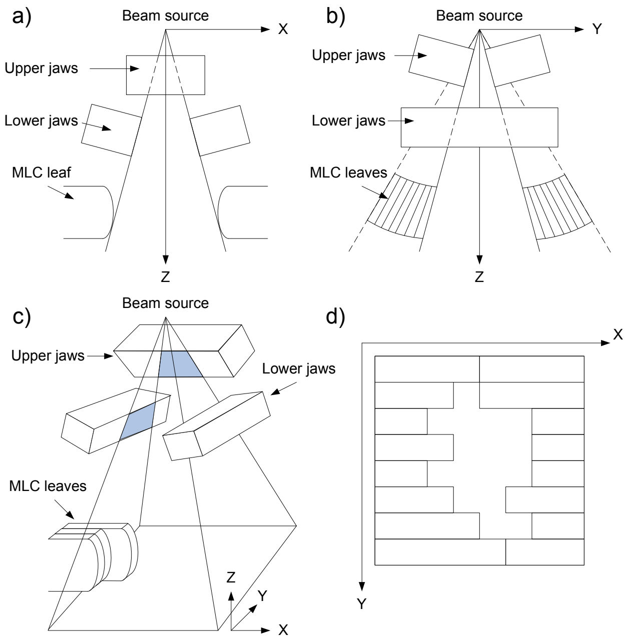


**Figure 3:** Schematic diagram of a generic collimator with upper and lower jaws, and a tertiary MLC. (a) Front (main) view. (b) Side view. (c) Oblique projection. (d) Bird's eye view (floor plan).

Those leaves may move inwards or outwards shaping the beam, as it can be seen in figure 3. The MLC can be used either in static MLC (SMLC) mode [25] or in dynamic MLC (DMLC) mode [58]. These two ways of using the MLC lead to two families of IMRT planning techniques: the SMLC (or step-and-shoot) mode consists of discrete steps, since the beam is off while the leaves are moving, whereas in the DMLC mode, the beam is on while the leaves are continuously moving with variable speed, like a sliding window.

*The step-and-shoot technique has become more popular than the dynamic mode since it is easier to generate the MLC leaf sequencing and validate the results.*

The purpose of this thesis is to develop new methods for the SMLC mode of IMRT, where the two-dimensional (2D) beam field of view is discretized into small beamlets and then the weight or intensity of each beamlet is optimized such that the total contribution of all the beamlets from all the beams produces the desired dose distribution. This process is commonly named as fluence map optimization (FMO). Every beamlet can be understood as a small radiation beam with an associated pencil beam kernel that it is able to describe how the dose is spread. If the relationship between the beamlets and the patient voxels is established, then a optimization algorithm can modify the beamlet intensities for obtaining the desired dose distribution in a *inverse planning*. Finally, it is important to mention that after the optimization, it is obtained the *ideal* or *planned* intensity map. The MLC decomposition and delivery tries to accurately reproduce the ideal map, but usually the delivery phase introduces changes, and the eventual map is called the *actual* or *delivered* intensity map.

In this thesis, it is supposed that the ROIs in a CT image of the patient (or any other image modality) and that the number, spatial position, and field of view of the beams are already defined or computed. The following sections will briefly introduce the last three steps for the treatment planning: the computation of the dose matrix, the optimization of the beamlets intensity and the final MLC decomposition for delivery. It is very important to remark that this thesis focuses on the first and the last step, the optimization of the beamlet intensities is only explained because it is in between both tasks and helps to understand the work done.

## 1.3 RADIATION MODEL

Radiation models are used in IMRT for computing the delivered dose in each iteration during the optimization and at the end of the MLC segmentation for obtaining the final dose distribution and validating the treatment. Basically, it is performed a convolution between the energy released in each voxel and a dose spread kernel introducing some approximations due to limitations in computer speed and incomplete physics. However, these two situations need different approaches because their requirements are the opposite of each other. The beamlet optimization often uses a finite-size *pencil beam* approach that sacrifices accuracy in order to speed up the computations, since the number of iterations may vary from $10^2$ to $10^4$ depending on the optimization algorithm. In contrast, the final dose distribution is usually obtained with a more accurate model such as the collapsed cone in order to obtain the precise dose that the final set of apertures and beam-on times generated will deliver to the patient. A very good introduction to this topic can be found in [28, pp 892–895].

In this thesis, the research performed seeks to address the relationship that must be established between beamlets and voxels for using the pencil beam model during the optimization phase. For each beamlet, the voxels directly affected by its beam are computed, and then the pencil beam kernel is used to compute which is contribution of the surrounding beamlets to the delivered dose. With this information a *dose matrix* is created that describes the amount of radiation deposited by one unit of intensity from a particular beamlet into a particular voxel in the treatment region. There has been a lot of discussion about the radiation spread modelling, such as the pencil beam precision [44, 51], the systematic errors that are introduced in the solution by this kind of model that leads to suboptimal solutions [35, 36], and the influence of the beamlet size [16, 69], for citing only a few studies. However, far too little attention has been paid to the modelling of the relationship between beamlets and voxels needed for generating the dose matrix. There are only a few examples, such as [65] and [29], were it is described in detail how to compute for a given voxel which beamlet or beamlets directly deliver radiation to it, despite the fact that the calculation of this relationship is something to solve in any commercial treatment planning system.

Chapter 2 describes the problem and different solutions. Then, it is proposed a method for fast computing the relationship between beamlets and voxels independently of their relative size, with an extensible and flexible approach for being used with different kind of MLCs and optimization techniques (described in the following section 1.4).

## 1.4 FLUENCE MAP OPTIMIZATION

The optimization of fluence maps can be separated into two components. On the one hand, the optimization criteria, that is, the objective function and the constraints; on the other hand, the optimization algorithm used for searching a solution.

The objective functions (also called cost functions or score functions) are used to evaluate how good is a plan by obtaining the *score* or *cost*, which is a single number, for a given combination of beamlet intensities. This index is for guidance only, the final validation of the treatment quality must be done using an accurate radiation model after the MLC segmentation, as mentioned previously in subsection 1.3.

The objective function is the mathematical representation of the clinical objectives in such a way that the optimization algorithm can use them to search for a solution that minimizes (or maximizes depending on how the function is build) the cost index. One of the most common ways of including dose-volume constraints in the objective function is to minimize the variance of the dose relative to the prescribed dose. The variance is defined as sum of the squares of the differences between the calculated dose and the prescribed dose for the target volume and dose-volume restrictions for the OARs, which can be integrated into the objective function as described in [64]. This approach leads to a *quadratic objective function*.

The optimization algorithm used will depend on the kind of objective function, the number of variables, and the additional constraints that may be added (e.g., positiveness of the beamlets or maximum beamlet value). For quadratic objective functions, the most common and the fastest optimization algorithms are based on a gradient search. *Gradient optimization methods* change the value of one beamlet or a small group of beamlets, then the dose distribution is computed multiplying the dose matrix (defined in previous subsection 1.3) by the beamlets serialized in a vector, and the score for the new solution is obtained. If the score is better than the former one, the changes are accepted; if not, the changes are rejected. This search continues in an iterative process until no improvement is found. At this point, it is said that the solution has *converged* and it is assumed that the *optimum solution* has been found.

An extended introduction to this topic with relevant bibliography can be found in [28, pp 888-892], in [32, pp 4-16-4-26] and in [1, pp R390-R394]. Finally, a very good review of several optimization methods is made in [54].

## 1.5 MLC SEGMENTATION

The last step for generating a SMLC plan in the FMO approach is the decomposition or segmentation of the fluence map (sometimes called *fluence matrix*) in a set of apertures with an associated beam-on time. An example of fluence map and its segmentation can be seen in figure 4.

The MLC segmentation problem is a combinatorial problem. The optimal solution minimizes the whole treatment time and this solution is usually not unique. Two relevant factors on this treatment time are the total exposure or beam-on time, i.e. the total number of monitor units (TNMU), and the MLC leaves motion time, which is directly related to the number of segments (NS). The computation of MLC segmentations with the minimum NS was proved in [38] that belongs to the complexity class NP-complete of decision problems.

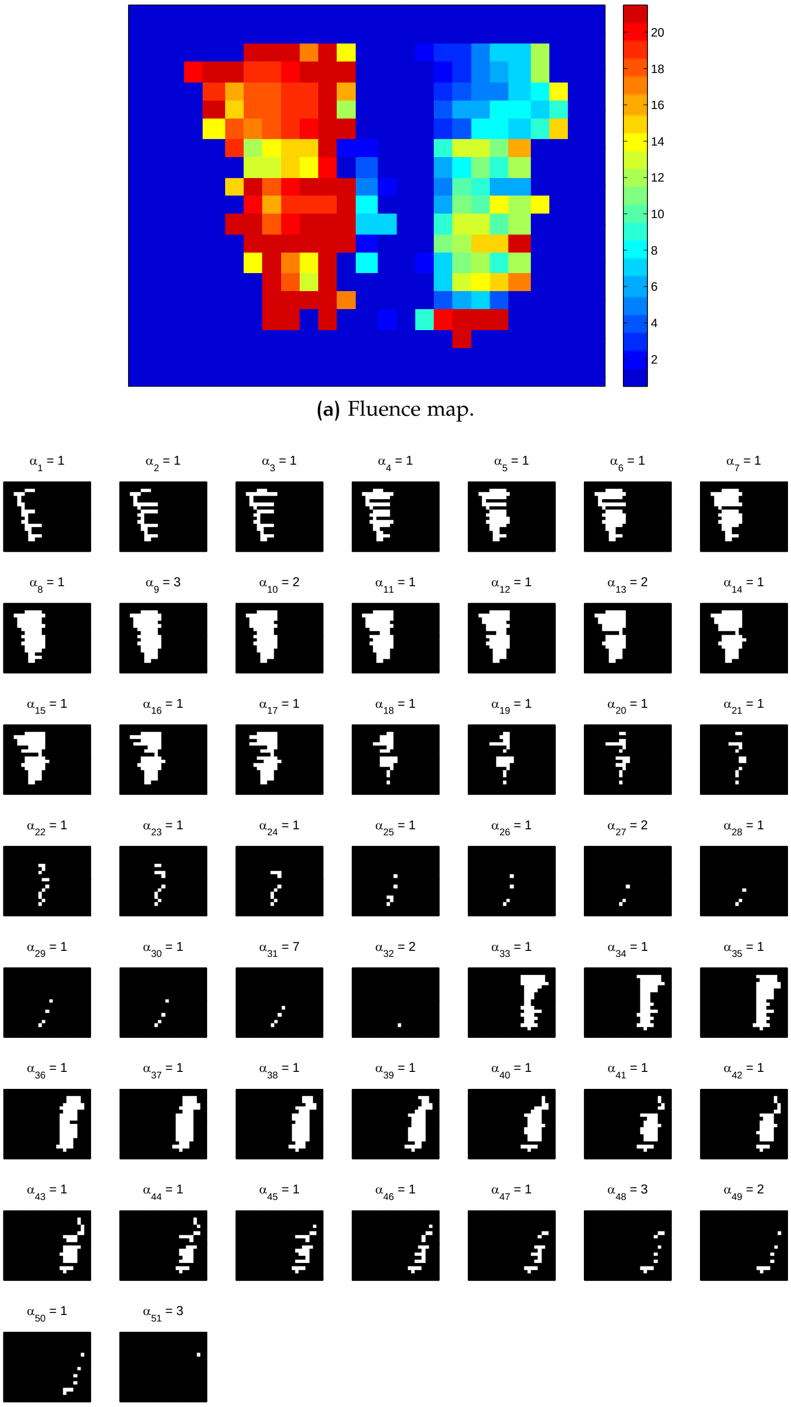
There are two main approaches for solving NP-complete problems: the exact approach, that guarantees to get the optimal solution, and the non-exact approach, that searches for a suboptimal solution [26], instead. The exact approach is only useful for very small inputs, because the computational complexity grows exponentially with the input size. In this thesis, the intensity map segmentation techniques described were developed having in mind clinical IMRT applications. Hence, the non-exact approach was selected and the proposed methods are and will be compared with other non-exact methods.

*The non-exact solutions are achieved by approximation, randomization, restriction, parameterization, or heuristic methods.*

The sequencing or temporal ordering of segments is highly relevant. For a given set of segments, it can be proofed that the total leaf motion time is minimized if the sequence of segments is arranged, so as the MLC leaves move in one direction (for instance, from left to right) [57]; if a segmentation can meet this criterion, it is called a unidirectional segmentation. It is important to mention that segmentations obtained with the optimal TNMU or NS are not necessarily unidirectional.

The MLC segmentation methods for the step-and-shoot technique can be either unidirectional, such as [4, 9] and the rod pushing (RP) technique described in [57], or bidirectional, such as [18, 19, 25, 40, 57, 67]. Unidirectional segmentation methods are usually optimum, or close to the optimum, regarding the TNMU and they inherently minimize the leaf travel time. However, the NS is generally larger than in bidirectional methods. Therefore, the arranged set of apertures that is obtained minimizes the MLC leaf movements and the beam-on times, but the set has more apertures than necessary.

Chapters 3 to 5 are focused on the MLC segmentation problem. In particular, chapter 3 describes a new unidirectional segmentation algorithm with the particularity of being able to control the shape of the apertures in order to avoid irregular contours and more than one connected component. Chapter 4 deals with the reduction of the NS in unidirectional segmentations, so that its main disadvantage diminishes or disappears. Finally, chapter 5 introduces a method for post-processing unidirectional segmentations and generating a solution with *a priori* fixed number of apertures in order to have some degree of control over the NS as done in other SMLC approaches than the FMO such as the direct step-and-shot (DSS) methods.



**(a)** Fluence map.

**(b)** MLC segmentation.

**Figure 4:** Example of a unidirectional fluence map decomposition with 51 apertures. The example comes from an oropharynx cancer case beam. The $\alpha_x$ denotes the beam-on time associated to an aperture, where $x$ is the aperture number or, alternatively, the delivery order.

# 2 VOXEL PROJECTION

## Contents

## 2.1 ABSTRACT

Finding which patient voxels are radiated by a beamlet is crucial when computing the dose matrix in the fluence map optimization approach. In this chapter, a compute-by-drawing method is proposed that improves the current mathematical projection of the voxel center points by projecting the whole voxel instead. This new approach makes the resolution of beamlets and voxels independent. Thus, eliminating the restriction of a voxel size smaller than the beamlet width and height in order to ensure that every beamlet at least radiates one voxel.

## 2.2 INTRODUCTION

During the optimization of beamlet intensities in the FMO approach, the search of the solution is based on whether the changes performed in the intensity values increase or decrease the objective function value obtained from the dose distribution. Therefore, the dose calculation algorithm may be called thousands of times before the optimization is completed. Hence, a fast and relatively accurate modelling of the radiation is needed in order to speed up computations and provide the optimization algorithm with a realistic approximation to the actual dose distribution.

One of the most common algorithms used for dose calculation in IMRT is the finite size pencil beam method [5, 6, 10, 15, 33, 34, 48]. This model is based on the assumption that a broader beam can be divided into identically and finitely sized beamlets (in opposition to an infinitesimally narrow ray) and that the dose for a given point can be calculated as the sum of contributions of all the beamlets to that point. The basic idea consists in casting a ray for each beamlet that pass through the patient anatomy. When a voxel is hit or is relatively close to the ray, the radiological depth [56] for the perpendicular projection to the ray of the its center point is computed. Then, a 2D pencil beam kernel, which characterises the linear accelerator output at that radiological depth, is placed with the origin in the projected point and perpendicular to the ray, thus containing the voxel center. With this configuration, it is computed the dose deposition coefficient for the beamlet to this particular voxel. Finally, the coefficients obtained for a ray will be placed at the corresponding column in the dose matrix, so the contribution of that beamlet to the patient dose distribution can be obtained just multiplying the column by the assigned fluence or intensity.

*The origin of all rays is the source point and each ray passes through the center point of a different beamlet.*

*The dose deposition coefficients are defined for 1 fluence or intensity unit.*

In this chapter, we will focus on how to find which voxels are directly affected by the radiation of a beamlet. This relationship can be formulated as a raytracing [29] or as a projection [65]. The raytracing is the direct and intuitive method that represents the radiation as a ray traversing the patient. The main drawback of this approach is that some voxels may not be in the path of any ray, so they will not be included in the optimization [65, pp N162-N163]. This problem can be solved defining a volume of interest along the ray as done in [29]. However, this solution is slow if it is executed outside a graphics processing unit (GPU) [29, p 6294]. The projection is the inverse method. It defines a straight line that joins the voxel center with the source point and provides which is the traversed beamlet, thus establishing a one-to-n relationship (a beamlet can radiate many voxels, but a voxel can be only radiated by one beamlet). In this case, the problem is the relative size between beamlets and voxels. In general, the beamlet size should be as small as possible for making the conforming of the dose to the tumour easier to achieve. On the other hand, the voxel size should be consistent with the image resolution in order to properly assign the density to each voxel for computing the radiological path. As a consequence, if the voxel size is big enough to be directly affected by more than one beamlet, this approach will fail to model the radiation. In conclusion, the size of beamlets and voxels is bound. The solution to this problem is to project the whole voxel to the beamlet plane, but it may slow down the computations to project each voxel vertex, to obtain the polygon that encloses all the points with a 2D convex hull algorithm, and finally to test the polygon against the beamlet grid for deciding which are the activated beamlets.

We propose to translate the mathematical projection method to a compute-by-drawing approach. The beamlet plane at the isocenter can be moved outside the patient to the MLC plane and converted into a screen, the source point can be exchanged by a camera, and the projection can be performed by drawing voxels using OpenGL[1] as if they were objects in a scene. The advantages of this approach are that: (1) every voxel is assigned to at least one beamlet, (2) the resolution of beamlets and voxels is independent if the drawing is done properly, (3) it is faster than a central processing unit (CPU) implementation and (4) it is possible to achieve sub-beamlet precision. We choose OpenGL as computer graphics API[2] because it is cross-platform, widely used, hardware-independent, practically supported by any graphic card on the market, and the documentation and examples that can be found on books and the Internet[3] are extensive.

The rest of the chapter is organised as follows. In section 2.3, it is described the method, the OpenGL extensions used, some possible improvements of the algorithm, and the data and experimental setup. The results obtained and a comparative with the original projection method is presented in section 2.4. Finally, the discussion and conclusions of the results are provided in sections 2.5 and 2.6.

## 2.3 METHOD

Our implementation of the radiation model is divided into two main phases. First, a macropencil kernel that contains the dose deposited by the central beamlet and their neighbours for one fluence unit at a given radiological depth is generated. The value assigned to each beamlet in the macropencil kernel is the dose delivered in the projected beamlet area. This dose is obtained by projecting the beamlet outline on a perpendicular plane located at a given depth, and then the corresponding finite size pencil beam kernel to that depth is integrated inside the outline area. The finite size pencil beam kernels are only available for a limited set of depths, and therefore so are the macropencil beam kernels. As a consequence, it is necessary to generate new macropencil beam kernels by interpolation to uniformly sample the depth axis at regular intervals. Second, the voxel projection is performed in order to obtain which is the beamlet that radiates a given voxel. Then, in order to compute the voxel radiological depth, it is chosen the appropriate macropencil beam kernel, which defines the contribution of the beamlet that directly radiates the voxel and also the contribution

*Note that the area will increase as the depth does.*

[1] OpenGL is a registered trademark of Silicon Graphics International.

[2] Application programming interface.

[3] Visit http://www.songho.ca/opengl/index.html for an OpenGL tutorial with a collection of helpful examples including source code.

of the surrounding beamlets, and the corresponding dose deposition coefficients for that voxel are updated.

The mathematical projection of a voxel is represented in figure 5. The algorithm is composed of the following steps: (1) the plane perpendicular to the current beam that contains the isocenter, from now on the isocenter plane, is computed, (2) the bounding box of each target volume (but, no OARs) is projected on the isocenter plane for obtaining the grid size and location, (3) the beamlet grid is generated as a set of points, (4) for each voxel of each organ is computed (4.a) its radiological depth and (4.b) the projection of its center on the isocenter plane, (5) the beamlet that contains the projected point is searched.

*Note that the isocenter and the center of the grid doesn't need to be the same point*

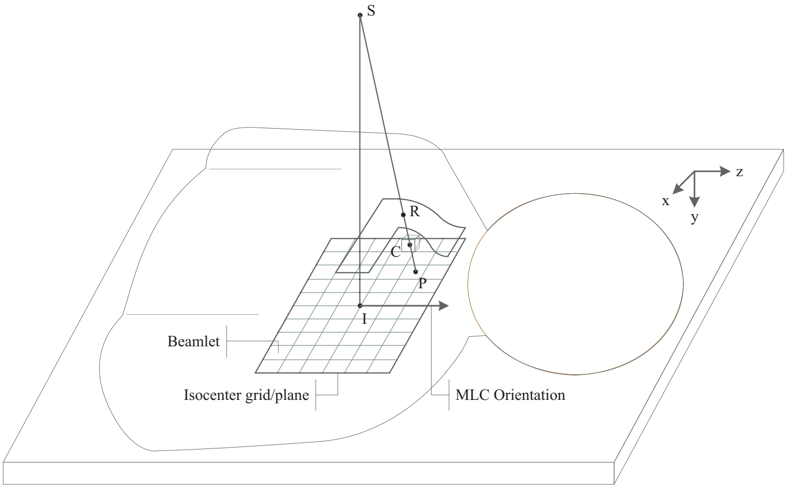


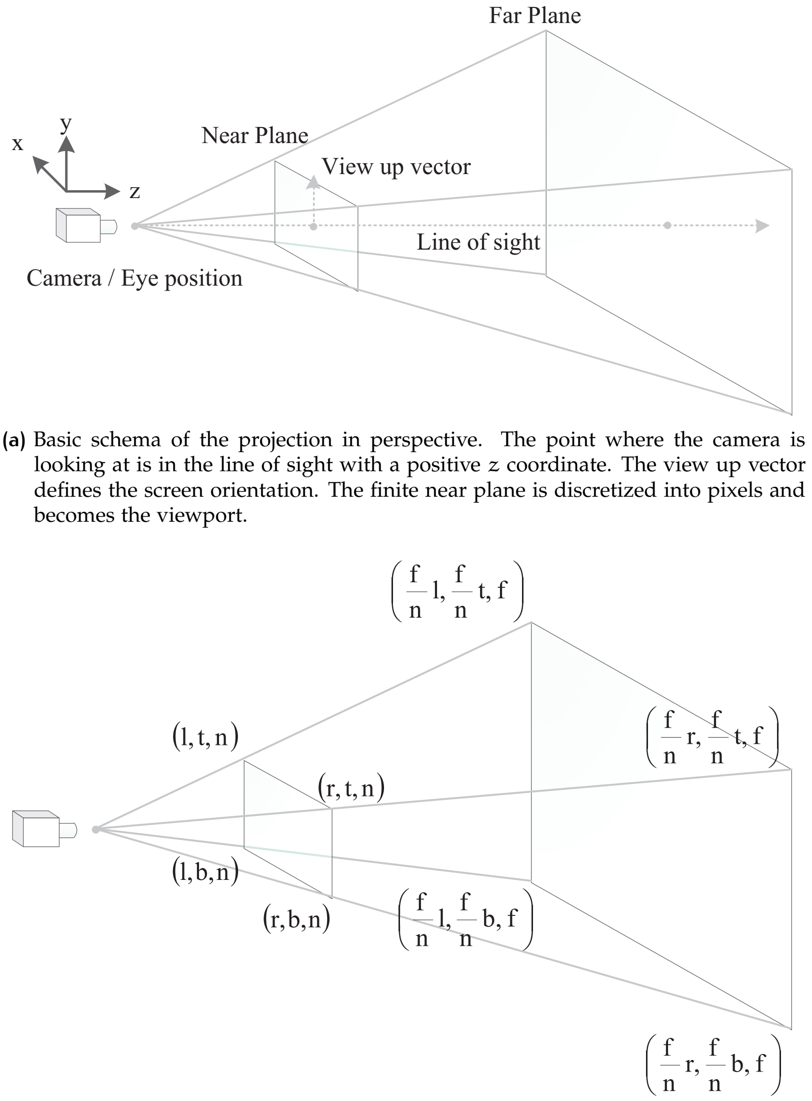
**Figure 5:** A schematic diagram of the mathematical voxel projection. The point S is the radiation source, I denotes de isocenter, C is the voxel center, which is projected in the isocenter plane as P, and R is the point where the ray intersect for the first time with the patient skin. The radiological depth is computed using the segment $\overline{RC}$.

### 2.3.1 Drawing with OpenGL

The mathematical projection problem introduced in steps (4.b and 5) can be translated to a drawing problem and solved in a computer graphic card by converting the MLC into a screen, since the hardware configuration, the discretization of the MLC field of view, and the radiation spreading has a lot of analogies with how to setup the visualization and draw in OpenGL. In addition, as a consequence of solving the problem by drawing, the generation of the beamlet grid in step 3 is no longer necessary.

The setup of the visualization in OpenGL has three parts. Firstly, it is needed to define the position and orientation of the camera and where is looking at. Secondly, the view frustum is defined for deciding which objects or portions of objects will be clipped out and for determining how the 3D scene is projected on the screen. This frustum is formulated in computer graphics as the *projection matrix*. In this case, the frustum is composed of six planes forming a pyramid that lies between two parallel planes cutting it. This pyramidal frustum is creating a projection in perspective. There are other kind of projections, such as the parallel that uses a rectangular prism instead of a pyramid. However, the projection in perspective draws the scene on the screen in the same way that the radiation would spread, i.e., the light ray for each pixel is equivalent to the radiation ray for a beamlet. OpenGL provides the function *glFrustum()* to produce a perspective projection, and it requires six parameters to specify the five coordinates needed for defining the near plane: left, right, bottom, top, near distance and one more coordinate for the far distance. This two steps are represented in figure 6a, and figure 6b shows how the finite planes are computed. Finally, the last step is setting the viewport resolution.

*The perspective projection is the most common type of projection for any application that requires the visualization of triangle meshes, such as videogames or finite element analysis.*


Far Plane
Near Plane
x
y
z
View up vector
Line of sight
Camera / Eye position
$\left(\frac{f}{n}l, \frac{f}{n}t, f\right)$
$(l, t, n)$
$\left(\frac{f}{n}r, \frac{f}{n}t, f\right)$
$(r, t, n)$
$(l, b, n)$
$(r, b, n)$
$\left(\frac{f}{n}l, \frac{f}{n}b, f\right)$
$\left(\frac{f}{n}r, \frac{f}{n}b, f\right)$

**(a)** Basic schema of the projection in perspective. The point where the camera is looking at is in the line of sight with a positive *z* coordinate. The view up vector defines the screen orientation. The finite near plane is discretized into pixels and becomes the viewport.

**(b)** The 8 points that define the near and far planes.

**Figure 6:** OpenGL view frustum.

Taking the entities involved in the OpenGL setup and its configuration into account, the mathematical projection and the drawing problem are analogous if: (1) the source point and the MLC orientation are exchanged by the camera position and orientation, respectively, (2) the isocenter is defined as the point where the camera is looking at, (3) the finite isocenter plane is translated to the MLC plane (or any other one outside the patient) and scaled preserving the aspect ratio, and (4) the MLC resolution is set to the viewport resolution. The last two steps transform beamlets into pixels. The third step implicitly defines five planes for the frustum: the near plane where the viewport/screen is, and four planes defining the body of the pyramid. The last plane for the bottom of the frustum should be set far enough to do not discard any voxel for its $z$ coordinate. Figure 7 represents the projection of voxels as a drawing problem in our case.

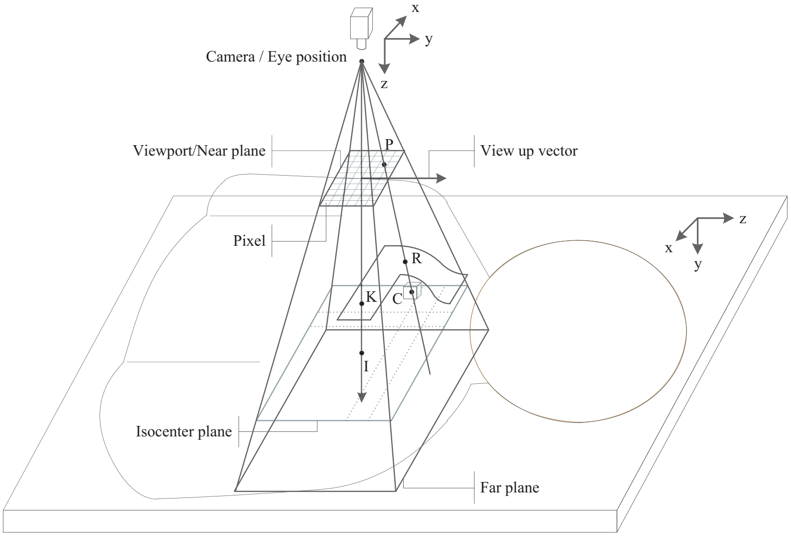


**Figure 7:** A schematic diagram of the voxel drawing with OpenGL. The isocenter plane is only showed for illustrative purposes. K is point where the line of sight intersects with the patient's skin for the first time.

If figures 5 and 7 are compared, it is straightforward to see that only the translation of the grid and the frustum need to be calculated, because the position and orientation of the camera and the point to look at remain the same. It is important to mention that the former points and orientation are set in world coordinates (in our case, patient coordinates), whereas the frustum is defined in camera coordinates.

### *Near and far planes setup*

The near plane, usually denoted $ZNear$ in computer graphics, has to be outside the patient anatomy, because anything between the camera and this plane will be discarded when drawing. Therefore, adopting the notation of figures 5 and 7, if $SK$ denotes the $\overline{SK}$ segment length and therefore the distance from the source to the skin for the main axis of the beam, $ZNear$ can be defined as

$$ZNear = SK/2 \tag{2.1}$$

Then, the finite region of this plane is defined by the top, left, bottom and right values. This values are proportional to the size of the isocenter grid. Let $SI$ be the distance from the source point to the isocenter and $IT$, $IL$, $IB$ and $IR$ be the distance from the isocenter to the top, left, bottom and right margins of the grid. Therefore,

$$top = ZNear \cdot (IT/SI) \tag{2.2}$$

$$left = ZNear \cdot (IL/SI) \tag{2.3}$$

$$bottom = ZNear \cdot (IB/SI) \tag{2.4}$$

$$right = ZNear \cdot (IR/SI) \tag{2.5}$$

The only parameter for the glFrustum function that remains to be computed is the far plane, denoted $ZFar$, but it can be assigned as twice the distance from the source to the isocenter

$$ZFar = SI * 2 \tag{2.6}$$

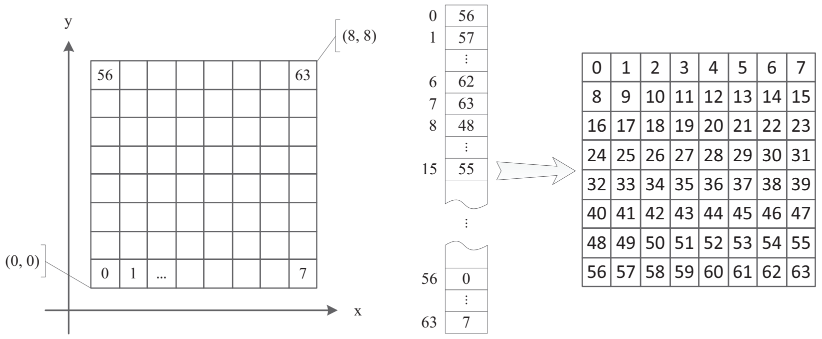
to be sure that the far plane does not cull any voxel. The proposed values for $ZNear$ and $ZFar$ are very conservative. In computer graphics, these values are intended to enclose the geometry of interest as close as possible in order to avoid problems with the z-buffering that decides which elements of a rendered scene are visible, and which are hidden. If the distance between both planes is relatively huge compared to the distance between two surfaces in the scene, both surfaces are rendered as if they were at the same distance from the screen, and the final rendering will contain artifacts in the affected pixels. This effect is called 'z-fighting'. However, we will draw voxel by voxel aiming to know which pixels/beamlets are activated with no visualization. Therefore, the z-buffer problem is not relevant in this case.

*Viewport setup*

Once the frustum is defined, the finite near plane is discretized to become the viewport. If the width and height of the isocenter grid is set to the viewport, each pixel will exactly correspond to one beamlet as shown in figure 7. However, the window coordinates for the viewport are different from the usual way of numbering the beamlets that one would expect. Therefore, the pixel and beamlet numbering will not match. As a consequence, it is needed some kind of mechanism for translating between pixels and beamlets positions.

*We will not enter into the details of the OpenGL vertex transformation pipeline, that is, how the different coordinate systems are handled and which are the transformations needed to go from the world coordinates to the windows coordinates.*

In OpenGL, the origin of coordinates for a screen is placed in the lower left corner, as shown in figure 8a that illustrates this convention with a viewport of 8x8 pixels. In contrast, the expected or typical numbering of beamlets would start in the higher left corner as shown in the beamlet grid of figure 8b. Thus, the positions are flipped in the up-down direction, that is, about a horizontal axis. We implemented a translation table that will enable the algorithm to directly find which is the corresponding beamlet for a given pixel when iterating over the viewport pixel buffer or *framebuffer*. This table is a vector that concatenates by rows the beamlet numbering from down to up, as can be seen in the vector of figure 8b. For example, the first pixel in the framebuffer is translated by the first position in the vector as the beamlet number 56 in the example of figure 8b.



|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 |
| 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 |
| 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 |
| 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 |
| 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 |

**(a)** OpenGL window coordinates convention. The coordinate system origin is placed in the lower left corner. The numbers represent the pixel order in the framebuffer where it is drawn.

**(b)** On the left, the table that, given the position of a pixel in the framebuffer, returns its associated beamlet. On the right, the pixel to beamlet correspondence that it is representing the table.

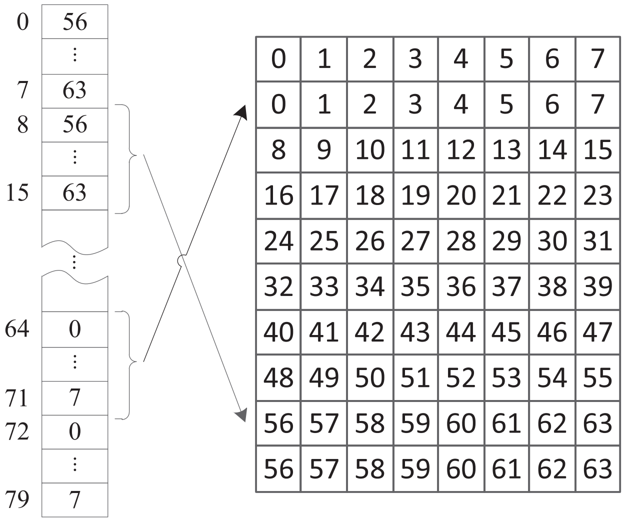
**Figure 8:** OpenGL window coordinates, pixel and beamlet numbering, and the translation from pixel to beamlet numbering.

The translation table makes converting on-the-fly pixel to beamlet positions possible, but it also allows to introduce in the beamlet emulation two additional features. First, the proposed solution supposes that all the MLC leaves have the same size, but there are some MLCs whose first and last leaf are wider than the rest, e.g., the regular size is one centimeter and the first and last leaves are two centimeters wide. In this case, it is possible to find the greatest common divisor, denoted $gcd$, of the different sizes and represent each row of beamlets with $leaf\_width/gcd$ pixels. In the former example, the first and last leaves are represented by two rows instead of one, which is illustrated in figure 9.

*The MLC can move leaves in a continuous way along the $x$ axis.*

Second, the leaf width determines the resolution for the $y$ axis, and only the resolution on the $x$ axis can be modified by setting a $\Delta x$ in order to increase the horizontal resolution and improve the conforming of the dose distribution to the target volume. The typical $y$ resolution is 10 millimeters, whereas the $x$ resolution usually varies from 1 to 10 millimeters. There is a trade-off, the better the resolution, the more conformed the dose distribution to the target volume can be, but the more complex the treatment plan becomes [69]. Generally, the resolution in the $x$ axis ranges from 4 to 8 mm. Thus, the grid discretization is coarse and this fact makes more difficult to compute which beamlets and in which proportion are activated by a voxel. This topic will be extended in the following subsection 2.3.2, but the viewport resolution can be modified without restrictions. Using the translation table, it is possible to achieve sub-beamlet precision.

The width of one beamlet is divided by its height and then the result is rounded to the nearest integer for deciding if the beamlet will be represented by more than one pixel. For example, if the width is five mm, and the height is ten, the beamlet can be split into two square sub-beamlets of five by five mm, aiming to use pixels as squared as possible. This solution is implemented building the translation table accounting for how many sub-beamlets represent a beamlet and in-



| | |
|---|---|
| 0 | 56 |
| | ⋮ |
| 7 | 63 |
| 8 | 56 |
| | ⋮ |
| 15 | 63 |
| | |
| | ⋮ |
| | |
| 64 | 0 |
| | ⋮ |
| 71 | 7 |
| 72 | 0 |
| | ⋮ |
| 79 | 7 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 |
| 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 |
| 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 |
| 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 |
| 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 |
| 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 |

**Figure 9:** Translation table and pixel to beamlet correspondence when the first and last leaves are two times bigger than the rest of leaves. Note that in this case 80 pixels are used for representing 64 beamlets, the indexing is accordingly updated from the former figure 8b, and the correspondence between pixels and beamlets is not anymore a one to one relationship.

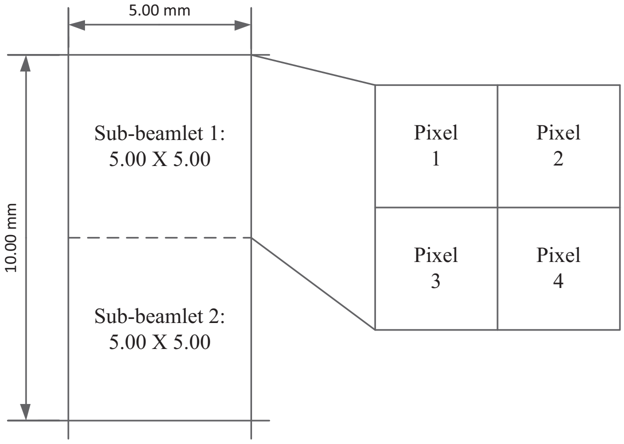


**Figure 10:** An anisotropic beamlet of 5 mm wide and 10 mm height is split into two sub-beamlets. In addition, each sub-beamlet is composed of four pixels to increase the accuracy when drawing. As a result, the original beamlet is represented with eight pixels in a 2 × 4 configuration.

creasing the number of rows in the screen. This enhancement can be applied recursively by specifying that each sub-beamlet is, at the same time, represented by more than one pixel. An example of this representation is shown in figure 10.

### 2.3.2 What and how to draw?

So far, we have configured OpenGL for drawing on the screen, and now we will tackle the problem of what is drawn and how to do it.

A voxel is a rectangular prism, where each axis may have a different size. GLUT[4] and its completely open-source alternative freeGLUT[5] are a window system independent toolkits for writing OpenGL programs that has some primitives for generating simple objects. The *glutSolidCube()* function of these toolkits can be extended for drawing rectangular prisms of any shape instead of cubes.

Given the $x$, $y$ and $z$ dimensions of a voxel and its center point. If the center point is drawn on the screen, it is obtained the same result than the original mathematical projection. Thus, this configuration allows us to validate the OpenGL implementation and test if the results in terms of accuracy are the same. Alternatively, if each face of the voxel is drawn on the screen, all the beamlets where the voxel is projected will be activated.

*Actually, OpenGL splits quads into triangles.*

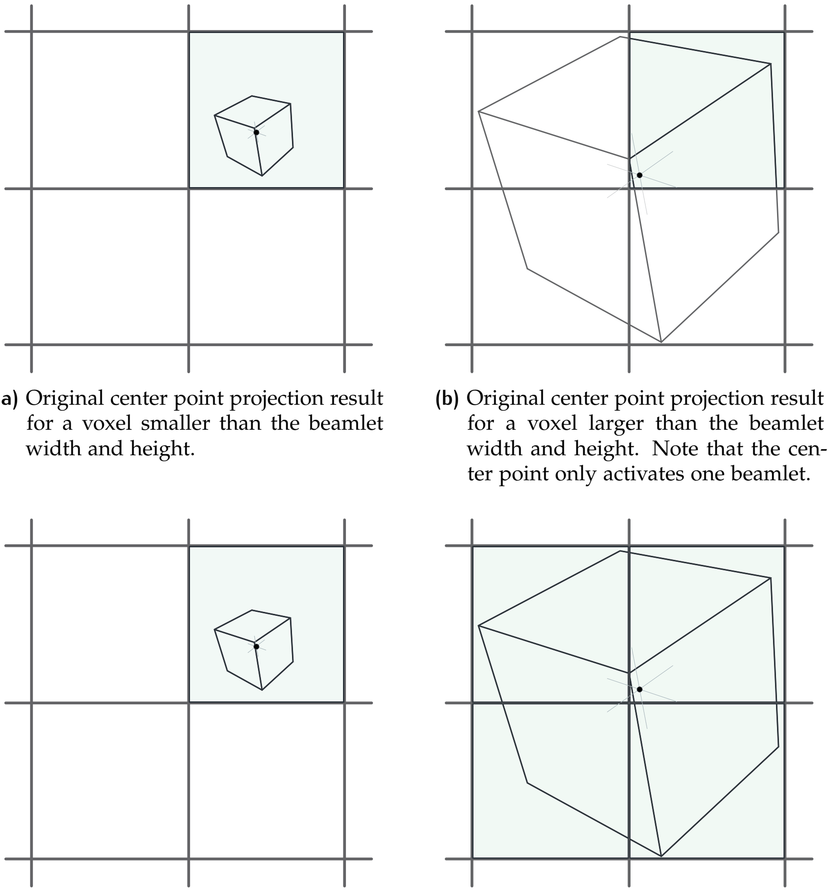
OpenGL is able to draw directly *quads* (finite planes with a rectangular shape, that is, one voxel face), triangles, points and lines. However, there are some considerations to take into account when drawing with OpenGL: (1) a point is a 0-dimensional object, meaning that it has neither volume, area, length, nor any other higher dimensional analogue, so OpenGL has to display such an entity lightning the pixel where the point is projected, (2) anything different from a point and smaller than one pixel could be discarded, because if the drawing of an object does not cover half of the area of a pixel, the pixel is not updated, and (3) any solid object or line has a continuous shape that is drawn on a discrete screen formed by pixels, therefore the final shape on the screen is an approximation.

*OpenGL has built-in functions for doing the approximations. The typical example of this problem is how to draw an oblique or a curved line.*

In order to make the algorithm independent of the relative size between pixels and voxels, all the faces of the voxel and also its center point are drawn. If the voxel is smaller than one pixel, drawing its center will activate the pixel where it is projected and if the voxel is bigger than one pixel, the drawing of the whole object will provide a set of pixels. Figure 11 shows an example where the original projection and the new drawing methods are compared in both situations. However, the idealized scenario of figure 11 is not realistic, because beamlets are usually very anisotropic as explained in previous subsection 2.3.1. As a consequence, the resolution of the screen plays an important role in the final results, and the coarse resolution of the isocenter grid may cause unexpected results in the compute-by-drawing solution if this factor is not taken into account. For solving this problem, the translation table can be used to increase the screen resolution and achieve sub-beamlet precision, as explained in previously in subsection 2.3.1,

[4] http://www.opengl.org/resources/libraries/glut
[5] http://freeglut.sourceforge.net

obtaining a finer grid where only a part of the original beamlet is highlighted by the voxel like in figure 12, but the whole beamlet will be activated.



(a) Original center point projection result for a voxel smaller than the beamlet width and height.

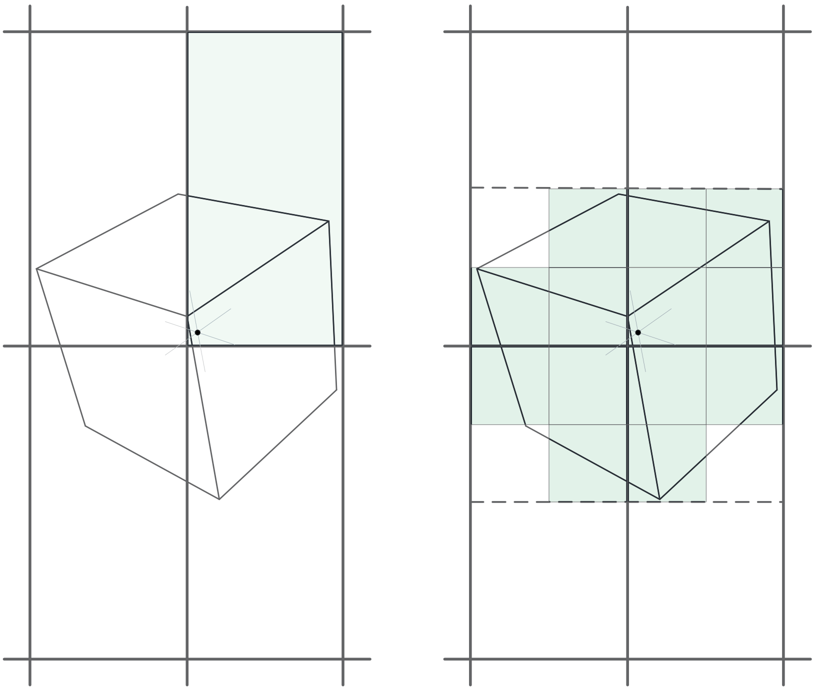
(b) Original center point projection result for a voxel larger than the beamlet width and height. Note that the center point only activates one beamlet.

(c) New drawing method result for a voxel smaller than the beamlet width and height. The center point drawing activates one beamlet, but the complete voxel drawing does not activate the beamlet because the object does not cover enough area of the pixel.

(d) New drawing method result for a voxel larger than the beamlet width and height. In this case, the drawing of the whole voxel activates the four beamlets.

**Figure 11:** Example of voxel projection with four isotropic beamlets where each beamlet corresponds to one pixel. The activated beamlets are shaded in light grey.

The final algorithm has the following steps for each beam: (1) the translation table is built taking into account the original isocenter grid resolution, leaf sizes, and sub-beamlet precision, (2) for each organ, the voxels are drawn in white one by one in the framebuffer, (3) after drawing a voxel, the framebuffer is retrieved from the graphic card and then it is initialised for the next voxel as a black screen, (4) it is iterated over the retrieved buffer for finding the activated pixels, (5) when a position in the buffer is different from the background color, it is



(a) One beamlet corresponds to one pixel. In this case, the original and the drawing methods provide the same result. The reason is that the object does not cover enough area of the pixels when it is drawn for activating them.

(b) Each beamlet is split into two subbeamlets that are composed of four pixels, as shown in figure 10. The object in this case is drawn due to a better resolution, and the four beamlets are activated in the drawing method.

**Figure 12:** Example of voxel projection on four anisotropic beamlets. The activated pixels are shaded in light grey.

checked in the translation table which is the corresponding beamlet, and the list of voxels that are projected over this beamlet is updated. At the end of this process a vector of beamlets, where each position has the list of hit voxels, is used to build the dose matrix.

### 2.3.3 OpenGL extensions

*A graphic card implements a subset of these functions. This subset determines to which OpenGL revisions the card is fully or partially compliant.*

*Screen tearing is a visual artifact in video where information from two or more different frames is shown in a single screen draw.*

OpenGL has many revisions that introduced progressively new functions, called *extensions*. The frame buffer object (FBO) is a non-displayable rendering destination that provides an efficient way of offscreen and texture rendering. This extension can be used to accelerate the drawing of voxels, since it is faster than drawing on the default display framebuffer. Nevertheless, if the graphic card does not support the creation of FBOs, the display framebuffer can be used disabling the vertical synchronization. Usually, the graphic card drawing and the monitor refresh are synchronized to avoid *tearing* effects. If the typical refresh rate in a LCD is 60Hz, this means that 60 frames are draw and displayed every second, and therefore only 60 voxels can be processed by second. If the vertical synchronization is switched off, this limit disappears and thousands of voxels can be rendered in a second.

### 2.3.4 Improvements

There are two OpenGL extensions that can increase the performance of the proposed algorithm. First, the Vertex Buffer Object (or the older display list) allows to efficiently compile and store the organ vertex data (quads) in the graphic card only once, and share this information between beams, instead of resending for each beam all the data. Second, the Pixel Buffer Object store pixel data in a OpenGL controlled memory and allows asynchronous pixel transfer through direct memory access, without involving CPU cycles, from/to the GPU. Using this extension and creating two buffers, it is possible draw in one of them and read it without blocking the OpenGL pipeline, while swapping to the other buffer and the next drawing is performed in parallel with the reading.

Another improvement would be the use of OpenCL (Open Computing Language), which is a framework for writing programs that execute across heterogeneous platforms consisting of CPUs, GPUs, and other processors. This framework is defined and developed by the same group of companies that created OpenGL, and both libraries can be interconnected. With OpenCL, the processing of the buffer could be done in the graphic card taking advantage of the inherent parallelism of GPUs and without transferring the buffer to the client/application side after drawing each voxel. Consequently, the performance of the method will be drastically increased.

Finally, the proposed algorithm could be extended in order to consider and account for the partial activation of beamlets in the radiation model, with a more accurate method than the macropencil. The partial activation of a beamlet, when it is represented by more than one pixel, can be computed as follows. If it is the first time that a voxel is hit by a beamlet, a new entry at the end of the hit voxel list for that beamlet is created, and it is stored the voxel number and the area covered, which is set to one divided by the number of pixels that represent the beamlet. Each subsequent hit of the same voxel by the same beamlet is detected because the entry already exist on the last position of the list and only the covered area is updated. As an example, in figure 12b the area activated by the voxel in the first beamlet is 3/8.

### 2.3.5 Data and experimental setup

The experiments reported in this chapter were performed using two clinical cases planned with PCRT 3D® (Técnicas Radiofísicas S.L., C/ Gil de Jasa, 18E, 50006 Zaragoza, Spain, www.trf.es) treatment planning system. The personal computer where the time measurements were obtained had an AMD processor Athlon 64 6400+ X2 Black Edition (dual core at 3215MHz with 2MB of cache), 8GB of DDR2 RAM at 800MHz with CL5 in dual channel mode, and an NVidia graphic card Geforce 8600GT with 256MB of GDDR3 RAM. Also, three graphic cards were used during the tests, an ATI Mobility Radeon X1700 with 256MB DDR2 RAM, an integrated Intel 855GM graphics chipset with 64MB of DDR, and a NVidia Mobile Geforce 8400GS with 64MB of GDDR3.

Detailed results achieved by the method are presented for a prostate cancer radiated from five coplanar and equiangular beams: 36°, 108°, 180°, 252° and 324°, in a 72 Gy plan, where the dose-volume constraint used for the rectum and bladder was 70% of the volume receives $\leqslant$ 40% of the goal dose, and for a larynx cancer planned using seven coplanar, but not equiangular, beams: 20°, 60°, 100°, 180°, 260°, 300° and 340°. The latter case has four target volumes; the prescribed dose for the gross disease region (PTVgr) was 74 Gy and for the elective nodal regions (CTVel) were 66 Gy, 56 Gy, and 50 Gy. The dose-volume constraint for the spinal cord was maximum dose $\leqslant$ 45 Gy, and the constraint for both parotids was 50% of the volume receives $\leqslant$ 40% of the prescribed dose to the PTVgr.

Firstly, the prostate case is used to illustrate how the original projection method fails to represent the relationship between beamlets and voxels with a beamlet size of 10 mm height and 4 mm width and a voxel size of 6 mm in each dimension. This experiment was done without including the macropencil, for detecting which are the actual beamlets activated by the voxels. As a consequence of this problem, the original projection method has been used in the treatment planning system imposing the restriction that every organ volume has to be discretized with a voxel size at least half of the beamlet width. As a result, every beamlet will radiate at least a voxel center point, but the amount of voxels is drastically increased. Thus, unnecessarily slowing down the optimization process due to the amount of computations needed to obtain the dose distribution for those voxels.

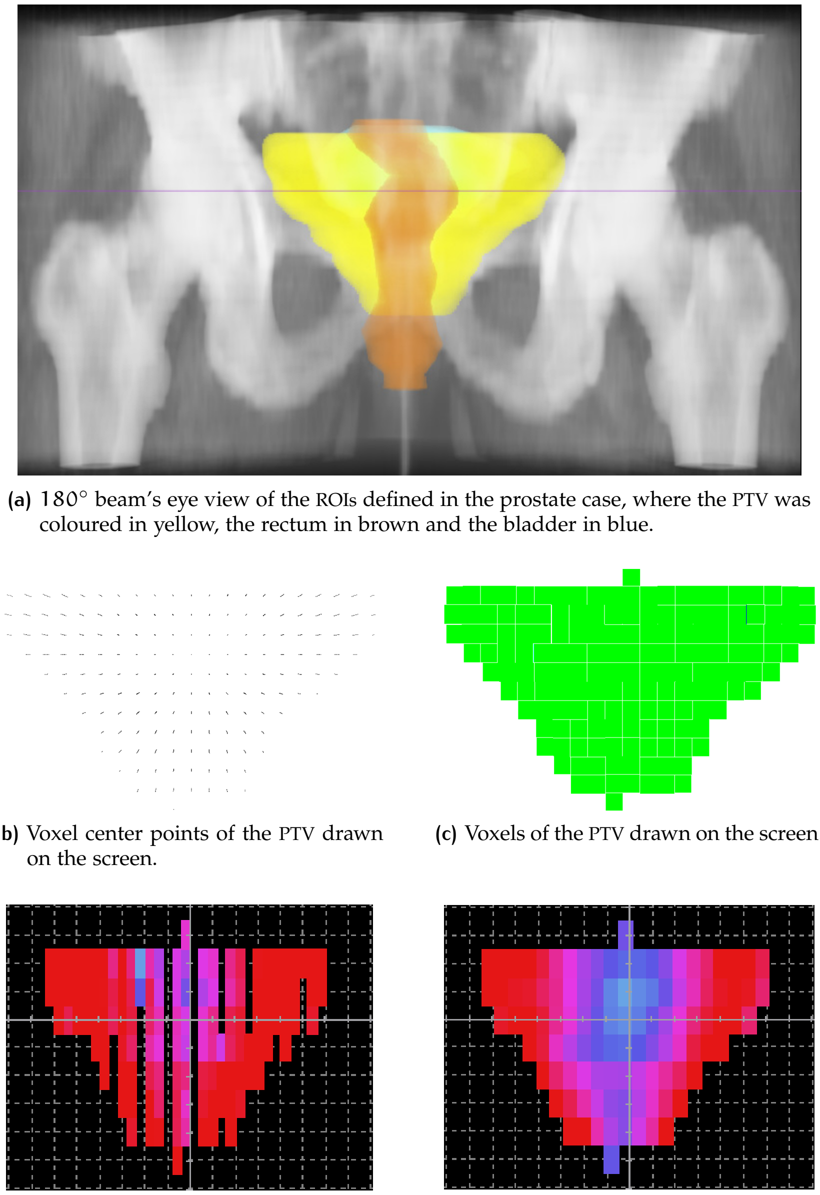
*Note that the beamlet height is usually 10 mm and only the width varies, as explained previously in section2.3.2*

Secondly, assuming the restriction in the voxel size imposed by the original projection method, the prostate case organs were discretized with a voxel size of 3 mm in each dimension, the beamlet width was set to 6 mm, and the drawing was configured so that every beamlet was represented by 2 sub-beamlets of 4 pixels each one, i.e., 8 pixel in total. The larynx target volumes were discretized with a voxel size of 2.75 mm, the OARs with a voxel size of 2.0 mm and the beamlet width was set to 8 mm. In this case, every beamlet was represented by 4 pixels. Under these conditions, the original mathematical projection and the drawing methods were compared in terms of accuracy by only drawing the voxel center on the screen in the latter case.

Finally, the computation times of both methods were also compared, including the voxel center and the complete voxel drawing approaches in the proposed method.

## 2.4 RESULTS

The 180° beam of the prostate case was used to show that the original center point projection technique is not suitable for building the relationship between beamlets and voxels when the voxel size is bigger than the beamlet height or width. The beam's eye view of the organs is provided in figure 13a. The voxel center points projected for the target volume can be seen in figure 13b, which can be compared with the complete voxel projection in figure 13c. The intensity maps obtained after the optimization with the original and the new projection methods are displayed in figures 13d and 13e, respectively.



(a) 180° beam's eye view of the ROIs defined in the prostate case, where the PTV was coloured in yellow, the rectum in brown and the bladder in blue.

(b) Voxel center points of the PTV drawn on the screen.

(c) Voxels of the PTV drawn on the screen.

(d) Fluence matrix obtained after the optimization step with the dose matrix obtained projecting points and without using the macropencil. The stripes of deactivated beamlets show that the relationship between beamlets and voxels is incomplete.

(e) Fluence matrix obtained after the optimization step with the dose matrix obtained projecting voxels and without using the macropencil. In this case, stripes of deactivated beamlets were not found.

Figure 13: Beam's eye view, data to project, and final results for the 180° beam of the prostate case.

The OpenGL projection method was set to draw only the voxel center points for the prostate and larynx case organs in order to test the correctness of the implementation and the accuracy of the drawing method against the mathematical projection. The results of comparing the beamlet to voxel assignations between both methods are provided in table 1, where the differences between the original projection and two graphic cards, one from the brand NVidia and another from ATI, can be seen. In addition, an Intel graphic card was also tested, yielding the same results than the NVidia card, and therefore they are not reported for conciseness. Finally, a different model of NVidia graphic card (8400GS) with a newer driver was tested to see if the card model or the driver version may influence the results. The experiment yielded exactly the same results than the original NVidia card.

A performance comparison between the mathematical projection and the drawing solution is presented in table 2, where the computation times for both methods can be seen. These results are only an example, because they are strongly dependent on the harware where the experiment is run. In this case, the NVidia 8600GT dates from 17 April 2007 and it is quite obsolete. This means that the measurements were done under unfavourable conditions, since almost any recent card may obtain better performance.

## 2.5 DISCUSSION

The results of the experiment with the prostate case, where the beamlet width is smaller than the voxel size, corroborate that the projection of the voxel center point, together with the assumption of one voxel is radiated by only one beamlet, is not enough to accurately model the radiation. The stripes of deactivated beamlets in figure 13d are caused by the alignment of the cloud of points with the beam's eye view. If this alignment is not present, the cloud of points will probably activate all the beamlets, but the relationship will be still wrong. When the macropencil is added, the inclusion of the beamlet neighbourhood in the relationship with a voxel completely masks this error and the only symptom is a strong disagreement between the dose computed using the dose matrix and the final dose computation algorithm for the treatment validation.

The differences found during the accuracy test, reported in table 1, were checked, and we found that the graphic card may assign the voxel to an adjacent beamlet when the voxel center point is really close to one of the four lines that defines the beamlet boundaries (distances $< 0.0001$mm). Actually, this is not especially relevant because any of the two beamlets would be a valid assignation under these conditions. The reason of this behaviour can be due to the computations with single-precision float numbers in the graphic card instead of the double-precision used in the original projection method. With this experiment, the implementation of the method is validated and it is also showed that the method does not depend on the driver version or the card model. Only the graphic card brand changes the results. Additionally, it is important to mention that if the number of pixels that represent a beamlet is increased, the number of differences decreases.

*The explanation for these results may be come from the fact that: (1) the OpenGL functions used in the implementation have been present in the library since the first version, and (2) each brand has its own implementation of the library.*

Table 1: Differences found in beamlet to voxel assignations between the original projection and the drawing method in two graphic cards (only the voxel center point was drawn). '# Voxels' is the total number of voxels in each organ. '# Diff' is the number of differences found. '%' shows the percentage of differences regarding the total.

| | | NVidia | | ATI | |
|---|---|---|---|---|---|
| Organ | # Voxels | # Diff | % | # Diff | % |
| | | **(a)** Prostate case. | | | |
| PTV | 40300 | 56 | 0.14 | 2202 | 5.46 |
| Rectum | 14019 | 26 | 0.19 | 885 | 6.31 |
| Bladder | 27263 | 59 | 0.22 | 1005 | 3.69 |
| Total | 81582 | 141 | 0.17 | 4092 | 5.02 |
| | | **(b)** Larynx case. | | | |
| PTVgr | 57225 | 199 | 0.35 | 3352 | 5.86 |
| CTVel I | 49161 | 121 | 0.25 | 2899 | 5.90 |
| CTVel II | 7833 | 32 | 0.41 | 518 | 6.61 |
| CTVel III | 20622 | 74 | 0.36 | 1449 | 7.03 |
| Spinal cord | 15358 | 34 | 0.22 | 1012 | 6.59 |
| Right parotid | 16422 | 57 | 0.35 | 1137 | 6.92 |
| Left parotid | 20916 | 51 | 0.24 | 1450 | 6.93 |
| Total | 187537 | 568 | 0.30 | 11817 | 6.30 |

Table 2: Computation times in seconds for the original and the proposed methods. The test with the proposed method was run two times. First, only the voxel center point was drawn for emulating the original method. Then, the whole voxel was drawn for comparing the performance.

| | | Drawing | |
|---|---|---|---|
| Beam | Original | Center | Voxel |
| | **(a)** Prostate case. | | |
| 36° | 0.19 | 0.75 | 0.83 |
| 108° | 0.17 | 0.77 | 0.83 |
| 180° | 0.20 | 0.78 | 0.84 |
| 252° | 0.16 | 0.75 | 0.85 |
| 324° | 0.19 | 0.77 | 0.84 |
| Total | 0.91 | 3.81 | 4.19 |
| | **(b)** Larynx case. | | |
| 260° | 0.64 | 1.22 | 1.36 |
| 300° | 0.72 | 1.22 | 1.33 |
| 340° | 0.64 | 1.19 | 1.34 |
| 20° | 0.56 | 1.25 | 1.34 |
| 60° | 0.61 | 1.20 | 1.34 |
| 100° | 0.69 | 1.22 | 1.37 |
| 180° | 0.64 | 1.22 | 1.34 |
| Total | 4.50 | 8.51 | 9.44 |

In most cases, it is enough to represent an anisotropic beamlet with 16 pixels (2 sub-beamlets composed by 8 pixels) to obtain the same solution in both methods.

The computation time comparison provided in table 2 showed that the drawing method is 4.60 times slower than the original method in the prostate case. However, longer computation times were expected since the drawing solution projects the whole voxel instead of only a point. Bearing this fact in mind, the results were indeed impressive, because projecting the whole voxel with the original method would involve to project eight points, create the polygonal line that encloses the points with a 2D convex hull algorithm, and finally test the polygon against the grid to obtain the activated beamlets. Only to project the eight voxel points would take longer than the drawing approach. An interesting finding was that the time needed in the drawing solution either for emulating the original method or for projecting the complete voxel was very similar. This may be explained by the time for setting up the OpenGL environment before dealing with each organ, since the drawing method in the larynx case, where the number of voxels is more than the double compared with the prostate case, is only 2.10 times slower than the original method. Therefore, the initialization of the OpenGL environment introduces an overhead time whose influence decreases as the amount of voxels to draw increases.

Finally, a comparison between the original and the drawing projection methods in terms of dose distribution results is not included. On the one hand, if the voxel size is set to work with the original method, the voxels are very small and the differences between both methods are hardly noticeable. On the other hand, if the voxel size is larger than the beamlet width or height, then the original method does not properly work and any comparison using its output will not be valid or fair.

## 2.6 CONCLUSIONS

In this chapter, we described a method for projecting the whole voxel over the intensity fluence map for finding which are the beamlets directly radiating that voxel. The projection is performed using a compute-by-drawing approach in a graphic card with the OpenGL library. The proposed method makes the resolution of voxels and bixels independent and fixed some situations were the current algorithm of projecting the voxel center point fails to properly model the relationship between beamlets and voxels.

# 3 UNIDIRECTIONAL SEGMENTATIONS WITH SEGMENT SHAPE CONTROL

## Contents

## 3.1 ABSTRACT

A unidirectional leaf sequencing algorithm which controls the shape of the segments and reduces leaf motion time for step-and-shoot dose delivery is presented. The problem of constructing segments controlling its shape was solved by synchronizing right leaves motion. This is done without increasing the number of segments, or the total number of monitor units, and taking into account the unidirectional leaf motion and the interleaf collision constraints. Compared to other unidirectional leaf sequencing methods, the proposed algorithm performs very similar. But, in addition, the segment shape control produces segments with smoother outlines and more compact shapes, which may help to reduce MLC specific effects when delivering the planned fluence map.

## 3.2 INTRODUCTION

In the SMLC mode, the non-uniform intensity radiation maps obtained after the FMO step are discrete fluence matrices, whose elements are naturals. The beam produced by a linear accelerator is uniform. Thus, a MLC segmentation method is needed for delivering this non-uniform maps, since it sequences the fluence matrix in different shaped beams with different beam weights [40].

In general, the MLC segmentation algorithms published assume that an MLC can deliver exactly the planned intensity map as is, without considering the MLC specific effects like the head scattering or the leaf transmission. This assumption can lead to significant discrepancies between the planned and the delivered intensity maps [7, 14, 30, 42, 52].

There are several published solutions for DMLC segmentations of this problem [17, 22]. In these papers, leaf transmission, collimator scatter and tongue-and-groove effects are considered, because the input intensity map is modified according to the difference between the planned and the delivered maps for including the previous MLC specific effects. These methods were designed for DMLC, and they can not be extended to SMLC, as pointed out in [68], which proposes an equivalent solution adapted to the static mode. These solutions have a serious drawback, because the modelling and verification of MLC specific effects is quite difficult and expensive.

On the other hand, two new methods or ways of planning doses for step-and-shoot IMRT were published, the direct aperture optimization (DAO) [55] and the direct machine parameter optimization (DMPO) [31]. Both articles discuss about the problem of considering the optimization and the segmentation as separated problems. This approach causes the differences between the planned and the delivered maps, because the MLC specific effects can not be included in the optimization. The proposed solution in both cases is to merge the optimization and the segmentation steps into a single one. This solution is probably the best one, but it has the drawback of coupling the optimization and the segmentation. Thus, it is not valid for conventional IMRT planning systems without changing completely their implementation.

The method proposed in this chapter is based on the results and conclusions of [12, 14, 30, 42]. The use of a large number of segments with complex shapes can increase collimator artefacts. In this situation, there are usually segments with small fields (or unbalanced X-Y axis) and low number of MU that will make difficult to accurately calculate the dose delivered to the patient. The output for these segments must be carefully computed and corrected by the dose calculation algorithm, considering the MLC specific effects. Therefore, these segments introduce tough requirements for geometric accuracy of the MLC and dosimetric accuracy of the linear accelerator.

The number of segments or their monitor units are not subject to changes, unless the DAO, the DMPO or any other DSS method is used, because the traditional leaf sequencing algorithms can not fix the NS or constrain the MU (the MU value directly depends on its segment, so the segment computation should consider the eventual associated weight as a new restriction). However, the segment shape can be influenced imposing a constraint for leaf synchronization. This leaf synchronization can be controlled by the algorithm, in order to balance shape uniformity versus NS or TNMU increase regarding the original solution.

In this chapter, it is described a SMLC segmentation method that includes: 1) unidirectionality [9, 57] for reducing the leaf motion time, 2) the interleaf collision constraint, so it can be used in MLCs with motion constraints, 3) a leaf synchronization constraint for controlling segment shape, generating segments with smoother outlines and more compact shapes, and 4) two different criteria for minimizing either the NS or the TNMU, opposite to the single criterion usually available on other algorithms. First criterion is a new one proposed in this chapter for the reduction of NS, and the second one is described in [24, 40] and it obtains segmentations with the optimal TNMU.

## 3.3 METHOD

Usually, a segmentation method decomposes a fluence matrix in different segments plus weights on an iterative process, and each iteration can be divided into two steps. First step is the computation of a segment (matrix of ones and zeros, understood as a mask) for a given fluence matrix, using a set of constraints. Second step is the computation of the weight associated to the obtained segment, following only one fixed criterion to minimize the NS or the TNMU. Finally, the segment multiplied by the weight is subtracted from the fluence matrix, generating a residual matrix that will be the input fluence matrix for next iteration.

The proposed algorithm allows the user, at the beginning of the process, to select which criteria will be used, in order to minimize the NS or the TNMU depending on the desired target. The pseudocode in section 2 illustrates this process.

This section will be divided in three subsections explaining: preliminary definitions, the computation of segments ($S_k$), and the computation of weights ($\alpha_k$).

### 3.3.1 Definitions and notation

A similar notation and definitions as given by Kalinowski in 2004, 2006 and Engel in 2005 are used in this chapter and the following chapters 4 and 5.

**Definition 1.** Let $A$ be a natural number matrix with $M$ rows and $N$ columns representing a given fluence matrix. Let $S$ be a segment; a segment is a matrix with the same dimensions as its fluence matrix $A$, but composed only of $\{0, 1\}$ natural numbers. When a given position in the segment is equal to $0$, it means this position is covered by a leaf. When it is equal to $1$, it means this position is letting radiation pass. The segmentation (or decomposition) of $A$ is expressed as

*This is the most generic definition of a segment or aperture because, as it is explained latter, some restrictions must be applied to generate feasible segments for a given MLC.*

$$A = \sum_{k=1}^{NS} \alpha_k \cdot S_k, \tag{3.1}$$

where NS is the number of segments, and $\alpha_k > 0$ is the weight accounting for a relative beam-on time, proportional to the MU to be delivered, for the kth segment.

**Definition 2.** Let $A_k$ be the residual matrix, which is obtained when $\alpha_{k-1} \cdot S_{k-1}$ is subtracted from $A_{k-1}$, i.e. :

$$\begin{aligned} A_1 &= A \\ A_2 &= A_1 - \alpha_1 \cdot S_1 \\ &\vdots \\ A_k &= A_{k-1} - \alpha_{k-1} \cdot S_{k-1} \\ \vec{0} &= A_k - \alpha_k \cdot S_k \end{aligned}$$

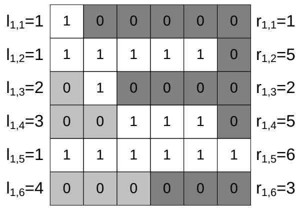
**Definition 3.** In order to simplify the notation, the $A$ matrix is expanded assuming that two zero rows and two zero columns are added at its boundaries:

$$\begin{array}{lll} \text{Rows}: & a_{0,j} = a_{M+1,j} = 0 & j \in [1, ..., N] \\ \text{Columns}: & a_{i,0} = a_{i,N+1} = 0 & i \in [1, ..., M] \end{array}$$

**Definition 4.** Let $l_{k,i}$ and $r_{k,i}$ denote the position of the left and right leaves at the i-th row in the k-th segment:

$$1 \leqslant l_{k,i} \leqslant r_{k,i} + 1 \leqslant N + 1 \qquad k \in [1, ..., NS], i \in [1, ..., M]$$

Where positions $l_{k,i}$ to $r_{k,i}$ are "opened" and exposed to radiation, while the left leaf at positions $[0..l_{k,i} - 1]$ and right leaf at positions $[r_{k,i} + 1..N + 1]$ are blocking radiation. The case of a row filled with zeros (totally closed) is included because it is allowed $l_{k,i} = r_{k,i} + 1$. Figure 14 shows $l_{k,i}$ and $r_{k,i}$ values for an example segment.



| | | | | | | | |
|---|---|---|---|---|---|---|---|
| $l_{1,1}$=1 | 1 | 0 | 0 | 0 | 0 | 0 | $r_{1,1}$=1 |
| $l_{1,2}$=1 | 1 | 1 | 1 | 1 | 1 | 0 | $r_{1,2}$=5 |
| $l_{1,3}$=2 | 0 | 1 | 0 | 0 | 0 | 0 | $r_{1,3}$=2 |
| $l_{1,4}$=3 | 0 | 0 | 1 | 1 | 1 | 0 | $r_{1,4}$=5 |
| $l_{1,5}$=1 | 1 | 1 | 1 | 1 | 1 | 1 | $r_{1,5}$=6 |
| $l_{1,6}$=4 | 0 | 0 | 0 | 0 | 0 | 0 | $r_{1,6}$=3 |

Figure 14: Values for $l_{k,i}$ and $r_{k,i}$ in a hypothetical segment. In light grey left leaves and in dark grey right leaves.

**Definition 5.** In a given row, there is a local peak at column p if both columns $p - 1$ and $p + 1$ have a lower value. When a set of contiguous repeated numbers is found, only the first one is taken into account and repetitions are ignored. Note that using definition 4, columns 0 and $N + 1$ are filled with zeros, and they must be taken into account. Table 3 shows some examples of peak detections. The number of local peaks in the i-th row can be expressed as:

$$\begin{aligned} \text{peaks}_i(A) = |\{x \in \mathbb{N} \ : \ & \exists p \in [1..N], \exists q \in [2..N+1] : \\ & x = q \ \land \ p < q \ \land \\ & A(i, p-1) < A(i, p) \ \land \ A(i, q) < A(i, p) \land \\ & \forall s : p < s < q : A(i, p) = A(i, s)\}\}| \end{aligned} \tag{3.2}$$

**Table 3:** Local row peak examples. The underlined elements are the peaks.

| 1 peak | 2 peaks |
|---|---|
| $(1\ \underline{3}\ 2\ 1)$ | $(1\ \underline{2}\ 1\ \underline{2}\ 2)$ |
| $(1\ \underline{2}\ 2\ 1)$ | $(1\ \underline{3}\ 2\ 3\ \underline{4})$ |
| $(1\ 2\ 2\ \underline{3})$ | $(\underline{3}\ 2\ \underline{4}\ 2\ 1)$ |
| $(\underline{5}\ 4\ 3\ 1)$ | $(\underline{3}\ 3\ 2\ \underline{3}\ 3)$ |

### 3.3.2 Segment computation ($S_k$)

The generation of the $S_k$ segment is driven by a set of constraints applied at the same time on the $A_k$ fluence matrix. Once the segment has been computed, a routine for solving collisions not predictable by the interleaf collision constraint (ICC) constraint is used, and the eventual segment is obtained.

*Basic constraints.*

The leaf sequencing algorithm is designed to meet two basic constraints: unidirectionality and interleaf collision constraint. For simplicity it is assumed that the leaf motion is from left to right, but it is straightforward to reverse the direction.

UNIDIRECTIONALITY ensures that no new maximums would be created in the i-th row, and therefore, the right leave would not move backwards (from right to left).

$$S_k(i,j) = \begin{cases} 1 & \text{if } \forall x \in [1..j-1] : A_k(i,j-x) \leqslant A_k(i,j) \\ 0 & \text{otherwise} \end{cases} \tag{3.3}$$
$$i \in [1..M],\ j \in [1..N]$$

*Example* 1. Consider the linear decomposition of a little test matrix using only this constraint.

$$\underbrace{\begin{pmatrix} 3 & 1 & 1 \\ 1 & 1 & 2 \end{pmatrix}}_{A} = \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}}_{S_1} + \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{S_2} + \underbrace{\begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \end{pmatrix}}_{S_3}$$

This constraint can be understood as a single row sliding window (or queue) that moves from the left to the right for each row independently. It adds (pushes) a new position to the window front only if it has equal or more intensity than the ones currently inside. It removes (pops) the last position at the window back if it has intensity 0. More than one position can be added or removed at the same time. Note that unidirectionality does not prevent collisions; the second mask of example 1 is a clear example.

INTERLEAF COLLISION CONSTRAINT do not allows the overlapping of opposite leaves in adjacent rows, which is basically a dependency among adjacent sliding windows.

$$S_k(i,j) = \begin{cases} 1 & \text{if } \forall x \in [1..j-1]: \\ & \quad A_k(i \pm 1, j-x) - x + 1 \leqslant A_k(i,j) \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$
$$i \in [1..M],\ j \in [1..N]$$

*Example* 2. In order to illustrate the meaning of this equation, it is applied on the little example together with unidirectionality. Upper row is the fluence matrix decomposed in segments and weights (no weight means it is equal to 1). Lower row is the residual matrix associated to each segment.

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

Equation 3.4 always ensures ICC. Although, in some cases, this restriction can be relaxed preserving ICC while reducing the NS. The relaxation is only for the mathematical constraint just as it is formulated, because sometimes it is too restrictive, but the real constraint will be always fulfilled. The softening is done adding a new variable called "subtraction" and represented by $r$:

$$S_k(i,j) = \begin{cases} 1 & \text{if } \forall x \in [1..j-1]\ \forall r \in [0..j]: \\ & \quad A_k(i \pm 1, j-x) - x + 1 - r \leqslant A_k(i,j) \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$
$$i \in [1..M],\ j \in [1..N]$$

The initial value of $r$ is set to $N$. The algorithm decreases its value iteratively as long as the segmentation process finishes without results, because an ICC violation occurs at some point of the segmentation and it is not solvable by the basic collision routine (explained in section 3.3.2). This minimization continues until the first value (and the highest) of $r$ fulfilling the ICC is found, and the segmentation process ends successfully. If the eventual value of $r$ is equal to $0$, this constraint is equivalent to the one described in equation 3.4, and it means the reduction of the NS is not possible.

The higher the $r$ value is, the lower the NS will be, because $r$ weakens the ICC, allowing the right leaf to advance, even if current fluence value is lower than the compared neighbour ones.

*Example* 3. To illustrate the effect of equation 3.5 in the segmentation process, the modified constraint is applied on the example 2 matrix with $r = 1$. Lower row is the residual matrix associated to each segment.

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

In this case, when $A_2$ is segmented, the second row can be included and it yields to a segmentation with one less segment.

*The segment shape constraint.*

A new variable called "depth" and represented by $d$ is added. It provides a certain degree of control over the segment shape, by limiting the difference between the adjacent leaves at the right side. As the ICC (section 3.3.2), it is a dependency among adjacent rows. Figure 15 illustrates the effect of $d$. Using the definition of $r_{k,i}$ previously given in definition 4:

$$|r_{k,i} - r_{k,i\pm1}| \leqslant d \qquad d \in [0..N] \tag{3.6}$$

The initial value of $d$ is set to $0$ and the algorithm maximizes it iteratively under the condition of keeping a maximum NS or TNMU according to the selected criterion. To be precise, when the subtraction value has been optimized, the segmentation process starts again with the subtraction value fixed and varying $d$. If the segmentation process fails with $d = n$, $n \in [0..N]$, it is repeated with $d = n + 1$, because the lower the $d$ value is, the smoother is the segment outline, but the higher is its NS or TNMU. This iterative process continues until the first value of $d$ that meets the maximum NS or TNMU is found. Section 3.3.3 explains in detail the relation between the $d$ variable and the NS and TNMU. The pseudocode in section 1 formalizes this process.

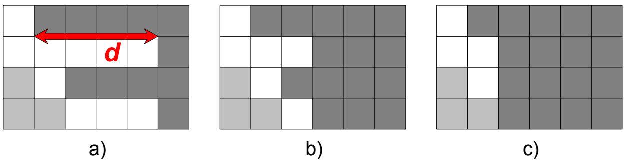


**Figure 15:** $d$ variable on a hypothetical segment: (a) original segment with $d$ represented graphically as an arrow, (b) $d \leqslant 2$ and (c) $d \leqslant 1$.

*General interleaf collision detection and solving.*

The ICC introduced in section 3.3.2 can be understood as a way of synchronizing the advance of the queues generated by the unidirectional constraint. The proposed constraint ensures that every single row is synchronized with its adjacent ones, but this is not enough to ensure no ICC violation on the whole segment. Thus, a routine for a global check is needed. Let $A$ be an example fluence matrix, and $\alpha_1 \cdot S_1$ its segmentation:

$$A = \begin{pmatrix} 0 & 0 & 0 & 5 & 5 \\ 0 & 0 & 0 & 0 & 0 \\ 5 & 5 & 0 & 0 & 0 \end{pmatrix} = 5 \cdot \begin{pmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix} = \alpha_1 \cdot S_1$$

Segment $S_1$ fulfils with the proposed unidirectionality and interleaf collision constraints, but the algorithm faces an unsolvable conflict if only these rules are applied. Therefore, these situations should be fixed using an algorithm like the following one: 1) find the row whose right leave is behind any other right leave in the segment, i.e the smallest $r_{k,i}$, in case of draw, the row whose left leave has the smallest $l_{k,i}$ is chosen, 2) use previous row index as the starting point for iterating with decreasing row indices (upwards) applying equation 3.7a. Then, the same procedure is applied with increasing row indices (downwards) applying equation 3.7b. At the end, any row violating the ICC must be closed.

$$\begin{aligned}
\text{Upwards: if } r_{k,i} < (l_{k,i-1} - 1) \;\rightarrow\;& l_{k,i-1} = r_{k,i} + 1; \quad (3.7a)\\
& r_{k,i-1} = r_{k,i};\\
\text{Downwards: if } r_{k,i} < (l_{k,i+1} - 1) \;\rightarrow\;& l_{k,i+1} = r_{k,i} + 1; \quad (3.7b)\\
& r_{k,i+1} = r_{k,i};
\end{aligned}$$

*The tongue-and-groove constraint.*

The tongue-and-groove design of the MLCs causes an underdose effect in a narrow band at the overlapping region between two adjacent rows. This effect can be removed from the segmentation methods introducing the tongue-and-groove constraint (TGC) [40, 57, 60].

The TGC also smooths segment outlines and compact segment shapes, so it can be used in a similar way as the segment shape constraint proposed in section 3.3.2. However, this constraint increases, in average, the NS and the TNMU [39, 60] and the increase can not be controlled, as it is done with the proposed shape constraint in section 3.3.3.

For the proposed algorithm, the TGC can be introduced following the same formulation described in [39].

$$\begin{aligned}
A(i,j) \leqslant & A(i+1,j) \wedge S(i,j) = 1 \Rightarrow & (3.8a)\\
& S(i+1,j) = 1 : \; i \in [1..M-1],\, j \in [1..N]\\
A(i,j) \leqslant & A(i-1,j) \wedge S(i,j) = 1 \Rightarrow & (3.8b)\\
& S(i-1,j) = 1 : \; i \in [2..M],\, j \in [1..N]
\end{aligned}$$

Equations 3.8a and 3.8b ensure that if a fluence value is smaller than its column neighbours ($i+1$ or $i-1$), it should be exposed at the same time than them. Thus, it can be guaranteed that the tongue-and-groove region, at least, receives the smaller dose.

### 3.3.3 Weight computation ($\alpha_k$)

The $\alpha_k$ weight associated to the $S_k$ segment is generated using a criterion for minimizing the NS or the TNMU. The proposed method allows to select which criterion will be used before the segmentation process starts, and it can not be changed during the execution. A novel criterion for the minimization of the NS is proposed and the criterion for obtaining optimum TNMU is taken from [24, 40].

First, the general strategy for computing weights is explained, and then each criterion is explained in detail.

*Minimization criteria.*

The selection of one criterion or another would depend on the desired factor to be minimized, the NS or the TNMU. They are used in the computation of the $\alpha_k$ weight associated to the $S_k$ segment. On both criteria, the procedure to obtain the candidate weights and search for the best one is the same:

1. The k-th candidate weights are obtained as the naturals in the range between 1 and the minimum fluence value (represented by $\omega$) in the k-th fluence matrix when the k-th mask is superimposed. Alternatively, $\omega$ can be understood as the lowest fluence value among the $l_{k,i}$ in opened leafs.

$$\begin{aligned} \omega = x \in \mathbb{N} :& \{\exists i \in [1..M] : l_{k,i} \leqslant r_{k,i} \wedge x = A_k(i, l_{k,i})\} \wedge \\ & \{\forall p \in [1..M] : (l_{k,p} \leqslant r_{k,p}) \Rightarrow x \geqslant A_k(p, l_{k,p})\} \\ & k \in [1..NS] \end{aligned} \tag{3.9}$$

Let be $\Omega$ the set with all the fluence values between 1 and $\omega$:

$$\Omega = \{x \in \mathbb{N} : 1 \leqslant x \leqslant \omega\} \tag{3.10}$$

2. The mask is multiplied with all the weights in the $\Omega$ set and subtracted from the current fluence matrix, so as to compute a set of pair: weight and its residual matrix.

$$\forall \beta \in \Omega : A^{\beta}_{k+1} = A_k - \beta \cdot S_k \tag{3.11}$$

3. Lastly, for each pair, the criterion selected is applied and if the current pair is better than previous ones, its candidate weight becomes the new temporary $\alpha_k$ weight.

Intuitively, the best choice is the highest weight in the $\Omega$ set, because it is the one delivering the larger dose. However, this way of proceeding is not the best, because smaller values may reduce better the value heterogeneity of the fluence matrix, facilitating subsequent segmentations and consequently achieving faster segmentations than a "greedy" approach.

*Optimizing the NS (ONS).*

The objective is to minimize the number of segments (NS) and note that the optimal NS does not imply the optimal TNMU.

The weight $\alpha$ at iteration $k$ is the natural value reducing the biggest number of row local peaks at $A_k$, i.e. $A_{k+1}$ will tend to have less peaks than $A_k$. The idea behind this criterion is that the fewer peaks in a row, the faster it is segmented. Thus, the fewer row local peaks in a matrix, the faster its segmentation is done.

As explained in section 3.3.3 a list of pair: candidate weight and its residual matrix will be computed. For each residual matrix, the total number of local row peaks is computed summing the local peaks of each row:

$$\text{peaks}(A) = \sum_{i=1}^{M} \text{peaks}_i(A) \tag{3.12}$$

The residual matrix with the fewer peaks will determine the candidate weight that becomes the final $\alpha_k$:

$$\begin{aligned}\alpha_k = \beta \in \Omega : \{A_{k+1} = A_k - \beta \cdot S_k\} \wedge \\ \{\forall \gamma \in \Omega : A^{\gamma}_{k+1} = A_k - \gamma \cdot S_k \wedge \\ peaks(A_{k+1}) \leqslant peaks(A^{\gamma}_{k+1})\}\end{aligned} \tag{3.13}$$

*Example* 4. Let be $\omega = 2$ the lowest fluence value for the first segment of a given fluence matrix. Thus, $\Omega = \{1, 2\}$ are the candidate values. After computing $A_2^{\beta=1} = A_1 - (1 \cdot S_1)$ and $A_2^{\beta=2} = A_1 - (2 \cdot S_1)$ the algorithm decides $\beta = 2$ is the best choice, and it becomes the final $\alpha_1$ value, because $peaks(A_2^{\beta=1}) = 27$ and $peaks(A_2^{\beta=2}) = 20$.

*Optimizing the* TNMU *(*OTNMU*).*

The objective is to minimize the following summation:

$$TNMU = \sum_{k=1}^{NS} \alpha_k \tag{3.14}$$

The weight $\alpha$ at iteration $k$ is computed in an equivalent way as defined in [24, 40].

1. The TNMU complexity of a row, denoted as $c_i(A)$, is the optimal TNMU for the i-th (single) row segmentation [24, 40]:

$$c_i(A) = \sum_{j=1}^{N} max(0, a_{i,j} - a_{i,j-1}) \qquad i \in [1..M], j \in [1..N] \tag{3.15}$$

2. The TNMU complexity of a matrix, denoted as $c(A)$, is the optimal TNMU for its segmentation [24, 40]:

$$c(A) = max_i(c_i(A)) \qquad i \in [1..M] \tag{3.16}$$

3. Using the TNMU complexity property, the algorithm knows in advance the optimal TNMU for a residual matrix. Therefore, the algorithm can use this property to select the candidate weight plus residual matrix with minimum value, achieving the optimal segmentation in terms of MU. Equation 3.17 formulates the process using previous equation 3.16 and the $\Omega$ set defined in equation 3.10.

$$\begin{aligned}\alpha_k = \beta \in \Omega : \{A_{k+1} = A_k - \beta \cdot S_k\} \wedge \\ \{\forall \gamma \in \Omega : A^{\gamma}_{k+1} = A_k - \gamma \cdot S_k \wedge \\ \beta + c(A_{k+1})) \leqslant (\gamma + c(A^{\gamma}_{k+1})\}\end{aligned} \tag{3.17}$$

*Example* 5. Using example 4 for the ONS. This time, the algorithm could decide $\beta = 1$ is the best choice, because $c(A_2^{\beta=1}) = 50$, $c(A_2^{\beta=2}) = 52$ and therefore $1 + 50 < 2 + 52$.

*Segment shape constraint relaxation.*

Summarizing, the segmentation process consists in two main steps. First, the segmentation of the fluence matrix is done several times, optimizing iteratively the subtraction value, but not using the shape constraint. Second, another loop of segmentations is executed, using the subtraction as a constant for optimizing the depth value, and improving segment shapes.

The optimization of the depth variable is done in the following way. The eventual NS or TNMU (depending on the criterion selected) of the subtraction optimization is stored as the "stop" value. The lower the depth value, the more synchronized the right leaves will be, but the this would also yield to bigger the NS or TNMU. The depth variable starts set to 0 and the segmentation is carried out imposing this maximum depth. If the segmentation process reaches the stop value without finishing, it means that the depth condition is too restrictive, and it should be increased. This iterative process continues until a value of depth is found that obtains a segmentation with equal NS or TNMU than the stop value. This condition can be relaxed by increasing the stop value by a percentage (equation 3.18); therefore, the result will be better in terms of shape, but worst in terms of NS or TNMU.

$$stop = original + (original \cdot percentage) \tag{3.18}$$

*Example* 6. Let the original NS be 22 and the percentage be 10%, which implies that the second step can produce segmentations up to 24 segments. In the first case, without soften the stop value, depth can be 4. In the second case, it may be reduced to 3 or 2.

### 3.3.4 Data and experimental setting

From now on, the proposed algorithm with different criteria will be referred as separate algorithms (optimizing the total number of monitor units (OTNMU) and optimizing the number of segments (ONS)) due to the differences found in results and behaviour.

The results section will show the performance of OTNMU and ONS against the leaf sequencing methods described in Galvin 1993 (Gal), Bortfeld 1994 (Bor), Xia 1998 (Xia), Siochi 1999 and Kalinowski 2006 (Kal).

The method published in Siochi 1999 is a combination of two algorithms applied in two steps embedded in an iterative optimization process. First step is called *extraction*. It is based on Galvin 1993, and its output is a bidirectional SMLC segmentation. Second step is called *RP*. It is a geometrical reformulation of the sweep technique described in Bortfeld 1994, and its output is a unidirectional segmentation. Both methods are combined in an iterative optimization process, which is driven by a formula that measures the treatment time in a realistic way, taking into account the TNMU, the leaf motion time, and the verification and recording (V&R).

Only the RP technique was implemented for comparison, discarding the extraction part. There were two reasons for this decision: (1) the ONS and OTNMU can be compared with another unidirectional method; also, the RP with ICC (and without the TGC) is optimum regarding the

TNMU [40, p 1016], and (2) there are not enough details in the original paper for reproduce accurately the implementation of the whole optimization process.

Finally, it is important to remark that:

1. Although Galvin 1993 and Bortfeld 1994 algorithms where designed without the ICC, they were modified in Xia 1998 to include this constraint and compare in a fair way, because the ICC increases the number of segments by approximately 25% on both algorithms.

2. Unidirectional segmentations with RP and the proposed algorithms are done in both directions, from left to right and from right to left, and the best solution is selected.

3. The results are divided into two groups, depending on the constraints applied. The ICC group only uses this constraint. The TGC group uses the ICC plus the TGC.

The methodology proposed by [67], also used in [49] and [40] is followed:

1. 1000 $15 \times 15$ matrices were segmented, each having random natural values from $0$ to $L$. The algorithm Xia 1998 and RP 1999 were implemented, but only the second one was used in the testing. The results for Galvin 1993, Bortfeld 1994, Xia 1998, RP 1999 and Kalinowski 2006 where taken from [39, 40, 67]. This experiment will allow to compare the proposed algorithm with the others from the statistical point of view.

2. A prostate cancer case was planned with the PCRT 3D® treatment planning system in order to compare results between the RP, Xia 1998, OTNMU and ONS. The comparison has two objectives: (1) comparing in a real clinical case, that can be substantially different from random generated matrices, and (2) comparing with well known methods; one unidirectional (RP) and another bidirectional (Xia). The clinical target volume (CTV) was radiated from 5 coplanar and equiangular beams (36°, 108°, 180°, 252° and 324°) in a 72 Gy plan. The dose volume constraints used were very similar to the ones described in [47]. For the rectum and the bladder, 70% of the volume receives $\leqslant 40\%$ of the prescribed dose.

## 3.4 RESULTS

The test results with random matrices are gathered in tables 4 and 5, which show the NS and the TNMU, respectively. As for the prostate results, they are provided in table 6 with the NS and the TNMU for the full set of fluence matrix segmentations. These results were obtained for OTNMU and ONS using the subtraction variable $r$ and the variable $d$ with $\mathit{percentage} = 0\%$. Thus, the optimization of $d$ is performed without increasing the NS or the TNMU originally achieved with the $r$ variable.

**Table 4:** Average NS. OT = OTNMU and ON = ONS.

| | Bidirectional | | | | Unidirectional | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | ICC | | | TGC | ICC | | | | TGC | | |
| L | Gal | Xia | Kal | Kal | Bor | RP | OT | ON | RP | OT | ON |
| 3 | 13.4 | 13.3 | 12.6 | 15.5 | 17.7 | 15.2 | 15.6 | 15.6 | 16.4 | 16.4 | 16.4 |
| 4 | 20.4 | 18.6 | 14.5 | 18.0 | 22.8 | 19.1 | 19.7 | 19.6 | 20.9 | 20.8 | 20.7 |
| 5 | 20.4 | 19.0 | 16.0 | 20.5 | 27.9 | 22.8 | 23.6 | 23.3 | 25.2 | 25.0 | 24.8 |
| 6 | 21.5 | 20.3 | 17.2 | 22.6 | 32.8 | 26.5 | 27.4 | 27.0 | 29.4 | 29.2 | 28.8 |
| 7 | 27.1 | 20.0 | 18.2 | 24.3 | 37.9 | 30.1 | 31.4 | 30.6 | 33.6 | 33.3 | 32.6 |
| 8 | 28.2 | 24.3 | 19.1 | 25.7 | 42.8 | 33.7 | 35.0 | 33.9 | 37.7 | 37.4 | 36.3 |
| 9 | 28.3 | 24.3 | 19.9 | 27.0 | 47.8 | 37.1 | 38.6 | 37.1 | 41.7 | 41.4 | 39.9 |
| 10 | 28.9 | 25.7 | 20.7 | 28.3 | 52.6 | 40.6 | 42.3 | 40.3 | 45.6 | 45.3 | 43.3 |
| 11 | 30.9 | 25.7 | 21.3 | 29.5 | 57.6 | 43.9 | 45.8 | 43.2 | 49.4 | 49.1 | 46.6 |
| 12 | 34.8 | 27.0 | 21.9 | 30.5 | 62.4 | 47.2 | 49.1 | 46.2 | 53.2 | 52.8 | 49.8 |
| 13 | 35.5 | 26.9 | 22.5 | 31.4 | 67.3 | 50.5 | 52.5 | 49.0 | 56.8 | 56.5 | 52.9 |
| 14 | 35.6 | 26.9 | 23.0 | 32.2 | 72.2 | 53.6 | 55.7 | 51.7 | 60.3 | 60.0 | 56.0 |
| 15 | 35.9 | 26.7 | 23.5 | 33.1 | 77.1 | 56.7 | 58.9 | 54.6 | 63.7 | 63.5 | 59.0 |
| 16 | 41.7 | 30.0 | 24.0 | 33.9 | 82.0 | 59.7 | 62.1 | 57.2 | 67.1 | 66.8 | 61.8 |

**Table 5:** Average TNMU. OT = OTNMU and ON = ONS.

| | Bidirectional | | | | Unidirectional | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | ICC | | | TGC | ICC | | | | TGC | | |
| L | Gal | Xia | Kal | Kal | Bor | RP | OT | ON | RP | OT | ON |
| 3 | 19.7 | 19.5 | 15.4 | 16.6 | 17.7 | 15.4 | 15.7 | 15.7 | 16.5 | 16.5 | 16.5 |
| 4 | 40.5 | 29.6 | 19.5 | 21.2 | 22.8 | 19.6 | 19.9 | 19.9 | 21.1 | 21.0 | 21.0 |
| 5 | 40.1 | 30.9 | 23.6 | 25.8 | 27.9 | 23.6 | 24.0 | 24.1 | 25.7 | 25.3 | 25.4 |
| 6 | 44.2 | 46.8 | 27.6 | 30.3 | 32.8 | 27.7 | 28.1 | 28.3 | 30.1 | 29.7 | 29.8 |
| 7 | 67.1 | 45.6 | 31.7 | 34.9 | 37.9 | 31.7 | 32.2 | 32.5 | 34.6 | 34.1 | 34.3 |
| 8 | 72.3 | 63.4 | 35.7 | 39.2 | 42.8 | 35.8 | 36.3 | 36.7 | 39.1 | 38.4 | 38.8 |
| 9 | 72.3 | 67.1 | 39.8 | 43.6 | 47.8 | 39.8 | 40.3 | 40.8 | 43.6 | 42.7 | 43.1 |
| 10 | 76.5 | 68.6 | 43.8 | 48.2 | 52.6 | 43.8 | 44.4 | 45.1 | 48.1 | 47.1 | 47.7 |
| 11 | 81.4 | 68.6 | 47.7 | 52.9 | 57.6 | 47.8 | 48.5 | 49.2 | 52.5 | 51.4 | 52.1 |
| 12 | 106.8 | 101.1 | 51.8 | 57.2 | 62.4 | 51.8 | 52.5 | 53.3 | 57.0 | 55.7 | 56.4 |
| 13 | 101.1 | 100.6 | 55.7 | 61.7 | 67.3 | 55.8 | 56.5 | 57.5 | 61.4 | 60.0 | 60.9 |
| 14 | 112.7 | 100.0 | 59.8 | 66.0 | 72.2 | 59.8 | 60.6 | 61.6 | 65.8 | 64.3 | 65.4 |
| 15 | 116.0 | 98.0 | 63.8 | 70.6 | 77.1 | 63.8 | 64.6 | 65.8 | 70.3 | 68.6 | 69.7 |
| 16 | 154.5 | 124.9 | 67.7 | 74.8 | 82.0 | 67.8 | 68.7 | 69.9 | 74.7 | 72.9 | 74.0 |

**Table 6:** NS and TNMU obtained for each beam in a real prostate cancer case.

| | NS | | | | | | TNMU | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 36° | 108° | 180° | 252° | 324° | Total | 36° | 108° | 180° | 252° | 324° | Total |
| ICC | | | | | | | | | | | | |
| Xia | 36 | 13 | 32 | 15 | 31 | 127 | 267 | 108 | 194 | 80 | 214 | 863 |
| RP | 25 | 13 | 32 | 13 | 28 | 111 | 92 | 86 | 80 | 64 | 106 | 428 |
| OTNMU | 26 | 14 | 31 | 13 | 29 | 113 | 92 | 86 | 80 | 64 | 106 | 428 |
| ONS | 25 | 14 | 28 | 11 | 25 | 103 | 96 | 88 | 126 | 64 | 142 | 516 |
| TGC | | | | | | | | | | | | |
| RP | 32 | 15 | 32 | 12 | 33 | 124 | 138 | 86 | 86 | 69 | 135 | 514 |
| OTNMU | 30 | 26 | 33 | 13 | 33 | 135 | 117 | 86 | 81 | 69 | 111 | 464 |
| ONS | 29 | 16 | 31 | 11 | 34 | 121 | 143 | 97 | 95 | 80 | 154 | 569 |

### *Segment shape comparison*

In this subsection, the segment shape between Xia, RP, OTNMU and ONS are compared. The third beam (180°) segmentation from previous real case is used, because all the unidirectional methods segment it from left to right, and it will help to see the differences when comparing. The fluence matrix to be considered is

$$\begin{pmatrix}
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&40&40&40&40&40&40&0&0&0&0&0&0&30&40&40&40&40&40&40&40&40&0\\
0&40&40&40&40&40&40&40&40&40&40&40&40&40&40&40&40&40&40&40&40&40&0\\
0&40&40&40&40&40&40&40&40&37&22&28&11&40&40&40&40&40&40&40&40&13&0\\
0&40&40&40&40&40&40&32&27&22&25&30&17&36&30&40&40&40&40&40&40&0&0\\
0&0&40&40&40&40&40&40&32&33&20&29&29&37&40&40&40&40&40&40&40&0&0\\
0&0&0&0&40&40&40&40&29&24&31&27&38&40&40&40&40&40&40&0&0&0&0\\
0&0&0&0&0&0&40&40&27&32&18&26&40&40&40&40&40&0&0&0&0&0&0\\
0&0&0&0&0&0&40&40&40&40&39&40&40&40&40&40&40&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&15&40&40&40&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0
\end{pmatrix}.$$

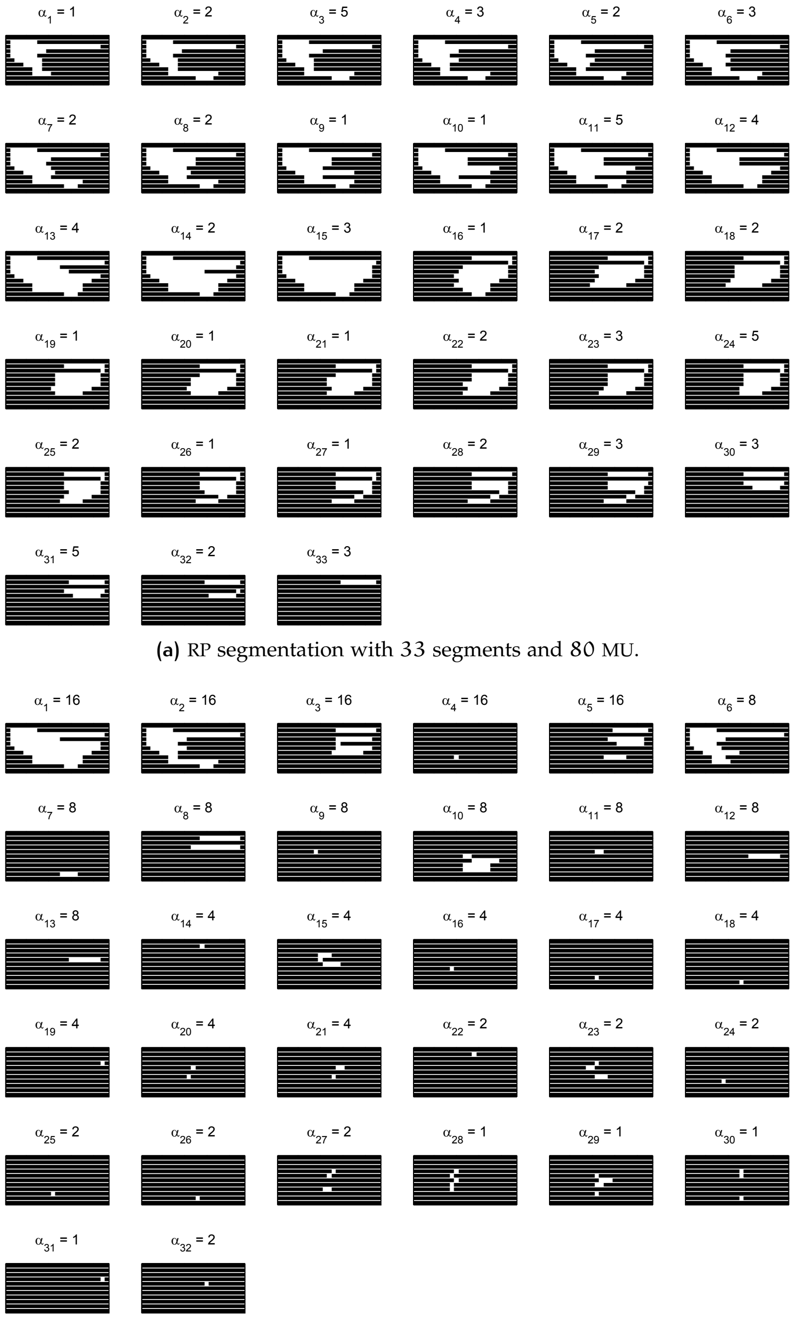
First, the segmentations obtained by Xia and RP methods are shown in figure 16, and the segmentations obtained by the ONS and the OTNMU without d are shown in figure 17. If it is only used the r variable, the algorithms try to obtain segmentations with less NS but not with better shapes. Thus, it is important to remark that some segment shapes in figure 16 and 17 have:

1. Quite irregular shapes, e.g. first and second row segments in RP and OTNMU.

2. Several disconnected subsegments, e.g. from 16th to 25th segment in RP, OTNMU and ONS.

3. Rather small areas, especially in the Xia method with most of the segments being tiny apertures. Although, the last ones of the dinamic methods are relatively small as well.

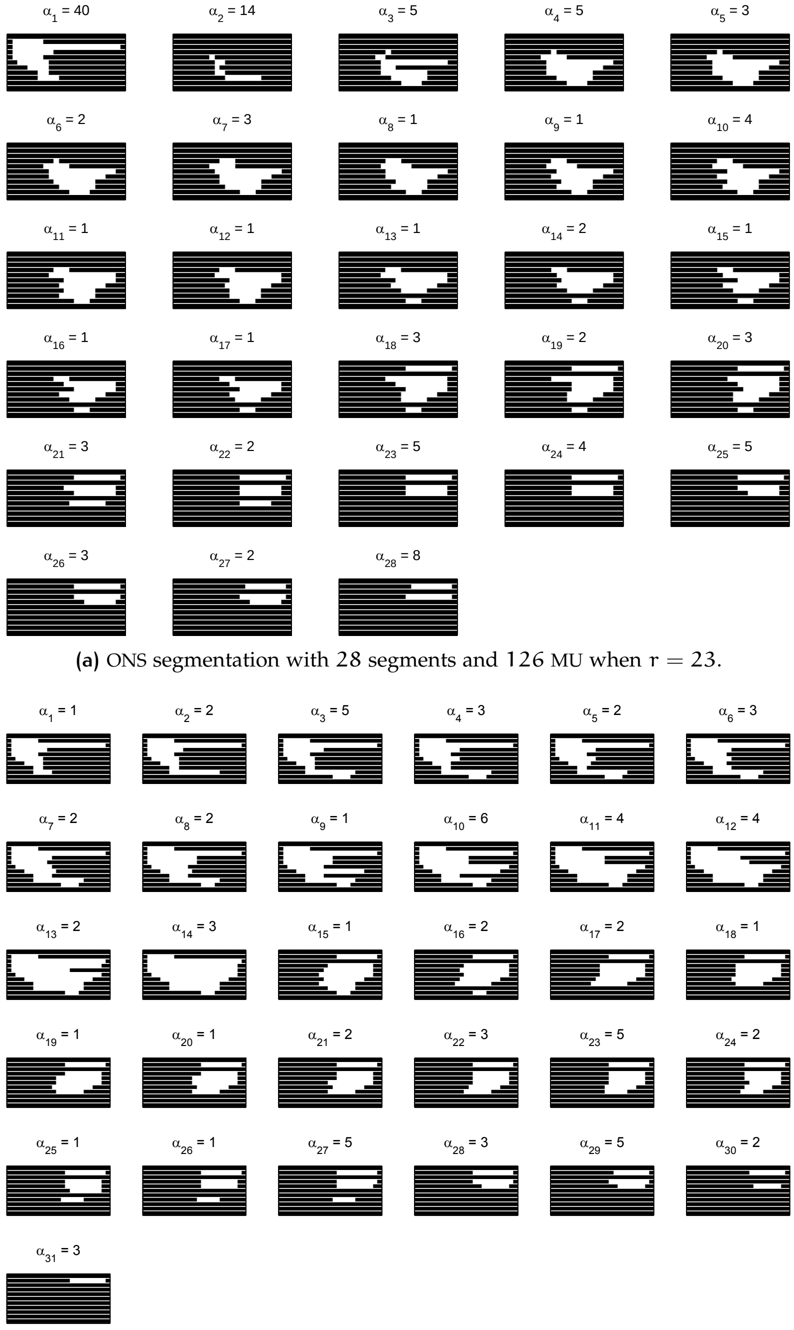
Second, figure 18 illustrates how adding the d variable can control and improve segment shapes, getting more regular outlines with less disconnected subsegments. Especially, when it is compared from 16th segment onwards in this figure with the same range of segments in



**(a)** RP segmentation with 33 segments and 80 MU.

**(b)** Xia segmentation with 32 segments and 194 MU.

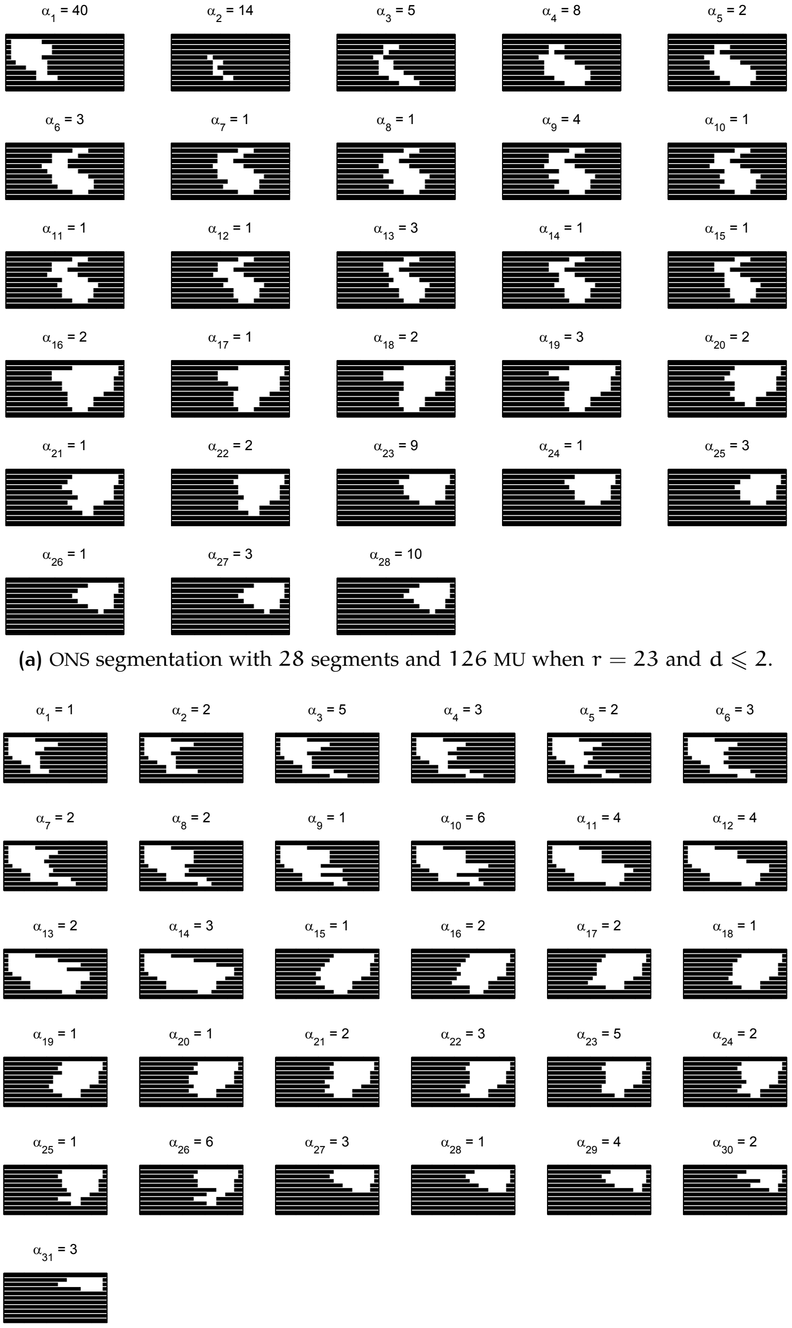
**Figure 16:** Segmentation of the 180° beam fluence matrix using RP and Xia algorithms.



**(a)** ONS segmentation with 28 segments and 126 MU when $r = 23$.

**(b)** OTNMU segmentation with 31 segments and 80 MU when $r = 23$.

**Figure 17:** OTNMU and ONS segmentations not using segment shape control for the 180° beam fluence matrix.



**(a)** ONS segmentation with 28 segments and 126 MU when $r = 23$ and $d \leqslant 2$.

**(b)** OTNMU segmentation with 31 segments and 80 MU when $r = 23$ and $d \leqslant 5$.

**Figure 18:** OTNMU and ONS segmentations using segment shape control for the 180° beam fluence matrix.

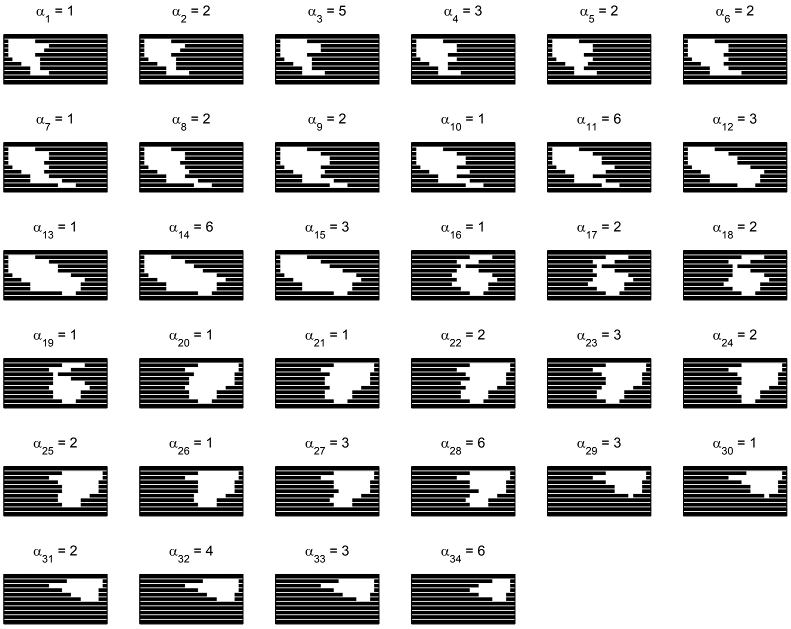


**Figure 19:** OTNMU segmentation with 34 segments and 86 MU when the TNMU is relaxed to reduce $d$, obtaining $r = 23$ and $d \leqslant 3$.

figures 16a and 17. The $d$ variable was optimized using the NS or TNMU obtained when the $r$ variable was computed.

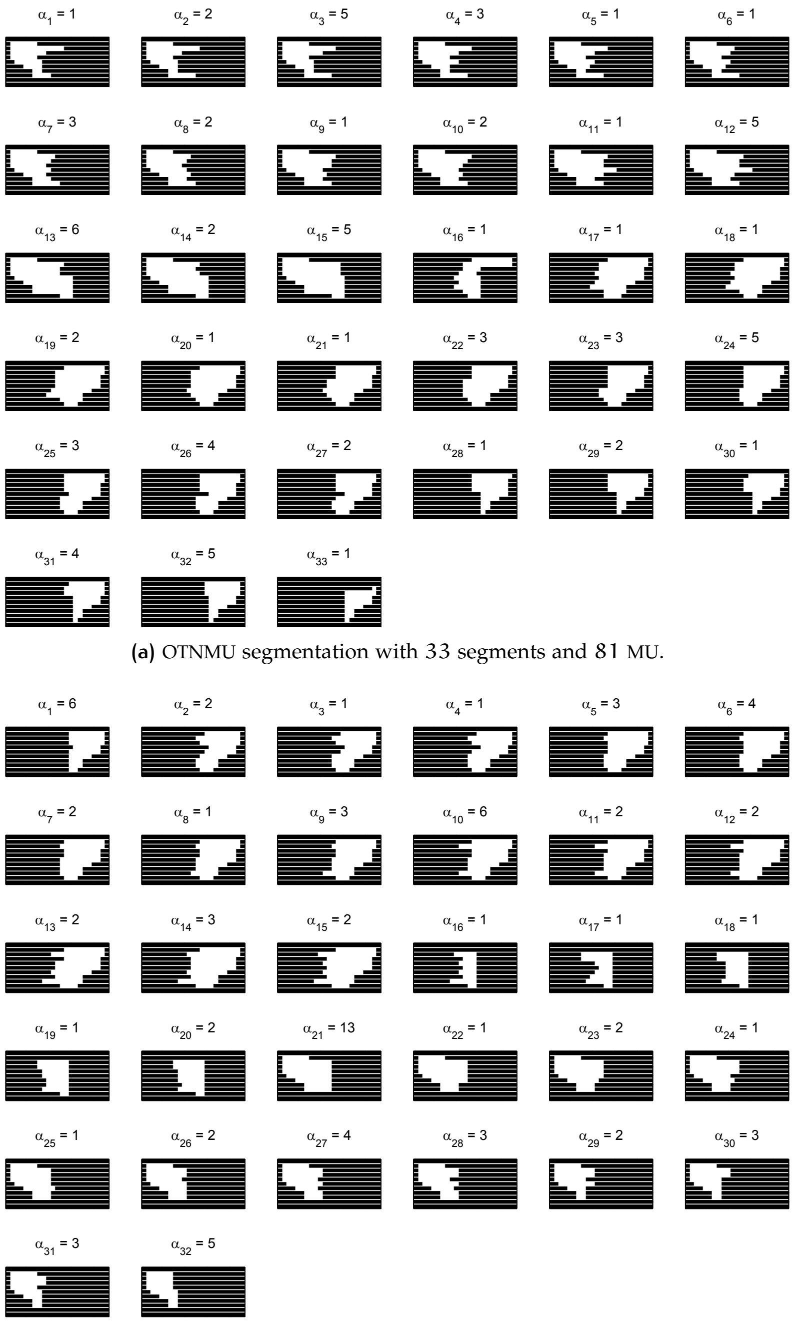
It is possible to reduce the value of $d$ with the purpose of getting even more regular shapes. This is done by relaxing the stop value using the percentage variable of equation 3.18. The OTNMU segmentation in figure 18b has $d \leqslant 5$, relaxing the TNMU by 10% as maximum, the result is a segmentation with $d \leqslant 3$ at the cost of 6 MU (7% more than the original). If a segmentation with $d \leqslant 2$ or lower is desired, the percentage could be increased. See figure 19 and compare with figure 18b the second and fifth row, where the differences are more evident.

Finally, figure 20 illustrates the influence of the TGC on the segment shapes. The effect is similar to the one seen in figure 18 using the proposed segment shape constraint.

## 3.5 DISCUSSION

Analyzing the random test results in tables 4 and 5, it can be concluded that the proposed algorithms show a good behaviour in terms of NS an TNMU, when comparing with other unidirectional segmentation methods, such as Bortfeld 1994 and the RP. In particular, the OTNMU gives almost identical results as the RP, whereas the ONS is slightly better than the RP and the OTNMU regarding the NS, but worst than both with respect to the TNMU, as one would expect.

The same behaviour can be observed in table 6 when looking at the real case test regarding both criteria for the RP, the OTNMU and the ONS, whereas Xia 1998 results in terms of the NS are not as good as the ones



**(a)** OTNMU segmentation with 33 segments and 81 MU.

**(b)** RP segmentation with 32 segments and 86 MU. The segmentation was from right to left.

**Figure 20:** Example of the TGC applied on the OTNMU and RP algorithms, showing its influence on the segment shape.

seen in the random test. However, the most remarkable difference is the improvement in the segment shapes observed in figures 18 and 19 compared to figures 16 and 17. The segment shape constraint has the properties of reducing the severe and intricate blocking, smoothing segment outlines and compacting segment shapes.

Those properties come from the fact that limiting the difference between a right leaf and its adjacent leaves: a) synchronizes their motion, creating a smoother segment "front" (and outline), b) contributes to the reduction of disconnected subsegments, minimizing the amount of interleaved closed rows, and c) unidirectionality plus leaf motion synchronization will tend to avoid situations were the shape has a short Y axis and a long X axis (e.g. only two opposite leaves opened). The motion of one leaf depends on its adjacent leaves; one leaf (or a little set of leaves) could not advance much more than their neighbours. Thus, the segment will be more compact and regular as it can be seen when comparing the latest segments in figures 16a, 17, and 18.

The leaf synchronization achieved in figure 18 is obtained without increasing the NS or the TNMU. However, if it is still not enough, there is the possibility of relaxing the stop value for the NS or the TNMU, e.g. 5% more NS or MU, and get even better shapes, but it is not worthy to increase it too much. Otherwise, segment shapes may be "perfect" but the segmentation will not be feasible in practice, due to the increase in delivery time. There is a trade-off between both factors, shape vs. NS and TNMU.

Finally, similar results were obtained in terms of shape when using the TGC, as it can be seen comparing figures 20 and 18. However, table 6 shows that this constraint increases the NS and TNMU more than 10% and 8%, respectively, on each algorithm. In addition, the TGC influence on the NS or TNMU is quite unpredictable and can not be limited or controlled.

## 3.6 CONCLUSIONS

A new MLC segmentation algorithm was developed. For the computation of segments, a novel constraint for controlling the segment shapes was added and two basic restrictions were considered, unidirectionality and the interleave collision. The segment shape control will generate compact and regular shapes, unidirectionality will minimize leaf movements, thus reducing one treatment time factor, and the interleave collision constraint will make suitable the proposed algorithm for MLCs with motion constraints.

For the computation of segment weights, the algorithm offers the possibility of selecting between two different criteria. First criterion is a novel one proposed in this chapter for minimizing the NS, and the second one is taken from [24, 40] for obtaining the optimal TNMU. The results show that the algorithm works well compared to other published algorithms, having the bonus of the shape control plus the criteria selection.

The next chapter will introduce a method for reducing the NS obtained in unidirectional segmentations, because it is quite common that the complexity of the solution makes impossible to deliver the plan obtained in the available time slots at the hospitals.

## 3.7 SEGMENTATION ALGORITHM PSEUDOCODE

**Algorithm 1** Main function. Optimization of $r$ and $d$ variables.

```
function SEGMENTATION($A, criterion, percentage$): returns $S, \alpha$
    $r \leftarrow N$; $d \leftarrow N$; $collision \leftarrow true$; $max \leftarrow 0$    ▷ Subtraction optimization
    while $r \geqslant 0 \wedge collision$ do
        $< collision, auxS, aux\alpha > \leftarrow decomposition(A, r, d, criterion, max)$
        if $collision$ then
            $r \leftarrow r - 1$
        end if
    end while
    $d \leftarrow 0$; $collision \leftarrow false$                    ▷ Depth optimization
    $max \leftarrow computeMaximum(S, \alpha, criterion, percentage)$
    while $d \leqslant N \wedge \neg collision$ do
        $< collision, auxS, aux\alpha > \leftarrow decomposition(A, r, d, criterion, max)$
        if $\neg collision$ then
            $S \leftarrow auxS$; $\alpha \leftarrow aux\alpha$
        else
            $d \leftarrow d + 1$
        end if
    end while
end function
```

**Algorithm 2** Function computing the segmentation of a fluence matrix with given parameters. $A, S$ and $\alpha$ are vectors containing the results for each iteration.

```
function DECOMPOSITION($F, r, d, criterion, max$): returns $collision, S, \alpha$
    $k \leftarrow 1$; $A(k) \leftarrow F$; $collision \leftarrow false$
    $lrPos \leftarrow initializeLeafPositions(A(k))$
    while $A(k) > 0 \wedge \neg collision$ do
        $< collision, S(k) > \leftarrow computeSegment(A(k), r, d, lrPos)$
        if $\neg collision$ then
            switch $criterion$
                case $NS$
                    $\alpha_k \leftarrow computeNSWeight(A(k), S(k))$
                case $TNMU$
                    $\alpha_k \leftarrow computeTNMUWeight(A(k), S(k))$
            end switch
            $A(k+1) \leftarrow A(k) - \alpha(k) \cdot S(k)$
            if $max \neq 0$ then                        ▷ Checking stop condition
                switch $criterion$           ▷ Collision boolean is reused to stop
                    case $NS$
                        $collision \leftarrow (k > max)$             ▷ Checking max. NS
                    case $TNMU$
                        $collison \leftarrow (sum(\alpha) > max)$  ▷ Checking max. TNMU
                end switch
            end if
            $k \leftarrow k + 1$
        end if
    end while
end function
```

**Algorithm 3** Function computing the segment for a fluence matrix with given parameters and previous segment stored in lrPos vector.

```
function COMPUTESEGMENT($A, r, d, lrPos$): returns $collision, S$
    for $i \leftarrow 1, M$ do
        $l_i \leftarrow advanceZeros(A, i, lrPos(i, 1))$
        if $lrPos(i, 2) < l_i$ then
            $r_i \leftarrow l_i - 1$                          ▷ Start with a closed aperture
        else
            $r_i \leftarrow lrPos(i, 2)$                      ▷ Start with k-1 position
        end if
        if $r_i < N + 1$ then
            $j \leftarrow r_i$; $continue \leftarrow true$
            while $j < N + 1 \wedge continue$ do
                if $\begin{cases} A(i, j-1) \leqslant A(i, j) \\ A(i \pm 1, j - x) - x + 1 - r < A(i, j) \\ j - r_{i-1} < d \end{cases}$ then
                    $r_i \leftarrow j$                        ▷ Checked $r_i < r_{i-1} + d$
                    $j \leftarrow j + 1$
                else
                    $continue \leftarrow false$
                end if
            end while
        end if
        if $i > 1$ then
            if $r_i - r_{i-1} < -d$ then                     ▷ Check $r_i \geqslant r_{i-1} - d$
                $r_{i-1} \leftarrow r_i + d$   ▷ $r_i$ can't reach $r_{i-1} - d$, and $r_{i-1}$ is modified
                $UpdateUpperrows(lrPos, i - 1)$  ▷ Propagate change upwards
            end if
        end if
    end for
    $solveBasicCollisions(lrPos)$                           ▷ Descrided in 3.3.2
    $collision \leftarrow detectCollisions(lrPos)$
    $S \leftarrow createSegmentMatrix(lrPos)$
end function
```

# 4 REDUCING THE NUMBER OF SEGMENTS IN UNIDIRECTIONAL MLC SEGMENTATIONS

## Contents

## 4.1 ABSTRACT

In this chapter, an algorithm for the reduction of the NS is presented for unidirectional segmentations, where there is no backtracking of the MLC leaves. It uses a geometrical representation of the segmentation output for searching the key values in a fluence matrix that complicate its decomposition. The NS reduction is achieved by performing minor modifications in these values, under the conditions of avoiding substantial modifications of the dose-volume histogram, and does not increase in average the total number of monitor units delivered.

## 4.2 INTRODUCTION

The optimal solution for a MLC segmentation minimizes the treatment time, which depends on three factors: (1) the total beam-on time or TNMU, (2) the MLC leaf motion time and (3) the V&R cycle overhead, which is the time needed for checking the correct position of the leaves. Delivery time is approximately proportional to the TNMU plus a function of the other two factors [57, p 673] that are directly related to the NS.

Some unidirectional segmentation methods like the RP or the OTNMU [4, p 577] are optimum, or close to the optimum, regarding the TNMU [4, 40]. However, the generally larger NS in unidirectional segmentations [4, 40] compared to bidirectional ones can seriously affect delivery times, especially if the V&R cycle of the MLC used is high [19, p 2113], since there are more segments to deliver and it is the only variable far from its optimum value. Therefore, a very valuable improvement in unidirectional segmentations would be the reduction of the NS, so that its main disadvantage diminishes or disappears.

In addition, fluence matrices with heterogeneous values and many gradient zones increase the NS and produce complex segment shapes. This heterogeneity is due to the treatment plan complexity, which can not be avoided, and inherent problems in the fluence matrix optimization process [59, p 2105].

*In general, the more conformed the radiation to the tumour is, the more heterogeneous the fluence matrix will be.*

In this chapter, a solution is proposed for unidirectional segmentations by finding some key elements in a heterogeneous fluence matrix, where a small modification can decrease the NS to be delivered. If these changes are done wisely, modifying only few elements and in a certain way, the alteration is performed without giving up quality on dose-volume histograms (DVHs) or increasing in average the TNMU. As a result, treatment time is decreased and the disadvantage of a larger NS compared to bidirectional algorithms is diminished.

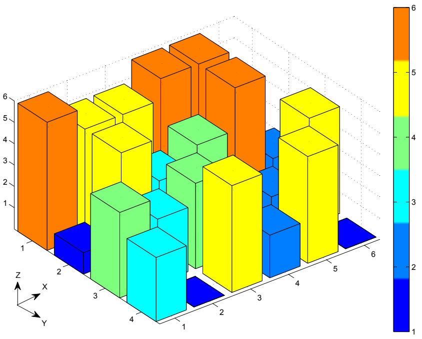
## 4.3 THE ROD PUSHING TECHNIQUE

The rod pushing (RP) technique is a unidirectional segmentation algorithm described in [57], and it is a geometrical reformulation of the sweep technique introduced in [9]. This method formulates the solution in a geometrical way, using two matrices for representing the segmentation. We propose to use this 3D representation so as to analyze the segmentation output for any unidirectional algorithm, and find the key fluence elements complicating its decomposition. In this section, the RP technique will be explained to introduce some concepts needed for a better understanding of the proposed method.

### 4.3.1 The Rod Pushing algorithm

The RP technique represents fluence matrices and their segmentation as an $M \times N$ matrix of rods, which are defined as solid blocks with height $a_{i,j}$. A handy and simple way of representing these rods is using two matrices of size $M \times N$. First one is called *base*, denoted by B, for the $z$ indexes where rod bases are. Second one is called *top*, denoted by T, for the $z$ indexes where rod tops are, and U is another $M \times N$ matrix filled with ones. Finally, a given fluence matrix A can be expressed in terms of B, T and U using equation 4.1. Figure 21 shows a random fluence matrix with its representation as a solid in 3D, and its corresponding top and base matrices.

$$A = T - B + U \tag{4.1}$$



(a) 3D object.

$$\underbrace{\begin{pmatrix} 6 & 5 & 5 & 6 & 6 & 0 \\ 1 & 5 & 3 & 4 & 6 & 2 \\ 4 & 3 & 4 & 1 & 2 & 5 \\ 3 & 0 & 5 & 2 & 5 & 0 \end{pmatrix}}_{A} = \underbrace{\begin{pmatrix} 6 & 5 & 5 & 6 & 6 & 0 \\ 1 & 5 & 3 & 4 & 6 & 2 \\ 4 & 3 & 4 & 1 & 2 & 5 \\ 3 & 0 & 5 & 2 & 5 & 0 \end{pmatrix}}_{T} - \underbrace{\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}}_{B} + U$$

(b) T and B matrices.

Figure 21: Fluence matrix representations.

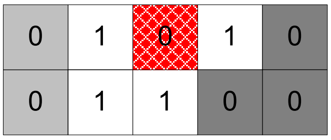
The RP technique spatially arranges rods in such a way that each plane $xy$ (same $z$ index) is a valid segment. The algorithm can move rods up and down along the $z$ axis, but $x$ and $y$ indexes can not be modified. The $x$ and $y$ indexes are MLC coordinates, and the $z$ index is the delivery order, as it can be seen in figures 23, 25, 26 and 28.

The rods are arranged following some rules, which formulate the physical and mechanical limitations of a given MLC, so that feasible segmentations for the device are generated.

### 4.3.2 Physical constraints

First constraint is the physical design of an MLC as two opposite banks of metal leaves. In a given row, there is one leaf located to the left and an opposite one located to the right. Thus, it is only possible to have one aperture, i.e. there will be a continuous set of ones and only one. Figure 22 shows an example where first row has two apertures, so it is not valid, whereas second row is feasible.

*The opposite banks of metal leaves can be seen in figure 1(c).*



|   |   |   |   |   |
|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 |

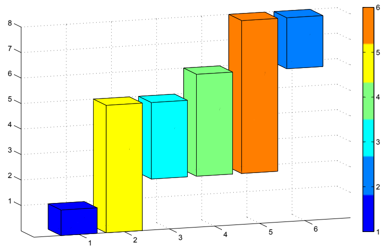
**Figure 22:** One aperture per row example. The red shaded position can not be generated with an MLC. The left and right MLC leaves are drawn in light and dark grey, respectively.

The rule to avoid non-contiguous sets of ones is given by the RP rules 4.2a and 4.2b [57]:

$$\text{if } a_{i,j+1} \geqslant a_{i,j} \rightarrow b_{i,j+1} = b_{i,j} \quad (4.2a)$$

$$\text{if } a_{i,j+1} < a_{i,j} \rightarrow t_{i,j+1} = t_{i,j} \quad (4.2b)$$

where $b_{i,j}$, $t_{i,j}$ are the $z$ indexes of the rod base and top in the $i$th row and $j$th column, respectively. Figure 23 shows this constraint applied to a single row.



**(a)** 3D rods for the RP rule.

$$\begin{array}{rl}
 & (1\ 5\ 3\ 4\ 6\ 2) = \\
8) & (0\ 0\ 0\ 0\ 1\ 1) + \\
7) & (0\ 0\ 0\ 0\ 1\ 1) + \\
6) & (0\ 0\ 0\ 1\ 1\ 0) + \\
5) & (0\ 1\ 1\ 1\ 1\ 0) + \\
4) & (0\ 1\ 1\ 1\ 1\ 0) + \\
3) & (0\ 1\ 1\ 1\ 1\ 0) + \\
2) & (0\ 1\ 0\ 0\ 0\ 0) + \\
1) & (1\ 1\ 0\ 0\ 0\ 0)
\end{array}$$

**(b)** Segments for the RP rule.

**Figure 23:** (a) The rod pushing rule applied to a single example row. (b) The original fluence row decomposed in segments using the rod pushing rule. The number on the left side of each segment stands for the height and it is associated to its delivery order (if the MLC leaves move from left to right). It should be noted that segments 3, 4, 5 and 7, 8 can be merged.

### 4.3.3 Mechanical constraints

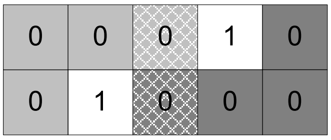
It is common in many MLC models that two opposite leaves in adjacent rows can not be overlapped. This limitation is translated as the inter-leaf collision constraint (ICC). Figure 24 shows an example where the upper left leaf and the lower right leaf are overlapped, and it is not a valid segment.

*With the rod pushing rule and the ICC constraint is enough to model most of the commercial MLC models.*

In order to add this constraint to the process, rods with size zero must be included in the RP process by applying rules 4.3a and 4.3b [57].

$$t_{i,j} = b_{i,j} + a_{i,j} - 1 \quad (4.3a)$$

$$b_{i,j} = t_{i,j} - a_{i,j} + 1 \quad (4.3b)$$



| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 |

Figure 24: Interleaf collision example. The upper left leaf and the lower right leaf can not block positions with the same column number.

Once previous rules 4.3a and 4.3b have been applied to the whole fluence matrix, and rods with height 0 have $b_{i,j} = 1$ and $t_{i,j} = 0$. The procedure to avoid interleaf collisions consists of finding in each column the row with the highest top. This row will be the starting point for applying rules 4.4a and 4.4b, from its upper adjacent row and decreasing the row index until first row is reached.

$$\begin{aligned} \text{if } b_{i,j} > t_{i+1,j} + 1 \rightarrow t_{i+1,j} &= b_{i,j} - 1 \\ b_{i+1,j} &= t_{i+1,j} - a_{i+1,j} + 1 \qquad (4.4a) \\ \text{if } t_{i,j} + 1 < b_{i+1,j} \rightarrow t_{i,j} &= b_{i+1,j} - 1 \\ b_{i,j} &= t_{i,j} - a_{i,j} + 1 \qquad (4.4b) \end{aligned}$$

This procedure is repeated, but starting from the lower adjacent row of the starting point, and replacing $j+1$ for $j-1$ in rules 4.4a and 4.4b, iterating until last row is reached. This process is repeated until no more modifications are possible. Figure 25 shows the difference regarding using the RP rule alone or combined with the ICC rule; figures 25(a) and (b) show a matrix segmented using the RP rule, third segment has a collision in its second column (the underlined elements in 25(b)), and figures 25(c) and (d) show the matrix segmented using both rules, solving the collision.

Finally, the tongue-and-groove effect can be included in the RP technique [57, p 675]. However, it will not be considered in this method, because there are studies reporting that for IMRT plans with 5 or more beams and a large number of segments, the tongue-and-groove effect on the IMRT dose distribution is clinically insignificant due to the "smearing" of dose in tissue [21]. Besides, the tongue-and-groove constraint added to solve it, would yield to segmentations with around 10% in average more segments [4, 40], depending on the algorithm and case.

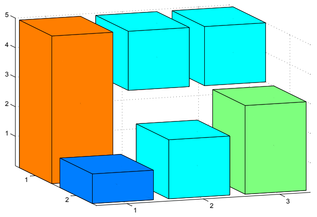
*This effect can be considered in any of the segmentation methods used in this master thesis, and the proposed method for reducing treatment times can obtain equal or even better results. However, we would be artificially complicating the experiments as we are interested in cases with a large number of beams and segments, where this effect is not important.*

## 4.4 METHOD

### 4.4.1 Segmentation insight

Observing how the RP technique placed the rods in figure 23, it is possible to think several ways of reducing the NS in that particular situation by adding or subtracting one fluence unit to a column, and figure 26 shows two examples.

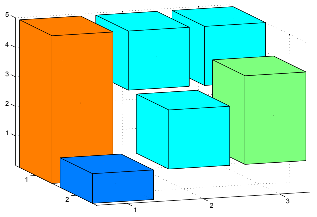
Two facts should be pointed out from the example shown in figure 26. First, a single unit change in one rod produces a "chain reaction" in top and base values of its posterior row rods. Besides, this



(a) 3D rods for RP rule.

$$\begin{pmatrix}5&2&2\\1&2&3\end{pmatrix} = \begin{pmatrix}5&5&5\\1&2&3\end{pmatrix} - \begin{pmatrix}1&4&4\\1&1&1\end{pmatrix} + 1 = \begin{pmatrix}1&0&0\\1&1&1\end{pmatrix} + \begin{pmatrix}1&0&0\\0&1&1\end{pmatrix} + \begin{pmatrix}1&\underline{0}&0\\0&\underline{0}&1\end{pmatrix} + \begin{pmatrix}1&1&1\\0&0&0\end{pmatrix} + \begin{pmatrix}1&1&1\\0&0&0\end{pmatrix}$$

(b) Segmentation result for RP rule.

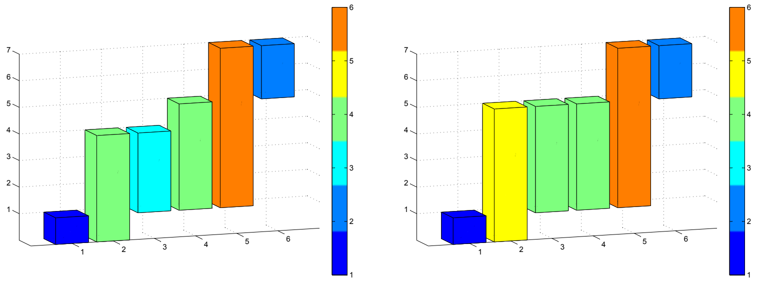


(c) 3D rods for ICC.

$$\begin{pmatrix}5&2&2\\1&2&3\end{pmatrix} = \begin{pmatrix}5&5&5\\1&3&4\end{pmatrix} - \begin{pmatrix}1&4&4\\1&2&2\end{pmatrix} + 1 = \begin{pmatrix}1&0&0\\1&0&0\end{pmatrix} + \begin{pmatrix}1&0&0\\0&1&1\end{pmatrix} + \begin{pmatrix}1&0&0\\0&1&1\end{pmatrix} + \begin{pmatrix}1&1&1\\0&0&1\end{pmatrix} + \begin{pmatrix}1&1&1\\0&0&0\end{pmatrix}$$

(d) Segmentation result for ICC.

**Figure 25:** (a) and (c) 3D representation of the rods when the RP rule and the ICC are applied, respectively, to the same example matrix. (b) and (d) The decomposition of the matrix in the top and base matrices and the resulting segmentation. Collision elements are underlined.



(a) Subtraction to 2nd column. NS = 4.

(b) Addition to 3rd column. NS = 3.

**Figure 26:** Reduction of the NS modifying one unit figure 23 fluence row.

chain reaction can reduce the NS in one or more segments, and the lower the $z$ index associated to the modified position is, the bigger the chain reaction will be. Second, the NS is not equal to the highest $z$ top index. Let $S_t$ be the set with the different $z$ indexes of T. Let $S_b$ be the set with the different $z$ indexes of $B - U$. Then, the NS can be computed with equation 4.5.

$$NS = |S_b \cup S_t| - 1 \tag{4.5}$$

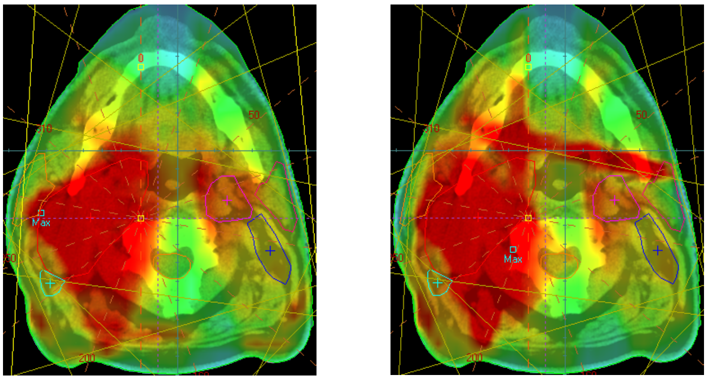
Intuitively, it is clear that each number in $S_b$ corresponds to the "base" of one slice (or segment), likewise each number in $S_t$ is the "top" of one segment. Hence, combining both sets, the eventual NS is obtained.

### 4.4.2 Fluence change considerations

Fluence matrices are obtained from an optimization process and modifying them outside this process can ruin the results. For this reason, it should be taken into account that:

1. The number of modified elements in a fluence matrix should be as few as possible.

2. The number of units added or subtracted in each position is even more important than the number of elements being modified. If a small set of positions is highly altered, the eventual dose could be seriously affected. Figure 27 illustrates this problem with an example, where this undesired effect can be seen.



**(a)** Small changes in many values. **(b)** Large changes in few values.

**Figure 27:** Axial section of dose distributions with different reduction strategies. The original dose is not showed, because it was very similar to the figure 27(a) dose distribution.

### 4.4.3 The algorithm

The method for modifying a fluence matrix and reducing the NS was designed as an iterative process. The objective is the reduction of its original NS by a given percentage or number of segments, controlled by the variable $\mathtt{maxReduction}$, while trying to modify the fewest elements as little as possible by carefully selecting the ones complicating the segmentation. The output of one iteration is the input for the next one, and the NS reduction is accumulated until the objective is achieved.

Two variables are defined for controlling which $z$ indexes are candidates to be modified. The first variable is the maximum $z$ distance to be taken into account, and it is denoted by $\mathtt{maxd}$. The second variable is the current $z$ distance limit, denoted by $\mathtt{dl}$, and the maximum number of fluence units that can be added or subtracted in a single operation. For example, if $\mathtt{maxd} = 2$, the algorithm will try to reduce the NS considering positions at distance $\pm 2$ using two iterations. In the first iteration, with $dl = 1$, only positions with $z$ indexes at distance $\pm 1$ of other $z$ indexes in T and B are modified adding or subtracting 1 fluence unit. If there is no NS reduction at the end of this iteration, then it starts again with $dl = 2$, checking positions at distance $\pm 2$ and modifying them 2 units. The algorithm is formulated in the following steps:

*Describing the algorithm with steps aims to its reproducibility.*

**Step 1.** A given fluence matrix A is segmented; T, B and its NS are obtained.

**Step 2.** A table with the following information regarding top and base matrices is filled. The table has as many rows as different indexes appear in T and B. For each row there are two columns containing: (1) the $z$ index in T and/or B, and (2) the total number of rows in T and B containing this index.

**Step 3.** Step 2 table is ordered increasingly by column 2, from less to more row occurrences. In case of draw, rows are ordered decreasingly by column 1, from higher to lower $z$ index. Finally, the first value of column 1 is selected, the higher $z$ index with less row occurrences.

**Step 4.** It is checked whether the current $z$ index is at distance $\pm dl$ from any other $z$ index, storing in a boolean variable $canBeDecreased$ if distance $-dl$ can be reached. If it is not the case, this index is ignored, because there is too much distance to its $z$ neighbour indexes and it is directly jumped to step 8.

**Step 5.** For each row in T and B, the selected $z$ index's first occurrence is searched. If it is found, its $i \in [1..M]$ and $j \in [1..N]$ indexes are stored. Remarks: (1) it is not possible to find the selected $z$ index in a row two times in non-contiguous places, because it would violate the RP rule (section 4.3.2), and (2) taking into account chain reactions, only its first occurrence should be treated to remove a set of continuous and identical $z$ indexes.

**Step 6.** For each T or B position computed in step 5, it is decided if the position fluence value in A will be modified and how following next rule:

$$\begin{array}{ll} \mathrm{if}\ (a_{i,j+1} - a_{i,j} = dl) \ \rightarrow\ a_{i,j} = a_{i,j} + dl \\ \mathrm{elseif}\ (canBeDecreased) \ \rightarrow\ a_{i,j} = a_{i,j} - dl \end{array} \tag{4.6}$$

It is not allowed to increment a fluence value $a_{i,j}$ unless its difference to next adjacent column neighbour is equal to $dl$. Previous column should not be checked because its $z$ value is always equal or lower to the current one by the RP rule. If it is not the case, it is checked whether a $dl$ subtraction generates a $z$ value already present in B and T. If so, $a_{i,j}$ will be decremented. Otherwise, the fluence value remains unchanged.

**Step 7.** Once the current $z$ index has been treated and A has been modified, segmentation is done again to check the NS reduction achieved.

**Step 8.** Steps 4 to 7 are repeated for each $z$ index in the first column of step 3 table. The NS obtained for each index are compared, and the smallest one determines which is the eventual fluence matrix.

**Step 9.** Steps 4 to 8 are repeated increasing $dl$ by 1 unit until there is a NS reduction or $maxd$ is exceeded. By default, $maxd$ should be equal to 1.

**Step 10.** Steps 1 to 9 are repeated accumulating the NS reduction. The eventual fluence matrix of step 9 is the input for next iteration. This iterative process continues until $maxReduction$ is reached, or it is not possible to reduce the NS (e.g. all $z$ indexes fail step 4 checking or the changes can not reduce the NS).

### 4.4.4 Application example

This example illustrates how the proposed method reduces the NS on a given fluence matrix. Figures 28(a) and (b) show a random $A$ matrix and its segmentation, respectively. The NS required for $A$ is

$$\left.\begin{array}{ll} S_b & = \{0,1,2\} \\ S_t & = \{1,2,3,4,5,6\} \end{array}\right\} \Rightarrow S_b \cup S_t = \{0,1,2,3,4,5,6\} \Rightarrow$$
$$\Rightarrow |S_b \cup S_t| - 1 = 6 = \mathrm{NS}.$$

The $z$ index appearing in less rows in $B$ and $T$ is $t_{2,4} = 6$. Applying equation 4.6, $a_{2,5} - a_{2,4} = 0 - 4 = -4$ (by definition 3); hence, the fluence can not be incremented and the eventual value is $a_{2,4} = 4 - 1 = 3$. Figures 28(c) and (d) show the segmentation obtained. The new NS is 5, because $S_b \cup S_t = \{0,1,2,3,4,5\}$, and it is easy to see which is the modification done comparing figures 28(a) and (c). The only rod whose top was at height 6 has been modified and the last segment disappears.

Figures 28(e) and (f) show the result for a second iteration, which obtains 4 segments. The algorithm seeks for the $z$ index appearing in less rows, looking at the $T$ and $B$ matrices in figure 28(d). The selected index is 4. Applying equation 4.6 the eventual value is $a_{3,3} = 3 - 1 = 2$. Modifying $a_{3,3}$ causes a chain reaction, and the second index equal to 4 at position $a_{3,4}$ also disappears. Figures 28(c) and (e) can be compared to see the changes.

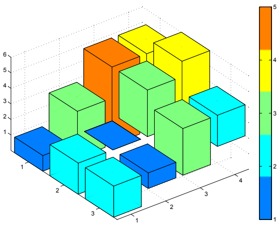
### 4.4.5 Generalization of the NS reduction algorithm

The method described in section 4.4.3 can be straightforwardly extended to other unidirectional segmentation methods, because their output can be considered as an RP output, computing its corresponding top and base matrices. Afterwards, top, base and fluence matrix can be passed to the NS reduction procedure under the condition of using the new unidirectional algorithm for segmenting in steps 1 and 7, instead of the original RP.

In short, the same algorithm can be used exactly as is, but exchanging the RP technique by another unidirectional segmentation method, and adding a function for computing top and base matrices for its output.

### 4.4.6 Data and experimental setup

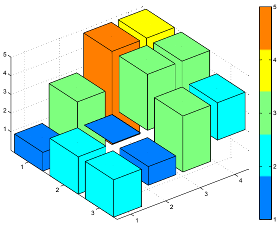
The experiments were performed using:



**(a)** Initial segmentation in 3D.

$$\begin{pmatrix}1&3&5&4\\2&0&3&4\\2&1&3&2\end{pmatrix}=\begin{pmatrix}1&3&5&5\\2&2&5&6\\2&2&4&4\end{pmatrix}-\begin{pmatrix}1&1&1&2\\1&3&3&3\\1&2&2&3\end{pmatrix}+U=$$
$$\begin{pmatrix}1&1&1&0\\1&0&0&0\\1&0&0&0\end{pmatrix}+\begin{pmatrix}0&1&1&1\\1&0&0&0\\1&1&1&0\end{pmatrix}+\begin{pmatrix}0&1&1&1\\0&0&1&1\\0&0&1&1\end{pmatrix}+$$
$$\begin{pmatrix}0&0&1&1\\0&0&1&1\\0&0&1&1\end{pmatrix}+\begin{pmatrix}0&0&1&1\\0&0&1&1\\0&0&0&0\end{pmatrix}+\begin{pmatrix}0&0&0&0\\0&0&0&1\\0&0&0&0\end{pmatrix}$$

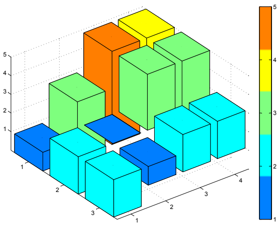
**(b)** Initial B and T matrices. $NS = 6$.



**(c)** First iteration in 3D.

$$\begin{pmatrix}1&3&5&4\\2&0&3&3\\2&1&3&2\end{pmatrix}=\begin{pmatrix}1&3&5&5\\2&2&5&5\\2&2&4&4\end{pmatrix}-\begin{pmatrix}1&1&1&2\\1&3&3&3\\1&2&2&3\end{pmatrix}+U=$$
$$\begin{pmatrix}1&1&1&0\\1&0&0&0\\1&0&0&0\end{pmatrix}+\begin{pmatrix}0&1&1&1\\1&0&0&0\\1&1&1&0\end{pmatrix}+\begin{pmatrix}0&1&1&1\\0&0&1&1\\0&0&1&1\end{pmatrix}+$$
$$\begin{pmatrix}0&0&1&1\\0&0&1&1\\0&0&1&1\end{pmatrix}+\begin{pmatrix}0&0&1&1\\0&0&1&1\\0&0&0&0\end{pmatrix}$$

**(d)** First iteration B and T matrices. $NS = 5$.



**(e)** Second iteration in 3D.

$$\begin{pmatrix}1&3&5&4\\2&0&3&3\\2&1&2&2\end{pmatrix}=\begin{pmatrix}1&3&5&5\\2&2&5&5\\2&2&3&3\end{pmatrix}-\begin{pmatrix}1&1&1&2\\1&3&3&3\\1&2&2&2\end{pmatrix}+U=$$
$$\begin{pmatrix}1&1&1&0\\1&0&0&0\\1&0&0&0\end{pmatrix}+\begin{pmatrix}0&1&1&1\\1&0&0&0\\1&1&1&1\end{pmatrix}+\begin{pmatrix}0&1&1&1\\0&0&1&1\\0&0&1&1\end{pmatrix}+$$
$$\begin{pmatrix}0&0&1&1\\0&0&1&1\\0&0&0&0\end{pmatrix}\cdot 2$$

**(f)** Second iteration B and T matrices. $NS = 4$.

**Figure 28:** (b), (d) and (f) The fluence matrix and its segmentation using the RP technique with the ICC. (a), (c) and (e) 3D representation of the results for the initial matrix, the first and the second iterations of the algorithm, respectively.

1. Three clinical cases planned with the PCRT 3D® treatment planning system.

2. The RP technique [57], the OTNMU and ONS algorithms described in [4, p 577], and a bidirectional method based on a branch-and-cut (BC) strategy included in the PCRT 3D® software.

3. The Siemens ONCOR™ Impression Plus linear accelerator installed at the Hospital Clínico Universitario Lozano Blesa (Avenida San Juan Bosco, 15, 50009 Zaragoza, Spain) for obtaining beam delivery times.

Besides, the tests were done under the following conditions:

1. Unidirectional segmentations with RP, OTNMU and ONS are done in both directions, from left to right and from right to left, and the best solution is selected.

2. None of the methods use the tongue-and-groove constraint for the reasons explained in section 4.3.3, and if only the ICC is included in the RP and OTNMU, the TNMU obtained is optimum [4, 40].

3. All the results were obtained with $\mathtt{maxReduction} = 100\%$, so the objective was the maximum possible NS reduction.

4. $\mathtt{maxd}$ was set to 1 for the RP and 2 for the other unidirectional methods, because the ONS and OTNMU obtain steeper segmentations in the 3D space.

The first clinical case is a prostate cancer radiated from five different coplanar and equiangular beams: 36°, 108°, 180°, 252° and 324°, in a 72 Gy plan. The dose volume constraint used for the rectum and bladder was 70% of their volume receives $\leqslant 40\%$ of the prescribed dose to the CTV.

The second case is a oropharynx cancer with three CTVs treated with seven coplanar but not equiangular beams: 20°, 60°, 100°, 180°, 260°, 300° and 340°. The prescribed dose for the clinical target volume gross disease region (CTVgr) was 74 Gy, and the doses for the clinical target volume elective nodal regions (CTVels) were 54 Gy for all of them. The constraint for the spinal cord was maximum dose $\leqslant 45$ Gy, and the constraint for both parotids was 50% of their volumes receive $\leqslant 40\%$ of the prescribed dose to the main CTV.

Third case is a larynx cancer treated with a seven coplanar beam plan with identical beam angles, constraints and prescribed dose for the gross disease region, but it has three elective nodal regions that were planned using 66 Gy, 56 Gy, and 50 Gy.

## 4.5 RESULTS

The results are presented using three tables that contain the NS and TNMU, the fluence change information, and the beam delivery times. Then, one DVH is used for each case to show the difference between the original plan, the plan with modified matrices and a third plan obtained with the aid of a radiotherapist by altering the original plan reducing the NS in each beam by the same percentage that achieved the proposed method.

Table 7 shows the NS and TNMU of the segmentations. For each unidirectional algorithm, there are two columns with the results for the original fluence matrix (labeled with "Or") and for the modified fluence matrix (labeled with "Md"). The NS results have a third column with the reduction percentage (labeled with "%"), and the BC columns show the results for the branch-and-cut bidirectional algorithm. Besides, some information regarding fluence changes is gathered in table 8, so it can be compared the ratio between the reduction achieved, the number of elements modified, and the maximum fluence changes to a single element. Lastly, delivery times for each beam of the RP algorithm are gathered in table 9 for the prostate and larynx cases only, which had the smallest and biggest NS for this segmentation method, in order to illustrate the relation between the NS and the delivery time reductions.

The DVHs obtained for the cases are shown in figure 29. These DVHs have a third set of data apart from the original and the modified matrices, which is the original segmentation manually modified by a radiotherapist (labeled with "Man"). These modifications were performed in order to discard several segments in each matrix segmentation for reducing the NS by the same percentage obtained from the modified fluence matrix. For the sake of brevity and conciseness, the ONS and OTNMU results are not showed, because they were very similar to the RP results.

**Table 7**: NS and TNMU obtained for each beam in three cancer cases. Or = original matrix, Md = modified matrix, % = reduction percentage.

| | NS | | | | | | | | | | TNMU | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | RP | | | OTNMU | | | ONS | | | | RP | | OTNMU | | ONS | | |
| Beam | Or | Md | % | Or | Md | % | Or | Md | % | BC | Or | Md | Or | Md | Or | Md | BC |
| **(a)** Prostate case results. | | | | | | | | | | | | | | | | | |
| 36° | 37 | 24 | 35.1 | 43 | 28 | 34.9 | 32 | 24 | 25.0 | 27 | 76 | 75 | 76 | 75 | 92 | 86 | 137 |
| 108° | 29 | 21 | 27.6 | 29 | 21 | 27.6 | 28 | 20 | 28.6 | 21 | 97 | 97 | 97 | 97 | 99 | 100 | 141 |
| 180° | 47 | 42 | 10.6 | 49 | 38 | 22.4 | 45 | 35 | 22.2 | 32 | 95 | 95 | 95 | 94 | 105 | 105 | 169 |
| 252° | 25 | 19 | 24.0 | 27 | 22 | 18.5 | 26 | 24 | 7.7 | 22 | 82 | 82 | 82 | 82 | 82 | 82 | 124 |
| 324° | 41 | 32 | 22.0 | 43 | 38 | 11.6 | 40 | 28 | 30.0 | 28 | 109 | 108 | 109 | 108 | 131 | 122 | 164 |
| Total | 179 | 138 | 22.9 | 191 | 147 | 23.0 | 171 | 131 | 23.4 | 130 | 459 | 457 | 459 | 456 | 509 | 495 | 735 |
| **(b)** Oropharynx case results. | | | | | | | | | | | | | | | | | |
| 260° | 28 | 24 | 14.3 | 30 | 21 | 30.0 | 27 | 19 | 29.6 | 26 | 49 | 49 | 49 | 48 | 54 | 51 | 79 |
| 300° | 40 | 27 | 32.5 | 44 | 35 | 20.5 | 41 | 30 | 26.8 | 30 | 49 | 47 | 49 | 48 | 49 | 49 | 76 |
| 340° | 51 | 36 | 29.4 | 53 | 41 | 22.6 | 44 | 39 | 11.4 | 51 | 68 | 68 | 68 | 67 | 69 | 68 | 207 |
| 20° | 52 | 40 | 23.1 | 53 | 42 | 20.8 | 52 | 36 | 30.8 | 36 | 74 | 74 | 77 | 77 | 87 | 77 | 359 |
| 60° | 32 | 23 | 28.1 | 27 | 24 | 11.1 | 26 | 22 | 15.4 | 33 | 38 | 38 | 38 | 38 | 39 | 39 | 80 |
| 100° | 29 | 23 | 20.7 | 30 | 27 | 10.0 | 28 | 23 | 17.9 | 20 | 45 | 47 | 46 | 46 | 47 | 51 | 67 |
| 180° | 46 | 37 | 19.6 | 45 | 41 | 8.9 | 44 | 33 | 25.0 | 44 | 55 | 55 | 56 | 55 | 57 | 60 | 384 |
| Total | 278 | 210 | 24.5 | 282 | 231 | 18.1 | 262 | 202 | 22.9 | 240 | 378 | 378 | 383 | 379 | 402 | 395 | 1252 |

**Table 7:** NS and TNMU obtained for each beam in three cancer cases. Or = original matrix, Md = modified matrix, % = reduction percentage.

| | NS | | | | | | | | | | TNMU | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | RP | | | OTNMU | | | ONS | | | | RP | | OTNMU | | ONS | | |
| Beam | Or | Md | % | Or | Md | % | Or | Md | % | BC | Or | Md | Or | Md | Or | Md | BC |
| (c) Larynx case results. | | | | | | | | | | | | | | | | | |
| 260° | 42 | 32 | 23.8 | 43 | 38 | 11.6 | 40 | 32 | 20.0 | 51 | 58 | 58 | 60 | 60 | 62 | 61 | 231 |
| 300° | 53 | 38 | 28.3 | 55 | 49 | 10.9 | 52 | 41 | 21.2 | 59 | 66 | 67 | 67 | 65 | 73 | 71 | 294 |
| 340° | 57 | 42 | 26.3 | 58 | 47 | 19.0 | 46 | 38 | 17.4 | 55 | 70 | 70 | 70 | 69 | 76 | 71 | 247 |
| 20° | 43 | 31 | 27.9 | 43 | 32 | 25.6 | 40 | 29 | 27.5 | 32 | 66 | 66 | 66 | 65 | 73 | 67 | 118 |
| 60° | 38 | 29 | 23.7 | 44 | 35 | 20.5 | 36 | 27 | 25.0 | 26 | 62 | 62 | 62 | 62 | 66 | 66 | 82 |
| 100° | 58 | 45 | 22.4 | 59 | 47 | 20.3 | 52 | 36 | 30.8 | 51 | 85 | 86 | 85 | 81 | 95 | 89 | 251 |
| 180° | 59 | 44 | 25.4 | 60 | 49 | 18.3 | 52 | 46 | 11.5 | 58 | 70 | 70 | 77 | 77 | 77 | 76 | 296 |
| Total | 350 | 261 | 25.4 | 362 | 297 | 18.0 | 318 | 249 | 21.7 | 332 | 477 | 479 | 487 | 479 | 522 | 501 | 1519 |

**Table 8:** Fluence change information for table 7 results. "+" and "−" are maximum addition and subtraction, respectively, to a single element, C = elements modified, NE = non-zero elements.

| | | RP | | | OTNMU | | | ONS | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Beam | NE | + | − | C | + | − | C | + | − | C |
| **(a)** Prostate case. | | | | | | | | | | |
| 36° | 101 | 1 | 2 | 10 | 2 | 2 | 13 | 2 | 2 | 9 |
| 108° | 77 | 1 | 1 | 8 | 1 | 2 | 6 | 1 | 2 | 8 |
| 180° | 121 | 1 | 1 | 5 | 1 | 3 | 12 | 1 | 3 | 12 |
| 252° | 76 | 1 | 1 | 5 | 1 | 1 | 4 | 1 | 1 | 3 |
| 324° | 96 | 1 | 1 | 10 | 1 | 1 | 9 | 1 | 2 | 16 |
| **(b)** Oropharynx case. | | | | | | | | | | |
| 260° | 114 | 0 | 1 | 7 | 1 | 2 | 25 | 1 | 2 | 15 |
| 300° | 166 | 1 | 3 | 30 | 1 | 1 | 25 | 1 | 1 | 13 |
| 340° | 195 | 1 | 2 | 25 | 1 | 1 | 20 | 1 | 4 | 7 |
| 20° | 228 | 1 | 1 | 20 | 0 | 4 | 15 | 1 | 2 | 22 |
| 60° | 146 | 1 | 1 | 23 | 0 | 1 | 5 | 1 | 2 | 7 |
| 100° | 104 | 1 | 1 | 12 | 0 | 2 | 6 | 0 | 1 | 6 |
| 180° | 236 | 1 | 1 | 13 | 1 | 1 | 11 | 1 | 3 | 20 |
| **(c)** Larynx case. | | | | | | | | | | |
| 260° | 143 | 1 | 1 | 23 | 2 | 2 | 12 | 1 | 1 | 26 |
| 300° | 195 | 1 | 1 | 22 | 0 | 1 | 7 | 1 | 1 | 22 |
| 340° | 173 | 1 | 2 | 27 | 0 | 2 | 13 | 0 | 2 | 17 |
| 20° | 153 | 1 | 3 | 22 | 1 | 2 | 23 | 1 | 2 | 34 |
| 60° | 135 | 1 | 1 | 13 | 1 | 3 | 18 | 1 | 2 | 13 |
| 100° | 172 | 1 | 1 | 30 | 2 | 4 | 23 | 1 | 2 | 28 |
| 180° | 199 | 1 | 1 | 30 | 1 | 1 | 20 | 1 | 1 | 15 |

**Table 9:** Beam delivery time table (in mm:ss) for the rod pushing technique. Beam order is the delivery order. 'Trans' is the transition time spent from the previous beam to the current one. 'Rad' is the total beam delivery time (beam-on time plus the time required for the leaves to move between segments).

| | Original | | Mod | |
|---|---|---|---|---|
| Beam | Trans | Rad | Trans | Rad |
| **(a)** Prostate case. | | | | |
| 180° | 00 : 00 | 03 : 10 | 00 : 00 | 03 : 00 |
| 252° | 00 : 26 | 02 : 04 | 00 : 25 | 01 : 50 |
| 324° | 00 : 25 | 03 : 08 | 00 : 25 | 02 : 50 |
| 36° | 00 : 25 | 02 : 35 | 00 : 25 | 02 : 04 |
| 108° | 00 : 27 | 02 : 21 | 00 : 27 | 02 : 03 |
| Total | 01 : 43 | 13 : 18 | 01 : 45 | 11 : 47 |
| **(b)** Larynx case. | | | | |
| 180° | 00 : 00 | 02 : 44 | 00 : 00 | 02 : 17 |
| 100° | 00 : 25 | 03 : 19 | 00 : 19 | 02 : 43 |
| 60° | 00 : 19 | 03 : 29 | 00 : 19 | 02 : 51 |
| 20° | 00 : 19 | 02 : 47 | 00 : 20 | 02 : 20 |
| 340° | 00 : 21 | 02 : 41 | 00 : 19 | 02 : 16 |
| 300° | 00 : 21 | 03 : 38 | 00 : 19 | 03 : 11 |
| 260° | 00 : 29 | 03 : 38 | 00 : 28 | 03 : 04 |
| Total | 02 : 14 | 22 : 16 | 02 : 04 | 18 : 42 |

## 4.6 DISCUSSION

From the results showed in previous section, it can be seen that the NS reduction achieved by the proposed method is variable and it depends on the fluence matrix processed. However, for the tested real cases, tables 7 shows that the reduction achieved is above 20% for the RP, while the OTNMU and ONS results confirm that the algorithm works fine with other unidirectional segmentation methods. In addition, table 7 results show how the difference in terms of NS between the unidirectional and bidirectional methods (such as the BC) can be reduced. Indeed, if the results after the processing are compared, it can be seen that they have similar NS, but unidirectional results have a smaller TNMU and minimize the leaf motion factor. Therefore, taking into account the formula for computing treatment times, their delivery time will be significantly smaller than the bidirectional algorithm.

The results in table 9 show that the method reduces beam delivery time, without taking into account beam transitions, by 11.4% for the prostate case and by 16.0% for the larynx case, although the TNMU is not decreased. Besides, the Optifocus™ MLC used in these tests has a negligible V&R (no delays between segments, apart from the leaf travel time itself), meaning that the tests were done under the worst conditions, i.e., any MLC with a V&R $\geqslant 1$ second could obtain higher time reductions. Accordingly, these beam delivery time reductions are only an example, because they would strongly depend on the equipment. In systems with high V&R overheads, the NS will become more important, whereas in low MU/s rate systems, the TNMU will have a bigger influence on delivery times.

Finally, figure 29 DVHs show that the modifications done to fluence matrices:

1. Do not substantially alter the dose delivered to CTVs.

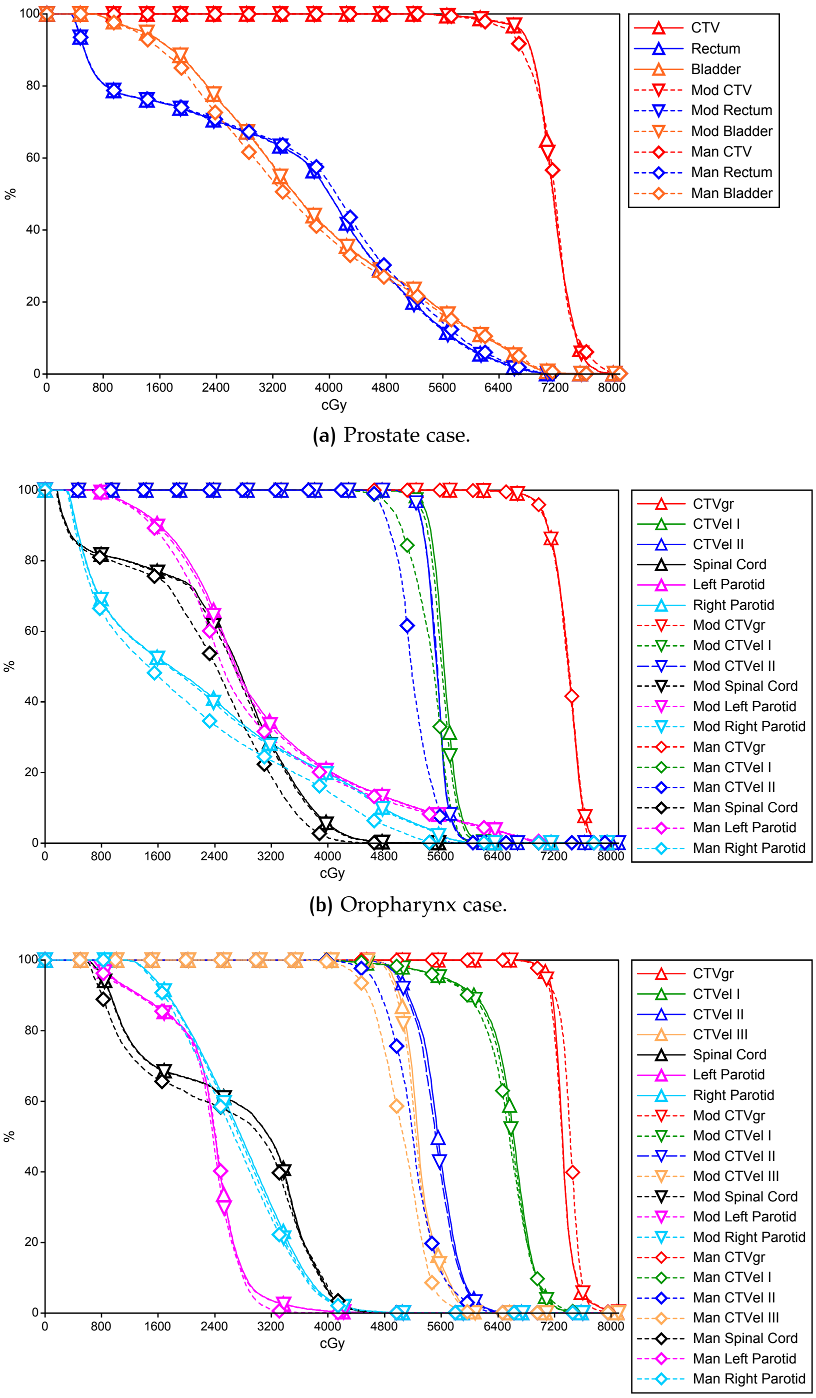
2. The dose delivered to OARs is always equal or lower, but never greater than its original dose.

3. In complex cases, manually discarding segments for achieving the same reduction is only possible at the expense of a substantial degradation of the dose delivered to CTVs, as it can be seen in figure 29.

In addition, the similarity between the original and modified plans suggest that the number of elements modified and the degree of modification for each algorithm showed in table 8 is enough for reducing the NS and the treatment time, but not for altering the plan quality.

## 4.7 CONCLUSIONS

This chapter presented a method for reducing the NS in unidirectional MLC segmentations. This reduction is achieved by modifying few units a small set of key positions in the fluence matrix of each beam. This processing was successfully applied to three real cases, reducing their treatment time without degrading the DVH.



**(a)** Prostate case.

**(b)** Oropharynx case.

**(c)** Larynx case.

**Figure 29:** DVHs for the rod pushing algorithm. Mod = modified matrix, Man = manually modified segmentation.

In general, radiotherapy sessions are time limited and IMRT treatment plans are manually modified in order to simplify them without compromising their quality. The proposed method allows to automatically reduce the number of segments, thus making easier the radiation oncologist work on modifying plans to fit them in the available time slot.

While developing the proposed method in this chapter, we realized that common techniques for reducing treatment time and complexity, like fluence smoothing at the optimization phase or the proposed method at the segmentation phase, usually can not obtain time reductions beyond 20% and, what is most important from the radiation oncologist point of view, that the final NS can not be *a priori* fixed, which would allow to directly generate plans for a given time slot. For this reason, the next step is the research on new methods that allow to control in some way the number of apertures obtained or, at least, increase the current treatment time reduction. The next chapter deals with this problem.

# 5 FIXED NUMBER OF SEGMENTS IN UNIDIRECTIONAL SEGMENTATIONS

## Contents

## 5.1 ABSTRACT

The decomposition of a fluence matrix in step-and-shoot mode for IMRT usually yields a large NS and, consequently, treatment time is substantially increased. In this chapter, we propose a method for reducing the original NS in multileaf collimator segmentations to a user-specified quantity. The proposed method clusters original segments into the same number of groups as desired NS, and computes for each group an equivalent segment and an associated weight. In order to avoid important changes in DVHs, equivalent segments and weights are computed taking into account the original fluence matrix and preserving the highest fluence zones, thus staying as close as possible to the original planned radiation. The method is applicable to unidirectional segmentations, where there is no backtracking of leaves, since this property facilitates the grouping of segments. The experiments showed that treatment times can be considerably reduced, while maintaining similar DVHs and dosimetric indexes. Furthermore, the algorithm achieved an excellent reduction/dose-quality ratio since the final NS was close to that reported for direct step-and-shoot solutions.

## 5.2 INTRODUCTION

Step-and-shoot (or static) mode in IMRT was originally devised with an optimization phase for planning the dose to be delivered by each beam as a fluence matrix, and a segmentation phase for decomposing this matrix into a feasible set of MLC segments [25]. One known drawback of this approach is that optimization algorithms often generate very heterogeneous fluence matrices [59, p 2105], whose corresponding segmentations yield large NS and large TNMU, and consequently treatment time is substantially increased [55, p 1007].

This drawback has been overcome with techniques that process fluence matrix values such as smoothing [2, 45, 59, 63], clustering [8, 66] or segmentation-driven smoothing [46]. However, the results of previous methods in terms of delivery efficiency have been surpassed by DSS solutions such as direct aperture optimization [55], direct machine parameter optimization [31, 41] or graph-based aperture optimization [13]. These approaches combine optimization and segmentation into a single phase by directly optimizing MLC leaf positions instead of fluence matrices. Their efficiency derives from the fact that the NS can be *a priori* fixed and, as a consequence, treatment time and complexity can be considerably reduced.

Methods with the ability of fixing the NS are advantageous, since the plans obtained are simple with compact and large apertures, bigger associated weights, but fewer TNMU than two-phase plans. This reduction in complexity allows (1) obtaining a short treatment time, thus patient comfort as well as throughput of patients can be improved; (2) decreasing leakage exposure, that reduces the risk of collateral effects and radiation-induced secondary cancers [11, 13, 50]; and (3) obtaining a plan that is easier to deliver [53, p 2719]. For all these reasons, the possibility of *a priori* fixing the NS would be highly desirable in two-phase step-and-shoot IMRT.

In this chapter, we propose a method for post-processing MLC segmentations that can be included in two-phase step-and-shoot IMRT treatment planning systems and allows to *a priori* fix the NS. This method is applicable to unidirectional MLC segmentations [4, 57], where leaves are moved in a single direction. Unidirectional leaf movement makes these segmentations very suitable inputs for our method, since the leaf arrangement provides highly correlated adjacent segments, as can be seen in figure 30. This facilitates their clustering into the same number of groups as desired NS, so as to generate for each group an equivalent segment and an associated weight, which is the basic idea of our method. In order to avoid substantial changes in DVHs, equivalent segments and their weights are computed taking into account the original fluence matrix and preserving the highest fluence zones, thus staying as close as possible to the original planned radiation. As reported in the experimental section, the proposed method has been shown to achieve an excellent reduction/dose-quality ratio since the method was able to reduce the original NS up to 75% without compromising plan quality.

This chapter is organized as follows. In section 5.3, we describe the proposed method, the experimental setup and the clinical cases used in section 5.4, where we present the numerical results, including treatment time measurements. Finally, the discussion and conclusions are presented in sections 5.5 and 5.6, respectively.

## 5.3 METHOD AND MATERIALS

The proposed method uses as input the decomposition of a fluence matrix, independently obtained by one of the unidirectional segmentation methods proposed in the literature such as Siochi 1999 or Artacho 2009, and the number of desired segments. The decomposition, or segmentation, of an $M \times N$ fluence matrix $A$ is defined as a set of pairs composed of an aperture (segment) plus an associated weight accounting for a relative beam-on time, $(S_k, \alpha_k)_{1 \leqslant k \leqslant K}$, in such a way that

$$A = \sum_{k=1}^{K} \alpha_k \cdot S_k \tag{5.1}$$

where $K$ is the NS and $k \in [1, ..., K]$ is the index representing the segment position in the original segmentation. The segments are subject to some constraints. We will consider one aperture per row and avoid interleaf collisions [57, p 672].

In our method, the processing of the original segmentation is divided into four steps. First, the original segments are clustered into as many groups as desired segments. Second, an equivalent segment $S^{eq}$ is generated for each group. Third, an associated weight $\alpha^{eq}$ is computed for each equivalent segment. These equivalent segments and weights are intended to obtain an approximation of the original segmentation in order to provide a simpler plan with similar quality

$$A = \sum_{k=1}^{K} \alpha_k \cdot S_k \approx \sum_{g=1}^{G} \alpha_g^{eq} \cdot S_g^{eq} \tag{5.2}$$

where $G$ is the number of desired segments and $g \in [1, ..., G]$ is the index representing the segment position in the processed segmentation. Finally, for each equivalent segment, it is checked in the overlap region with posterior segments that the fluence accumulated does not exceed that originally planned in matrix $A$. Otherwise, the posterior segments are modified to fulfil this requirement. These steps are detailed in the following subsections.

### 5.3.1 Clustering the original segments

The first step of our method consists of clustering the original segments from a fluence matrix decomposition into the same number of groups as the desired NS. Grouping is driven by similarity among the segments. To this end, we define the correlation between two segments $S_1$ and $S_2$ as

$$\sigma = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}(S_1(i,j) \wedge S_2(i,j))}{\max\left(\sum_{i=1}^{M}\sum_{j=1}^{N}S_1(i,j)\,,\ \sum_{i=1}^{M}\sum_{j=1}^{N}S_2(i,j)\right)} \tag{5.3}$$

Thus, $\sigma$ accounts for the overlap between both segments and their relative size. The $i \in [1, ..., M]$ and $j \in [1, ..., N]$ indexes are used to move through the rows and columns, respectively, of the fluence matrix or the original segment. The clustering procedure takes advantage of the similarity among adjacent segments in unidirectional segmentations measured by this correlation coefficient as follows.

First, clusters are uniformly initialized as groups of length round $(K/G)$. Second, for each group it is checked whether the last segment is more closely correlated (using equation 5.3) with the previous segment in this group or with the first segment in the next group. In the latter case, it becomes part of next group. Otherwise, the same test is applied to the first segment in the next group, in order to check whether it should become part of the current group. This exchange process is iteratively applied, allowing multiple changes while controlling group cardinalities. One or two iterations are usually enough to increase internal group correlation and exchange segments that were incorrectly assigned during cluster initialization. This number of iterations is intended to keep a balance between intergroup cardinality and intragroup correlation, thus avoiding groups with a relatively small NS. A very unbalanced distribution of segments would make some groups much larger than others and more difficult to represent with a single aperture, since the difference between the first segment and the last one may be considerable.

As an example, two iterations of the proposed procedure were applied to a fluence matrix segmentation of a prostate case with 25 segments clustered into six groups. The result is shown in figure 30.

### 5.3.2 Computing equivalent segments

After clustering, an equivalent segment $S^{eq}$ is computed for each group of segments. This computation is a weighted sum driven by the original fluence matrix values, in such a way that the equivalent segment shape will contain those leaf apertures that contribute to high radiation regions. Thus, the complexity of the segmentation is reduced, while keeping the delivered radiation as close as possible to the original planned one.

The weighted sum is based on the fact that the highest fluence values correspond to beamlets that radiate only a CTV, which should be included and not modified in the equivalent segment in order not to degrade the plan quality; whereas the lowest fluence values correspond to beamlets that radiate CTV and OAR at the same time, which are allowed to be modified or even excluded from the equivalent segment.

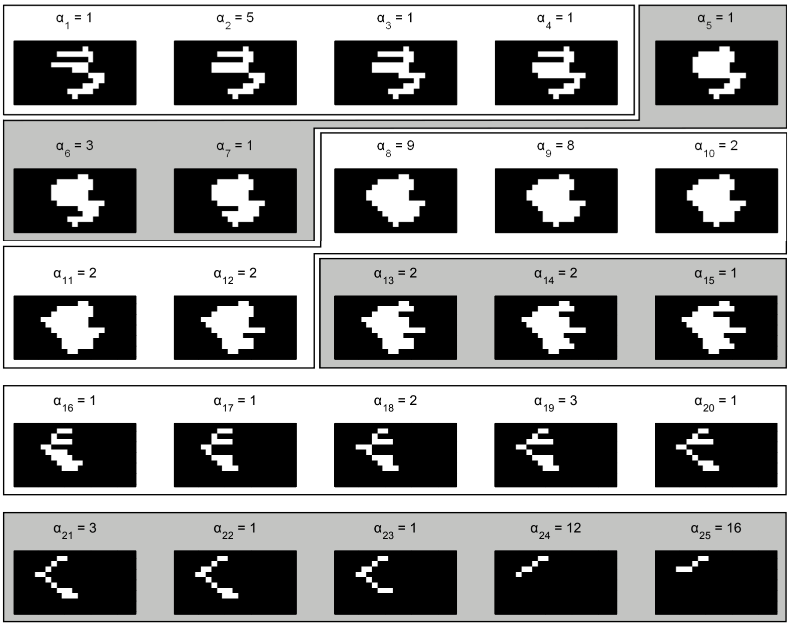


**Figure 30:** Fluence matrix segmentation of one beam from a prostate case with 25 segments, clustered into 6 groups. Final grouping is 4-3-5-3-5-5, whereas initial grouping was 4-4-4-4-4-5.

Therefore, we define a weighting matrix $W$ of size $M \times K$ in order to compute equivalent segments, where each column contains the sum of the original fluence values in each row between the left and right leaves denoted by $l_k$ and $r_k$, respectively, for the kth segment

$$W(i,k) = \sum_{j=l_k(i)+1}^{r_k(i)} A(i,j) \tag{5.4}$$

$$0 \leqslant l_k(i) \leqslant r_k(i) \leqslant N \tag{5.5}$$

$$i \in [1, ..., M], \ k \in [1, ..., K]$$

where positions $l_k(i) + 1$ to $r_k(i)$ are exposed to radiation, and left leaf at positions $[0, ..., l_k(i)]$ and right leaf at positions $[(r_k(i) + 1), ..., N]$ are blocking radiation. The case of a row totally closed is included as $l_k(i) = r_k(i)$.

When equivalent segments are computed, the corresponding leaves do not often match the beamlet positions. This is intensified by the use of a weighting matrix. Accordingly, it is necessary to redefine the segment representation used in equation 5.1 in order to deal with continuous leaf positions. Thus, the $S_k$ segment is now represented as a $M \times 2$ matrix of real numbers $S'_k$, where the first column contains the left leaf location $l_k$ and the second column contains the right leaf location $r_k$. From here on, any entity related to continuous leaf positions will be followed by an apostrophe $'$. This change of representation is illustrated in example 8 of 5.7. It should be noted that, although the step-and-shoot mode uses discrete leaf positions defined by the beamlets, this is not an MLC limitation, since the MLC is able to place leaves in continuous positions. Taking advantage of this feature, our method allows placing the leaves in any position even though this position does not match the horizontal discretization of the rows.

Using this new representation $S'$ for segments with continuous leaf positions, the generation of equivalent segment $S^{eq'}$ is performed by computing for all the leaves belonging to open rows

$$S_g^{eq'}(i,x) = \frac{\sum_{k=u_g}^{v_g}\left(W(i,k)\cdot S_k'(i,x)\right)}{\sum_{k=u_g}^{v_g} W(i,k)} \tag{5.6}$$
$$g \in [1,...,G], i \in [1,...,M],\ x \in [1,2],\ u_g, v_g \in [1,...,K]$$

where $u_g, v_g \in [1,...,K]$, and $u_g \leqslant v_g$, are the indexes in the original segmentation of the first and last segments for the gth group, respectively, and $x \in [1,2]$ is used to specify the leaf bank (left or right) of an MLC.

The final step is to ensure that $S^{eq'}$ is a feasible segment for the MLC used, otherwise it should be modified to meet the MLC constraints. In our current approach, one aperture per row is automatically generated in equation 5.6. Therefore, only the interleaf collision constraint is imposed by opening any pair of offending leaves until there is no collision. The whole process of computing an equivalent segment is illustrated in examples 7 and 8 of 5.7.

### 5.3.3 Computing associated weights

The weight associated with an equivalent segment, $\alpha^{eq'}$, is generated by accumulating the fluence delivered by the original group of segments and achieving a uniform delivery with the new equivalent segment area. For this purpose, we define the cumulative fluence matrix $A_g^{cu}$ of a group as the accumulation of the different segments previously multiplied by their corresponding weights

$$A_g^{cu} = \sum_{i=u_g}^{v_g} \alpha_i \cdot S_i \tag{5.7}$$
$$u_g, v_g \in [1..K],\ g \in [1..G]$$

Thus, $A_g^{cu}$ represents the contribution of the gth group to the fluence matrix $A$. Then, we define the weight $\beta_g'$ as the sum of the old fluence delivered by the group divided by the new equivalent segment area

$$\beta_g' = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} A_g^{cu}(i,j)}{\sum_{i=1}^{M}\left(S_g^{eq'}(i,2) - S_g^{eq'}(i,1)\right)} \tag{5.8}$$
$$g \in [1,...,G]$$

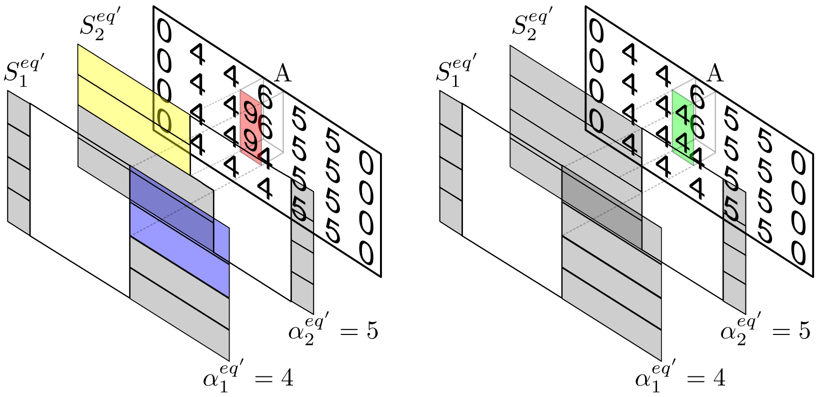
which is truncated to the maximum value found in $A_g^{cu}$, in order to prevent an overdose caused by a shrinking of the $S^{eq'}$ area compared to the original $A_g^{cu}$ area, yielding

$$\alpha_g^{eq'} = \max\left(\beta'_g, \max_{i=1}^{M}\left(\max_{j=1}^{N}\left(A_g^{cu}(i,j)\right)\right)\right) \qquad (5.9)$$
$$g \in [1, ..., G]$$

### 5.3.4 Checking the equivalent segment overlapping

Once equivalent segments and weights are computed, it remains to be checked that there is no region where several equivalent segments overlap and the fluence accumulated is higher than that originally planned in matrix A. In these cases, the spatial location and delivery order of the segments are used for solving this situation.

Let $S_1^{eq'}$ be the first equivalent segment obtained from an unidirectional segmentation in left-to-right direction. If there is an overlapping area with $S_2^{eq'}$, where $\alpha_1^{eq'} + \alpha_2^{eq'}$ exceeds the fluence planned, this situation is detected and fixed as follows. For each row $i \in [1..M]$ in $S_1^{eq'}$ with open leaves, the condition for overlapping is that $S_2^{eq'}(i,1)$ is smaller than $S_1^{eq'}(i,2)$. In this case, if the fluence added by $\alpha_2^{eq'}$ to $\alpha_1^{eq'}$ exceeds the maximum fluence value found in the original fluence matrix between both leaves, then $S_2^{eq'}(i,1)$ is moved forward until it reaches the location of $S_1^{eq'}(i,2)$. Figure 31 illustrates this example. The process has to be repeated comparing each segment with all the next ones until there is no overlapping.



(a) Initial solution.

(b) Overlap checking done.

**Figure 31:** Overlap checking example with two equivalent segments. A is the original fluence matrix. The right leaves $S_1^{eq'}(1,2)$ and $S_1^{eq'}(2,2)$ (in blue) will respectively cause the left leaves $S_2^{eq'}(1,1)$ and $S_2^{eq'}(2,1)$ (in yellow) to move forward in order to avoid the delivery of 9 MU instead of the original 6 MU.

### 5.3.5 Data and experimental setup

The experiments reported in this chapter were performed using (1) clinical cases planned with the PCRT 3D® treatment planning system, (2) two different unidirectional segmentation methods rod pushing (RP) [57] and OTNMU [4, p 577], (3) a Siemens ONCOR™ linear accelerator with an Optifocus™ MLC for obtaining beam delivery times.

Results were obtained under the following conditions:

1. The constraints used for Siemens MLC were one aperture per row and interleaf collision.

2. Unidirectional segmentations were performed from left to right and from right to left and the best solution was selected.

3. Unless otherwise stated, the applied reduction was one equivalent segment for each four original ones (4 : 1)[7], in order to have a good reduction ratio without considerably modifying the original DVHs. This condition was applied to all beams, independently of any criteria such as CTV and OAR positions, initial NS, or fluence matrix heterogeneity. However, it is possible to fix for each beam a different reduction ratio.

4. The number of iterations used for exchanging segments while clustering into groups was 2.

5. All plans were generated with a photon energy of 6 MV for each patient and normalized so as the mean dose of the main target volume contour is equal to the prescribed dose.

We present detailed results achieved by the method in three clinical cases. The first case is a prostate cancer radiated from five coplanar and equiangular beams: 36°, 108°, 180°, 252° and 324°, in a 72 Gy plan. The dose-volume constraint used for the rectum and bladder was 70% of the volume receives $\leqslant$ 40% of the goal dose.

The second case is an oropharynx cancer planned using seven coplanar, but not equiangular, beams: 20°, 60°, 100°, 180°, 260°, 300° and 340°. This case has three CTVs; the prescribed doses for the CTVgr and for the CTVel were 74 Gy and 54 Gy, respectively. The dose-volume constraint for the spinal cord was maximum dose $\leqslant$ 45 Gy, and the constraint for both parotids was 50% of the volume receives $\leqslant$ 40% of the prescribed dose to the CTVgr.

The third case is a larynx cancer treated with a seven coplanar beam plan with beam angles, prescribed doses and constraints identical to the oropharynx case, with the exception of including three CTVel with the following prescribed doses: 66 Gy, 56 Gy, and 50 Gy.

Additionally, we planned ten cancer cases in different body locations for including the dosimetric index results obtained in the main target volume. In these cases, only the number of beams and the prescribed dose for the main target volume were included in table 14.

[7] The reduction ratio was selected starting from the minimum possible reduction of 50% (2 : 1) and increasing it (3 : 1, 4 : 1, etc.) as long as the DVH remains similar to the original histogram using as criteria a change $<$ 5% in $D_{95}$ and $D_{105}$ for the main target volume.

## 5.4 RESULTS

In order to provide an example of the results achieved, we applied our method to the segmentation shown in figure 30, which corresponds to the 252° beam of the prostate case. The result can be seen in figure 32.

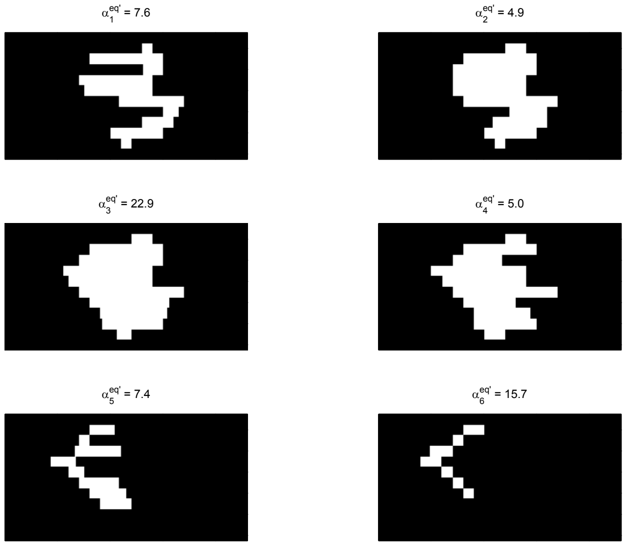


**Figure 32:** Prostate segmentation of figure 30 processed for obtaining six segments.

Table 10 summarizes the NS and TNMU results for the original and the processed plans. The columns referring to the latter have the prefix 'f' (meaning fixed). The NS reduction is not shown because it was 75% ± 0.67, whereas the MU reduction is explicitly reported since it was more variable. As an example of the time reduction achieved, the beam delivery times for the prostate and larynx cases, which had the smallest and biggest NS, are shown in table 11 for the RP. For the sake of brevity and conciseness, the results for the OTNMU algorithm are only presented in table 10, because they were very similar to those of the RP in the measurements.

The dosimetric comparison between each original plan and its corresponding processed plan was performed using a DVH. The DVHs for the three detailed cases are shown together in figure 33. In addition, we used the equivalent uniform dose (EUD), as described and implemented in Gay and Niemierko 2007, the $D_{95}$ and the $D_{100}$ indexes in order to quantify the dosimetric differences between both plans. The results of these indexes can be seen in table 12 together with the $a$ parameter used in the EUD formula for each ROI. Table 12a also includes a study of change in EUD, $D_{95}$ and $D_{100}$ as a function of the NS reduction for the prostate case. The ratios ranged from 2:1 to 6:1.

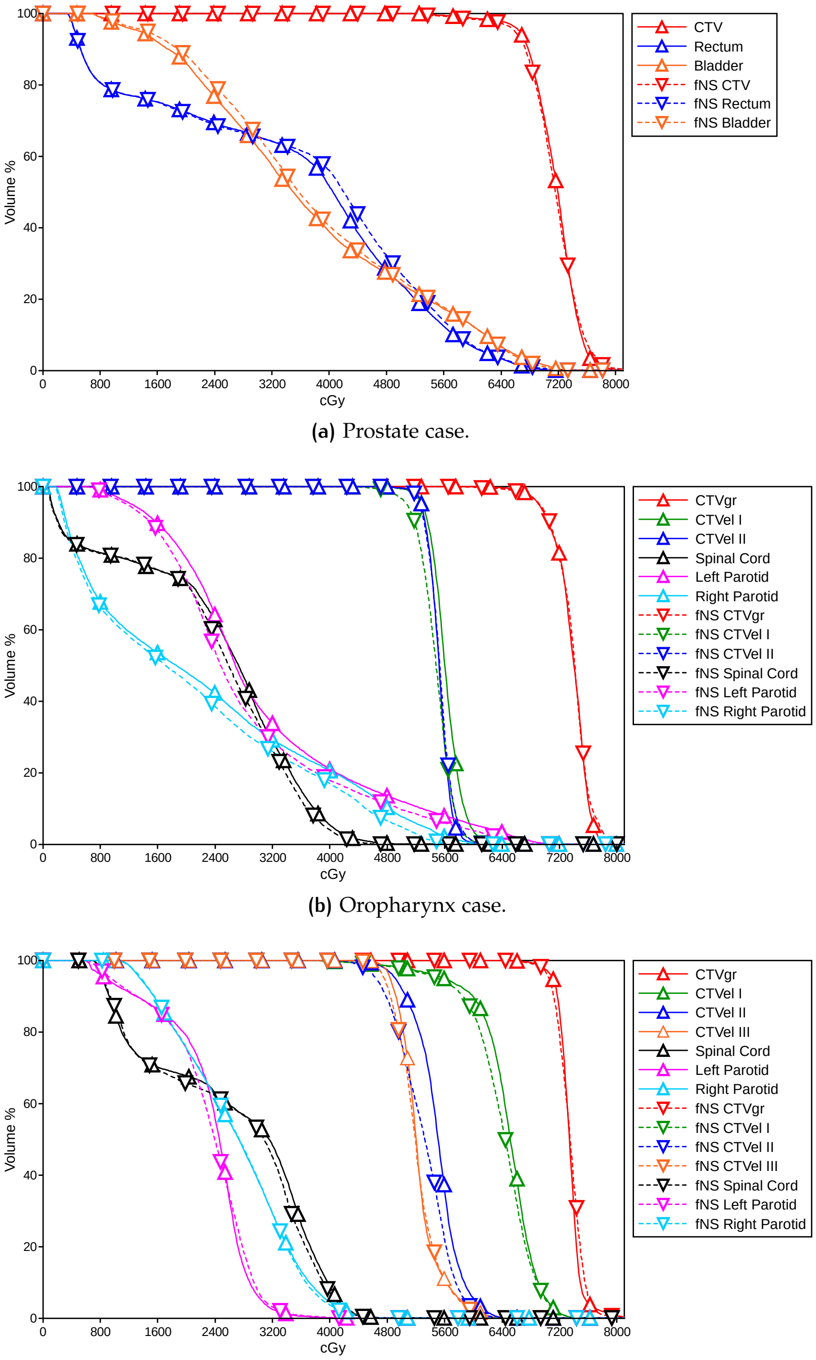
Additionally, we implemented and used the modulation index (MI) described in Webb 2003 for assessing how the complexity of the treatment plan varies between the original plan, with an unrestricted NS, and the fixed NS approach. The MI results are presented in table 13

**Table 10:** NS and TNMU results for the original and the fixed NS plans (4 : 1 ratio). fNS = fixed NS, fMU = MU for fNS, %MU = MU reduction percentage.

| | RP | | | | | OTNMU | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Beam | NS | fNS | MU | fMU | %MU | NS | fNS | MU | fMU | %MU |
| **(a)** Prostate case. | | | | | | | | | | |
| 36° | 37 | 9 | 76 | 68 | 10.53 | 43 | 11 | 76 | 73 | 3.95 |
| 108° | 29 | 7 | 97 | 79 | 18.56 | 29 | 7 | 97 | 61 | 37.11 |
| 180° | 47 | 12 | 95 | 92 | 3.16 | 49 | 12 | 95 | 84 | 11.58 |
| 252° | 25 | 6 | 82 | 63 | 23.17 | 27 | 7 | 82 | 67 | 18.29 |
| 324° | 41 | 10 | 109 | 92 | 15.60 | 43 | 11 | 109 | 80 | 26.61 |
| Total | 179 | 44 | 459 | 394 | 14.16 | 191 | 48 | 459 | 365 | 20.48 |
| **(b)** Oropharynx case. | | | | | | | | | | |
| 260° | 28 | 7 | 49 | 30 | 38.78 | 30 | 8 | 49 | 45 | 8.16 |
| 300° | 40 | 10 | 49 | 43 | 12.24 | 44 | 11 | 49 | 45 | 8.16 |
| 340° | 51 | 13 | 68 | 55 | 19.12 | 53 | 13 | 68 | 54 | 20.59 |
| 20° | 52 | 13 | 74 | 65 | 12.16 | 53 | 13 | 77 | 58 | 24.68 |
| 60° | 32 | 8 | 38 | 29 | 23.68 | 27 | 7 | 38 | 30 | 21.05 |
| 100° | 29 | 7 | 45 | 27 | 40.00 | 30 | 8 | 46 | 29 | 36.96 |
| 180° | 46 | 12 | 55 | 51 | 7.27 | 45 | 11 | 56 | 49 | 12.50 |
| Total | 278 | 70 | 378 | 300 | 20.63 | 282 | 71 | 383 | 310 | 19.06 |
| **(c)** Larynx case. | | | | | | | | | | |
| 260° | 42 | 11 | 58 | 47 | 18.97 | 43 | 11 | 60 | 51 | 15.00 |
| 300° | 53 | 13 | 66 | 57 | 13.64 | 55 | 14 | 67 | 55 | 17.91 |
| 340° | 57 | 14 | 70 | 59 | 15.71 | 58 | 15 | 70 | 59 | 15.71 |
| 20° | 43 | 11 | 66 | 55 | 16.67 | 43 | 11 | 66 | 53 | 19.70 |
| 60° | 38 | 10 | 62 | 50 | 19.35 | 44 | 11 | 62 | 49 | 20.97 |
| 100° | 58 | 15 | 85 | 72 | 15.29 | 59 | 15 | 85 | 66 | 22.35 |
| 180° | 59 | 15 | 70 | 61 | 12.86 | 60 | 15 | 77 | 58 | 24.68 |
| Total | 350 | 89 | 477 | 401 | 15.93 | 362 | 92 | 487 | 391 | 19.71 |

**Table 11:** Beam delivery time table (in mm:ss) for the rod pushing technique in the original and the fixed NS plans (4 : 1 ratio). Beam order is the delivery order. 'Trans' is the transition time spent from the previous beam to the current one. 'Rad' is the total beam delivery time (beam-on time plus the time required for the leaves to move between segments).

| | Original | | Fixed | |
|---|---|---|---|---|
| Beam | Trans | Rad | Trans | Rad |
| **(a)** Prostate case. | | | | |
| 180° | 00 : 00 | 03 : 10 | 00 : 00 | 01 : 44 |
| 252° | 00 : 26 | 02 : 04 | 00 : 30 | 01 : 02 |
| 324° | 00 : 25 | 03 : 08 | 00 : 31 | 01 : 39 |
| 36° | 00 : 25 | 02 : 35 | 00 : 25 | 01 : 19 |
| 108° | 00 : 27 | 02 : 21 | 00 : 25 | 01 : 17 |
| Total | 01 : 43 | 13 : 18 | 01 : 51 | 07 : 01 |
| **(b)** Larynx case. | | | | |
| 180° | 00 : 00 | 02 : 44 | 00 : 00 | 01 : 15 |
| 100° | 00 : 25 | 03 : 19 | 00 : 19 | 01 : 29 |
| 60° | 00 : 19 | 03 : 29 | 00 : 19 | 01 : 27 |
| 20° | 00 : 19 | 02 : 47 | 00 : 20 | 01 : 17 |
| 340° | 00 : 21 | 02 : 41 | 00 : 19 | 01 : 17 |
| 300° | 00 : 21 | 03 : 38 | 00 : 25 | 01 : 42 |
| 260° | 00 : 29 | 03 : 38 | 00 : 28 | 01 : 32 |
| Total | 02 : 14 | 22 : 16 | 02 : 10 | 09 : 59 |



**(a)** Prostate case.

**(b)** Oropharynx case.

**(c)** Larynx case.

**Figure 33:** DVHs for the rod pushing algorithm. fNS = fixed NS.

**Table 12:** Dosimetric index comparison between the original and the fixed NS plans (4 : 1 ratio) using the EUD, $D_{95}$ and $D_{100}$. $D_x$ is defined as the percentage of volume receiving x% of the prescribed dose.

(a) The EUD, $D_{95}$ and $D_{100}$ are included as a function of the NS reduction for the prostate case. The total NS for each plan is below its ratio between brackets.

| | | | Original | 2 : 1 | 3 : 1 | 4 : 1 | 5 : 1 | 6 : 1 |
|---|---|---|---|---|---|---|---|---|
| ROI | Index | *a* | (179) | (92) | (60) | (44) | (35) | (30) |
| CTV | EUD | −10 | 70.07 Gy | 69.86 Gy | 70.04 Gy | 69.53 Gy | 69.52 Gy | 69.47 Gy |
| | $D_{95}$ | | 85.72% | 84.61% | 85.42% | 83.56% | 80.04% | 75.51% |
| | $D_{100}$ | | 48.68% | 44.15% | 47.01% | 45.21% | 44.79% | 45.97% |
| Rectum | EUD | 6 | 48.00 Gy | 47.98 Gy | 48.86 Gy | 48.82 Gy | 49.55 Gy | 50.26 Gy |
| Bladder | EUD | 6 | 49.72 Gy | 49.50 Gy | 50.09 Gy | 49.71 Gy | 50.47 Gy | 51.03 Gy |

Table 12: Dosimetric index comparison between the original and the fixed NS plans (4 : 1 ratio) using the EUD, $D_{95}$ and $D_{100}$. $D_x$ is defined as the percentage of volume receiving x% of the prescribed dose.

| ROI | Index | $a$ | Original | Fixed |
|---|---|---|---|---|
| **(b)** Oropharynx case. | | | | |
| CTVgr | EUD | −10 | 73.20 | 72.92 |
| | $D_{95}$ | | 93.78 | 90.40 |
| | $D_{100}$ | | 49.00 | 46.65 |
| CTVel I | EUD | −10 | 56.07 | 54.29 |
| CTVel II | EUD | −10 | 55.04 | 54.37 |
| Spinal cord | EUD | 13 | 35.77 | 33.59 |
| Left parotid | EUD | 0.5 | 18.69 | 17.59 |
| Right parotid | EUD | 0.5 | 29.19 | 28.26 |
| **(c)** Larynx case. | | | | |
| CTVgr | EUD | −10 | 73.05 | 72.91 |
| | $D_{95}$ | | 97.58 | 93.37 |
| | $D_{100}$ | | 22.54 | 35.94 |
| CTVel I | EUD | −10 | 60.51 | 59.23 |
| CTVel II | EUD | −10 | 54.46 | 52.83 |
| CTVel III | EUD | −10 | 52.41 | 51.12 |
| Spinal cord | EUD | 13 | 36.00 | 35.06 |
| Left parotid | EUD | 0.5 | 27.05 | 26.98 |
| Right parotid | EUD | 0.5 | 22.47 | 21.78 |

for the original and the new fluence matrices. The latter was a reconstruction using the segmentation obtained from the proposed method and adjusting the leaves to the original beamlet positions with a round function, since this index was designed for discrete fluence matrices.

Lastly, table 14 provides the EUD, the $D_{95}$ and the $D_{100}$ dosimetric indexes for the main target volume in ten cancer cases using again a ratio of 4 : 1 for fixing the NS.

## 5.5 DISCUSSION

The rationale behind the proposed method for reducing the NS is to provide the radiation oncologist the possibility of having some control over the NS obtained in two-phase step-and-shoot IMRT, as has been done in DSS or direct IMRT approaches based on class solutions and patient anatomy [3, 20].

For all the cases presented in section 5.4, we fixed the NS with a ratio of 4:1 for all beams, since this was the biggest reduction that did not considerably modify the DVHs using as criteria a change $< 5\%$ in $D_{95}$ and $D_{105}$ for the main target volume. For the three cases reported in detail, the number of apertures obtained can be seen in table 10. These numbers are much closer to the results reported for DSS methods in similar cases [11, 13, 23, 37], which are between five and ten apertures per beam, than to two-phase IMRT systems. The MI results in table 13

Table 13: Modulation index for each beam in the original and the fixed NS plans (4 : 1 ratio).

(a) Prostate case.

| Beam | Original | Fixed |
|---|---|---|
| 36° | 2.18 | 1.86 |
| 108° | 2.08 | 1.63 |
| 180° | 2.15 | 1.94 |
| 252° | 2.33 | 2.19 |
| 324° | 2.31 | 1.53 |
| Average | 2.21 | 1.83 |

(b) Oropharynx and Larynx cases.

| | Oropharynx | | Larynx | |
|---|---|---|---|---|
| Beam | Original | Fixed | Original | Fixed |
| 180° | 5.18 | 3.11 | 6.00 | 5.06 |
| 100° | 5.18 | 4.29 | 4.61 | 3.99 |
| 60° | 4.24 | 3.39 | 5.40 | 4.05 |
| 20° | 5.13 | 4.56 | 4.61 | 3.66 |
| 340° | 5.72 | 4.64 | 5.09 | 4.85 |
| 300° | 6.63 | 4.33 | 5.49 | 4.70 |
| 260° | 5.60 | 4.33 | 5.60 | 4.64 |
| Average | 5.38 | 4.09 | 5.26 | 4.42 |

showed that there is also an important simplification of the fluence delivered and treatment complexity when using the fixed NS approach. In addition, the TNMU is reduced more than 14% in all cases, and this NS and TNMU reduction considerably decreases treatment times. As an example of the time-saving effect that can be achieved, the total delivery time for the RP technique, without taking into account beam transitions, is reduced by 47.2% for the prostate case and by 55.1% for the larynx case, as can be seen in table 11. These measurements were obtained using an Optifocus$^{\text{TM}}$ MLC with a negligible verification and V&R overhead (no delays between segments, apart from the leaf travel time itself). This means that experiments were performed under the most unfavourable conditions, i.e., any MLC with a V&R $\geqslant 1$ s can obtain greater time reductions.

The DVHs in figure 33 show that the method is able to reduce the original NS in our plans while (1) keeping the dose delivered to the main CTV close to its original one, (2) the OAR histogram curves are very similar, and (3) the maximum dose delivered to OARs is never significantly increased. In addition, the EUD for each ROI and plan presented in table 12 also suggests that there are no substantial modifications, in dosimetric terms, between the original and the fixed NS approaches. Similar results can be seen in table 14 for the main target volume in ten cancer cases, where the difference in EUD between both plans is smaller than 1 Gy. The observed changes for $D_{100}$ in table 14 are explained by the fact that the curve steepness is often slightly mod-

**Table 14:** Comparison of the EUD, $D_{95}$ and $D_{100}$ for the main target volume in ten additional cancer cases between the original and the fixed number of segments approaches (4 : 1 ratio).

| | | | Goal Dose | | EUD | $D_{95}$ | $D_{100}$ |
|---|---|---|---|---|---|---|---|
| ID | Location | Beams | (Gy) | Approach | (Gy) | (%) | (%) |
| 1 | Endometrium | 6 | 18.00[a] | Original | 16.62 | 85.90 | 54.61 |
| | | | | Fixed | 16.03 | 83.19 | 50.67 |
| 2 | Prostate | 6 | 78.00 | Original | 77.33 | 91.61 | 55.63 |
| | | | | Fixed | 77.11 | 88.06 | 57.42 |
| 3 | Prostate | 5 | 78.00 | Original | 76.70 | 96.27 | 5.61 |
| | | | | Fixed | 76.29 | 92.03 | 19.11 |
| 4 | Prostate | 6 | 76.00 | Original | 75.96 | 97.70 | 59.17 |
| | | | | Fixed | 75.57 | 93.97 | 54.67 |
| 5 | Prostate | 6 | 74.00 | Original | 72.38 | 93.32 | 18.86 |
| | | | | Fixed | 72.22 | 88.18 | 25.89 |
| 6 | Head-and-neck | 7 | 68.40 | Original | 67.86 | 96.25 | 40.03 |
| | | | | Fixed | 68.01 | 91.60 | 59.23 |
| 7 | Head-and-neck | 8 | 70.00 | Original | 69.40 | 90.71 | 57.15 |
| | | | | Fixed | 68.72 | 86.05 | 45.86 |
| 8 | Head-and-neck | 6 | 66.00 | Original | 65.96 | 96.16 | 57.56 |
| | | | | Fixed | 66.10 | 95.69 | 60.39 |
| 9 | Pancreas | 6 | 10.00[b] | Original | 10.14 | 99.79 | 66.98 |
| | | | | Fixed | 9.84 | 92.19 | 23.70 |
| 10 | Pelvis & Sacrum | 5 | 50.40 | Original | 52.44 | 99.89 | 97.92 |
| | | | | Fixed | 53.21 | 99.82 | 95.56 |

[a] Twice a day, hyper-fractionation.
[b] Re-irradiation.

ified and the kind of normalization applied tends to preserve the $D_{95}$ index. As a consequence, the $D_{100}$ index is prone to suffer variations.

As expected, the larger the NS reduction for all the beams, the greater is the difference between the original and the final dose, as reported in table 12a for the prostate case. During the experimental stage we found that reductions beyond the ratio 5:1 may cause modifications in the DVH that would not be acceptable in many cases. We observed that this behaviour is due to the difficulty of representing with a single aperture a group with five segments or more, because the difference between the first segment and the last one within the group may be considerable.

## 5.6 CONCLUSIONS

The method presented in this chapter is able to reduce the NS in two-phase step-and-shoot IMRT treatment planning systems to an *a priori* fixed value. This NS reduction is computed by clustering the original segments into groups, and creating an equivalent segment with its associated weight for each group.

The results of the testing in clinical cases show that final segmentation with a reduction in the NS up to 75% obtained a DVH and dosimetric indexes very similar to the original ones, so the plan quality was not compromised. In addition, the TNMU was also decreased and both NS and TNMU reductions considerably shortened treatment times.

## 5.7 EQUIVALENT SEGMENT COMPUTATION EXAMPLE

*Example* 7. The $W$ weighting matrix is computed for a random fluence matrix and its unidirectional segmentation. The original segments (with their left and right leaves shaded in light and dark grey, respectively)

$$A = \begin{bmatrix} 0 & 2 & 4 \\ 1 & 2 & 1 \\ 2 & 4 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

are projected over the original fluence matrix $A$

$$\begin{bmatrix} 0 & 2 & 4 \\ 1 & 2 & 0 \\ 2 & 4 & 0 \end{bmatrix} ; \begin{bmatrix} 0 & 2 & 4 \\ 0 & 2 & 0 \\ 2 & 4 & 0 \end{bmatrix} ; \begin{bmatrix} 0 & 0 & 4 \\ 0 & 0 & 0 \\ 0 & 4 & 0 \end{bmatrix} ; \begin{bmatrix} 0 & 0 & 4 \\ 0 & 0 & 1 \\ 0 & 4 & 1 \end{bmatrix} ,$$

and the weighting matrix is obtained applying equation 5.4

$$W = \begin{bmatrix} 6 & 6 & 4 & 4 \\ 3 & 2 & 0 & 1 \\ 6 & 6 & 4 & 5 \end{bmatrix} .$$

*Example* 8. Let us assume example 7 segmentation is reduced to two segments, and let us assume the first group has three segments and the second group has the last segment. Accordingly,

$$A_1^{cu} = \begin{bmatrix} 0 & 2 & 3 \\ 1 & 2 & 0 \\ 2 & 3 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} .$$

Then, the $S_1^{eq'}$ equivalent segment is computed porting the segments to the continuous leaf position representation

$$\begin{bmatrix} 1.0 & 3.0 \\ 0.0 & 2.0 \\ 0.0 & 2.0 \end{bmatrix} + \begin{bmatrix} 1.0 & 3.0 \\ 1.0 & 2.0 \\ 0.0 & 2.0 \end{bmatrix} + \begin{bmatrix} 2.0 & 3.0 \\ 2.0 & 2.0 \\ 1.0 & 2.0 \end{bmatrix} ,$$

and applying equation 5.6 with example 7 weighting matrix

$$S_1^{eq'} = \begin{bmatrix} 1.2 & 3.0 \\ 0.4 & 2.0 \\ 0.2 & 2.0 \end{bmatrix} .$$

# 6 CONTRIBUTIONS, CONCLUSIONS AND PERSPECTIVES

## Contents

In this final chapter, it is summarized the main contributions and conclusions derived from the work developed in this thesis, together with the description of future extensions and possible alternative applications.

## 6.1 CONTRIBUTIONS AND CONCLUSIONS

The contribution of this thesis focuses on the development of novel methods for the SMLC mode of IMRT in order to improve the modelling of the problems solved or reduce the treatment delivery times. In addition, the proposed methods were devised for being incorporated in the daily clinical use of IMRT at hospitals.

Specifically, the algorithm for computing the relationship between beamlets and voxels:

1. improves the modelling of the radiation, since it makes the relationship between both entities become n-to-n, in opposition to the former one-to-n relationship assumed by the original method used in the treatment planning system.

2. allows to independently choose the voxel size and the beamlet width, in contrast with the original method that forces the user to choose a voxel size smaller than the beamlet width in order to properly model the radiation. As a result, the voxel size can be increased, thus reducing the number of voxels per organ and consequently the dose matrix size. Therefore, the optimization of the beamlet intensities becomes faster due to the smaller number of computations needed for obtaining the dose distribution achieved inside the patient.

3. was implemented using a compute-by-drawing approach with the cross-platform library OpenGL for fast computing the projection of thousands of voxels on the beamlet grid, instead of the former voxel center point projection, without spending more than a few seconds.

Regarding the methods developed for the MLC segmentation step, the novel unidirectional segmentation method:

1. decomposes fluence maps in segments with compact and regular shapes and offers the possibility of selecting between the minimization of the NS or the TNMU, whereas MLC segmentation methods can not control the segment shape and they are only able to minimize one fixed criterion.
2. relaxes the tough requirements imposed to the hardware used for the treatment delivery, since it tends to generate only one connected component (aperture) per segment with a smoother outline.
3. is as efficient as any other unidirectional method despite the leaf synchronization introduced for controlling the aperture shape.

The method for reducing the NS in unidirectional segmentations:

1. performs a segmentation-driven fluence smoothing for achieving a reduction up to 25% in the NS that is translated into a 16% decreasing of the beam delivery time.
2. does not substantially change the DVHs. Thus, the quality of the original and the new plan are equivalent.

Finally, the post-processing algorithm of unidirectional segmentations:

1. generates solutions with a user-specified number of apertures, where a reduction of 75% in the original NS shortened the beam delivery time up to 55%.
2. obtains a DVH and dosimetric indexes (EUD and $D_{95}$) very similar to the original ones, so the plan quality is not compromised.
3. simplifies the fluence map and the treatment complexity. Thus, making the clinical validation of the treatment easier for the radiation oncologist.

## 6.2 PERSPECTIVES AND FUTURE WORK

The methods developed in this thesis have opened some possibilities that will be explored in future work. The most interesting ones are presented below.

### 6.2.1 Radiation model

- **Complete implementation of the voxel projection method in the GPU using OpenCL.** This would make the method even faster than the original.
- **Explore the sub-beamlet precision feature.** The accuracy of the calculations for obtaining the radiation dose deposited in each

voxel may be increased if the sub-beamlet precision feature of the projection method is used when applying the radiation spreading model.

- **Extend the algorithm to other techniques of external radiotherapy.** The voxel projection method could be, for example, applied in a rotational radiation therapy technique like arc-modulated radiation therapy (AMRT) [61].

### 6.2.2 MLC segmentation

- **The shape constraint can be extended to other aperture generators.** The idea of penalizing any leaf far from its adjacent neighbours introduced in the unidirectional MLC segmentation method is applicable to other algorithms that generate apertures, such as DAO or, in general, any DSS method.
- **Apply both NS reduction methods to one solution.** The segmentation-driven smoothing algorithm described in chapter 4 could be applied prior the post-processing method of chapter 5 in order to reduce the original NS. The first method will indirectly reduce the number of elements in each group. Therefore, it will be easier for the second method to generate the equivalent segment. As a consequence, it would be also possible to reduce the dosimetric differences between the original and the new plan observed after the post-processing.

### 6.2.3 General

- **Extend the use of the GPU to other parts of the pipeline for obtaining a treatment plan.** The dose matrix is multiplied by the current beamlet intensities after each iteration in the optimization process. This task can be performed much faster in a GPU than in a CPU because the inherent parallelism of the operation. Other parts of the code may also benefit from a parallelization in the computer graphic card.
- **Development of a DSS method.** Many of the problems found in IMRT regarding accuracy and treatment complexity come from the fact that optimization and segmentation of the intensity maps are separated tasks. Optimizing leaf positions instead of beamlet intensities directly overcomes many problems, such as the optimization regularization or an efficient MLC segmentation. Using the knowledge gained, one of the most challenging, and at the same time enjoyable, tasks would be the design and implementation of a DSS method.
- **Research on alternatives to the beamlet and voxel discretization approach.** In a large scale problems of rotation therapy like a AMRT plan with 100 beams, the traditional radiation model would require huge amounts of memory space and computational resources to deal with the relationship between beamlets and voxels for computing the dose distribution. There are non-voxel and

non-beamlet based approaches where objective functions and derivatives are evaluated based on the continuous viewpoint, abandoning voxel and beamlet discretizations and representations [43]. This kind of alternatives are probably a must for implementing a fast and extensible treatment planning system targeting rotation therapy.

# CONCLUSIONES

La contribución de esta tesis se centra en el desarrollo de nuevos métodos para el modo SMLC de IMRT, con el objetivo de mejorar el modelado de los problemas resueltos o reducir el tiempo de administración de los tratamientos. Además, los métodos propuestos fueron concebidos para ser incorporados en la rutina clínica diaria del uso de IMRT en los hospitales.

En concreto, el algoritmo para el cálculo de la relación entre beamlet y vóxels:

1. mejora el modelado de la radiación al permitir que la relación entre ambas entidades sea *n* a *n*, en contraposición con la relación uno a *n* que asumía el método original usado en el software de planificación.

2. permite seleccionar independientemente el tamaño del vóxel y la anchura del beamlet, al contrario que ocurría en el método original donde el usuario se veía forzado a especificar un tamaño de vóxel más pequeño que la anchura del beamlet para poder modelar correctamente la radiación. Como resultado, el tamaño de los vóxeles puede ser aumentado, lo que provoca la disminución del número de vóxels por órgano y por consiguiente del tamaño de la matriz de dosis. Por lo tanto, la optimización de la intensidad de los beamlets se ejecuta de forma más rápida debido a que es necesario un menor tiempo para calcular la distribución de dosis dentro del paciente.

3. fue implementado usando un planteamiento de cálculo mediante dibujo, utilizando la biblioteca multiplataforma OpenGL, para el cálculo rápido de la proyección completa de miles de vóxeles sobre la rejilla de beamlets, en vez de proyectar únicamente el punto central del vóxel, sin emplear más de unos pocos segundos.

Respecto a los métodos desarrollados para la segmentación MLC, el nuevo método unidireccional de segmentación:

1. descompone los mapas de fluencia en segmentos con formas compactas y suaves, ofreciendo la posibilidad de seleccionar entre la minimización del NS o del TNMU, mientras que el resto de métodos de segmentación MLC no controlan la forma de las aperturas y solamente permiten la minimización de un criterio prefijado.

2. relaja los fuertes requerimientos impuestos al hardware de administración, ya que tiende a generar una única apertura por segmento con un contorno suave.

3. es tan eficiente como cualquier otro algoritmo de descomposición unidireccional a pesar de la sincronización de láminas introducida para controlar la forma de las aperturas.

El método para la reducción del NS en las segmentaciones unidireccionales:

1. realiza un suavizado de la fluencia basado en la segmentación para alcanzar una reducción de hasta un 25 % en el NS que se traduce en un 16 % menos de tiempo en la administración.

2. no modifica de forma sustancial los DVHs. Por lo tanto, la calidad del plan resultante es igual a la del original.

Finalmente, el algoritmo de postprocesado de segmentaciones unidireccionales:

1. genera soluciones con un número de segmentos fijado por el usuario, donde una reducción del 75 % sobre el NS original acorta los tiempos de administración hasta en un 55 %.

2. obtiene un DVH y unos índices dosimétricos (EUD y $D_{95}$) muy similares a los originales, por lo tanto la calidad del plan no se ve comprometida.

3. simplifica el mapa de fluencia y disminuye la complejidad del tratamiento. Por lo tanto, hace más fácil la validación clínica del tratamiento para el especialista en radiofísica hospitalaria.

# PUBLICATIONS

This thesis leads to the following journal publications:

- J. Artacho, X. Mellado, G. Tobías, S. Cruz, and M. Hernández. A novel unidirectional intensity map segmentation method for step-and-shoot IMRT delivery with segment shape control. *Phys Med Biol*, 54(3):569–589, Jan 2009. DOI: 10.1088/0031-9155/54/3/007. URL `http://dx.doi.org/10.1088/0031-9155/54/3/007`.
- X. Mellado, S. Cruz, J. M. Artacho, and M. Canellas. Reducing the number of segments in unidirectional MLC segmentations. *Phys Med Biol*, 55(3):N75–N85, Jan 2010. DOI: 10.1088/0031-9155/55/3/N01. URL `http://dx.doi.org/10.1088/0031-9155/55/3/N01`.
- X. Mellado, J. M. Artacho, M. Hernández, S. Cruz, and E. Millán. Fixed number of segments in unidirectional decompositions of fluence matrices for step-and-shoot IMRT. *Phys Med Biol*, 56(8):2601–2615, Apr 2011. DOI: 10.1088/0031-9155/56/8/017. URL `http://dx.doi.org/10.1088/0031-9155/56/8/017`.

# GRANTS

Part of the research carried out in this thesis was performed within the framework of the research projects:

- Técnicas de segmentación MLC para la administración eficiente de la dosis en IMRT. Ref: OTRI2008/0223. Principal researcher: Juan Manuel Artacho Terrer. From 01/03/2008 to 31/08/2010. Funded by Advanced Core, S.L.

- Diseño de algoritmos y estrategias para la administración efectiva de radioterapia de intensidad modulada (DESAER). Ref: PM021/2007. Principal researcher: Juan Manuel Artacho Terrer. From 01/10/2007 to 31/09/2009. Funded by the Aragón Goverment.

# BIBLIOGRAPHY

[1] A. Ahnesö, B. Hårdemark, U. Isacsson, and A. Montelius. The IMRT information process - mastering the degrees of freedom in external beam therapy. *Physics in Medicine and Biology*, 51(13):R381, 2006. URL http://stacks.iop.org/0031-9155/51/i=13/a=R22. (Cited on page 6.)

[2] M. Alber and F. Nüsslin. Intensity modulated photon beams subject to a minimal surface smoothing constraint. *Phys Med Biol*, 45 (5):N49–N52, May 2000. (Cited on page 72.)

[3] R. Arráns, M. I. Gallardo, J. Roselló, and F. Sánchez-Doblado. Computer optimization of class solutions designed on a beam segmentation basis. *Radiother Oncol*, 69(3):315–321, Dec 2003. (Cited on page 84.)

[4] J. Artacho, X. Mellado, G. Tobías, S. Cruz, and M. Hernández. A novel unidirectional intensity map segmentation method for step-and-shoot IMRT delivery with segment shape control. *Phys Med Biol*, 54(3):569–589, Jan 2009. DOI: 10.1088/0031-9155/54/3/007. URL http://dx.doi.org/10.1088/0031-9155/54/3/007. (Cited on pages 7, 52, 55, 61, 72, 73, and 78.)

[5] J. D. Azcona and J. Burguete. A system for intensity modulated dose plan verification based on an experimental pencil beam kernel obtained by deconvolution. *Med Phys*, 35(1):248–259, Jan 2008. (Cited on page 10.)

[6] J. D. Azcona and J. Burguete. Intensity modulated dose calculation with an improved experimental pencil-beam kernel. *Med Phys*, 37(9):4634–4642, Sep 2010. (Cited on page 10.)

[7] J. D. Azcona, R. A. C. Siochi, and I. Azinovic. Quality assurance in IMRT: importance of the transmission through the jaws for an accurate calculation of absolute doses and relative distributions. *Med Phys*, 29(3):269–274, Mar 2002. (Cited on page 28.)

[8] W. Bär, M. Alber, and F. Nüsslin. A variable fluence step clustering and segmentation algorithm for step and shoot IMRT. *Phys Med Biol*, 46(7):1997–2007, Jul 2001. (Cited on page 72.)

[9] T. R. Bortfeld, D. L. Kahler, T. J. Waldron, and A. L. Boyer. X-ray field compensation with multileaf collimators. *Int J Radiat Oncol Biol Phys*, 28(3):723–730, Feb 1994. (Cited on pages 7, 29, 37, 38, 45, and 52.)

[10] J. D. Bourland and E. L. Chaney. A finite-size pencil beam model for photon dose calculations in three dimensions. *Med Phys*, 19(6): 1401–1412, 1992. (Cited on page 10.)

[11] M. Broderick, M. Leech, and M. Coffey. Direct aperture optimization as a means of reducing the complexity of intensity modulated radiation therapy plans. *Radiat Oncol*, 4:8, 2009. DOI: 10.1186/1748-717X-4-8. URL `http://dx.doi.org/10.1186/1748-717X-4-8`. (Cited on pages 72 and 84.)

[12] G. J. Budgell, J. H. Mott, P. C. Williams, and K. J. Brown. Requirements for leaf position accuracy for dynamic multileaf collimation. *Phys Med Biol*, 45(5):1211–1227, May 2000. (Cited on page 28.)

[13] F. Carlsson. Combining segment generation with direct step-and-shoot optimization in intensity-modulated radiation therapy. *Med Phys*, 35(9):3828–3838, Sep 2008. (Cited on pages 72 and 84.)

[14] P. S. Cho and R. J. Marks. Hardware-sensitive optimization for intensity modulated radiotherapy. *Phys Med Biol*, 45(2):429–440, Feb 2000. (Cited on page 28.)

[15] C. S. Chui and R. Mohan. Extraction of pencil beam kernels by the deconvolution method. *Med Phys*, 15(2):138–144, 1988. (Cited on page 10.)

[16] H. Chung, H. Jin, J. Palta, T.-S. Suh, and S. Kim. Dose variations with varying calculation grid size in head and neck imrt. *Phys Med Biol*, 51(19):4841–4856, Oct 2006. DOI: 10.1088/0031-9155/51/19/008. URL `http://dx.doi.org/10.1088/0031-9155/51/19/008`. (Cited on page 5.)

[17] D. J. Convery and S. Webb. Generation of discrete beam-intensity modulation by dynamic multileaf collimation under minimum leaf separation constraints. *Phys Med Biol*, 43(9):2521–2538, Sep 1998. (Cited on page 28.)

[18] S. M. Crooks, L. F. McAven, D. F. Robinson, and L. Xing. Minimizing delivery time and monitor units in static IMRT by leaf-sequencing. *Phys Med Biol*, 47(17):3105–3116, Sep 2002. (Cited on page 7.)

[19] J. Dai and Y. Zhu. Minimizing the number of segments in a delivery sequence for intensity-modulated radiation therapy with a multileaf collimator. *Med Phys*, 28(10):2113–2120, Oct 2001. (Cited on pages 7 and 52.)

[20] E. M. Damen, M. J. Brugmans, A. van der Horst, L. Bos, J. V. Lebesque, B. J. Mijnheer, D. L. McShan, B. A. Fraass, and M. L. Kessler. Planning, computer optimization, and dosimetric verification of a segmented irradiation technique for prostate cancer. *Int J Radiat Oncol Biol Phys*, 49(4):1183–1195, Mar 2001. (Cited on page 84.)

[21] J. Deng, T. Pawlicki, Y. Chen, J. Li, S. B. Jiang, and C. M. Ma. The MLC tongue-and-groove effect on IMRT dose distributions. *Phys Med Biol*, 46(4):1039–1060, Apr 2001. (Cited on page 55.)

[22] M. L. Dirkx, B. J. Heijmen, and J. P. van Santvoort. Leaf trajectory calculation for dynamic multileaf collimation to realize optimized fluence profiles. *Phys Med Biol*, 43(5):1171–1184, May 1998. (Cited on page 28.)

[23] B. Dobler, F. Pohl, L. Bogner, and O. Koelbl. Comparison of direct machine parameter optimization versus fluence optimization with sequential sequencing in IMRT of hypopharyngeal carcinoma. *Radiat Oncol*, 2:33, 2007. DOI: 10.1186/1748-717X-2-33. URL `http://dx.doi.org/10.1186/1748-717X-2-33`. (Cited on page 84.)

[24] K. Engel. A new algorithm for optimal multileaf collimator field segmentation. *Discrete Appl Math*, 152(1-3):35–51, 2005. (Cited on pages 29, 34, 36, and 47.)

[25] J. M. Galvin, X. G. Chen, and R. M. Smith. Combining multileaf fields to modulate fluence distributions. *Int J Radiat Oncol Biol Phys*, 27(3):697–705, Oct 1993. (Cited on pages 4, 7, 37, 38, and 72.)

[26] M. Garey and D. Johnson. *Computers and intractability: a guide to the theory of NP-Completeness*. WH Freeman & Co. New York, NY, USA, 1979. (Cited on page 7.)

[27] H. A. Gay and A. Niemierko. A free program for calculating EUD-based NTCP and TCP in external beam radiotherapy. *Phis. Med.*, 23(3-4):115 – 125, 2007. ISSN 1120-1797. DOI: 10.1016/j.ejmp.2007.07.001. URL `http://dx.doi.org/10.1016/j.ejmp.2007.07.001`. (Cited on page 79.)

[28] I. M. R. T. C. W. Group. Intensity-modulated radiotherapy: current status and issues of interest. *Int J Radiat Oncol Biol Phys*, 51 (4):880–914, Nov 2001. (Cited on pages 5 and 6.)

[29] X. Gu, D. Choi, C. Men, H. Pan, A. Majumdar, and S. B. Jiang. GPU-based ultra-fast dose calculation using a finite size pencil beam model. *Phys Med Biol*, 54(20):6287–6297, Oct 2009. DOI: 10.1088/0031-9155/54/20/017. URL `http://dx.doi.org/10.1088/0031-9155/54/20/017`. (Cited on pages 5 and 10.)

[30] V. N. Hansen, P. M. Evans, G. J. Budgell, J. H. Mott, P. C. Williams, M. J. Brugmans, F. W. Wittkämper, B. J. Mijnheer, and K. Brown. Quality assurance of the dose delivered by small radiation segments. *Phys Med Biol*, 43(9):2665–2675, Sep 1998. (Cited on page 28.)

[31] B. Hårdemark, A. Liander, H. Rehbinder, and J. Löf. Direct machine parameter optimization with RayMachine in Pinnacle. RaySearch White Paper, RaySearch Laboratories, 2003. (Cited on pages 28 and 72.)

[32] A. Holder and B. Salter. A tutorial on radiation oncology and optimization. In H. J. G, editor, *Tutorials on Emerging Methodologies and Applications in Operations Research*, volume 76 of *International Series in Operations Research &amp; Management Science*,

pages 4–1–4–45. Springer New York, 2005. ISBN 978-0-387-22827-3. DOI: 10.1007/0-387-22827-6_4. URL http://dx.doi.org/10.1007/0-387-22827-6_4. (Cited on pages 2 and 6.)

[33] U. Jeleń and M. Alber. A finite size pencil beam algorithm for imrt dose optimization: density corrections. *Phys Med Biol*, 52(3): 617–633, Feb 2007. DOI: 10.1088/0031-9155/52/3/006. URL http://dx.doi.org/10.1088/0031-9155/52/3/006. (Cited on page 10.)

[34] U. Jeleń, M. Söhn, and M. Alber. A finite size pencil beam for imrt dose optimization. *Phys Med Biol*, 50(8):1747–1766, Apr 2005. DOI: 10.1088/0031-9155/50/8/009. URL http://dx.doi.org/10.1088/0031-9155/50/8/009. (Cited on page 10.)

[35] R. Jeraj, P. J. Keall, and J. V. Siebers. The effect of dose calculation accuracy on inverse treatment planning. *Phys Med Biol*, 47(3):391–407, Feb 2002. (Cited on page 5.)

[36] R. Jeraj, C. Wu, and T. R. Mackie. Optimizer convergence and local minima errors and their clinical importance. *Phys Med Biol*, 48(17):2809–2827, Sep 2003. (Cited on page 5.)

[37] Z. Jiang, M. A. Earl, G. W. Zhang, C. X. Yu, and D. M. Shepard. An examination of the number of required apertures for step-and-shoot IMRT. *Phys Med Biol*, 50(23):5653–5663, Dec 2005. DOI: 10.1088/0031-9155/50/23/017. URL http://dx.doi.org/10.1088/0031-9155/50/23/017. (Cited on page 84.)

[38] T. Kalinowski. *Optimal multileaf collimator field segmentation*. PhD thesis, Institut für Mathematik, Universität Rostock, 2004. (Cited on page 7.)

[39] T. Kalinowski. Multileaf collimator field segmentation without tongue-and-groove effect. Technical report, Institut für Mathematik, Universität Rostock, October 2004. (Cited on pages 29, 34, and 38.)

[40] T. Kalinowski. General theory of information transfer and combinatorics. In S. B. . Heidelberg, editor, *General Theory of Information Transfer and Combinatorics*, volume 4123 of *Lecture Notes in Computer Science*, chapter Realization of Intensity Modulated Radiation Fields Using Multileaf Collimators, pages 1010–1055. Springer, 2006. DOI: 10.1007/11889342_65. (Cited on pages 7, 28, 29, 34, 36, 37, 38, 47, 52, 55, and 61.)

[41] J. Löf and H. Rehbinder. Inverse planning optimization with RayOptimizer in Pinnacle. RaySearch White Paper, RaySearch Laboratories AB, Stockholm, Sweden, 2002. (Cited on page 72.)

[42] T. LoSasso, C. S. Chui, and C. C. Ling. Physical and dosimetric aspects of a multileaf collimation system used in the dynamic mode for implementing intensity modulated radiotherapy. *Med Phys*, 25(10):1919–1927, Oct 1998. (Cited on page 28.)

[43] W. Lu. A non-voxel-based broad-beam (nvbb) framework for imrt treatment planning. *Phys Med Biol*, 55(23):7175–7210, Dec 2010. DOI: 10.1088/0031-9155/55/23/002. URL http://dx.doi.org/10.1088/0031-9155/55/23/002. (Cited on page 92.)

[44] W. Lu and M. Chen. Fluence-convolution broad-beam (FCBB) dose calculation. *Phys Med Biol*, 55(23):7211–7229, Dec 2010. DOI: 10.1088/0031-9155/55/23/003. URL http://dx.doi.org/10.1088/0031-9155/55/23/003. (Cited on page 5.)

[45] M. M. Matuszak, E. W. Larsen, and B. A. Fraass. Reduction of IMRT beam complexity through the use of beam modulation penalties in the objective function. *Med Phys*, 34(2):507–520, Feb 2007. (Cited on page 72.)

[46] X. Mellado, S. Cruz, J. M. Artacho, and M. Canellas. Reducing the number of segments in unidirectional MLC segmentations. *Phys Med Biol*, 55(3):N75–N85, Jan 2010. DOI: 10.1088/0031-9155/55/3/N01. URL http://dx.doi.org/10.1088/0031-9155/55/3/N01. (Cited on page 72.)

[47] R. O. Memorial Sloan-Kettering Cancer Center. Departments of Medical Physics and Radiology. *A practical guide to intensity-modulated radiation therapy*. Medical Physics Publishing, 1st edition, 2003. (Cited on page 38.)

[48] O. Z. Ostapiak, Y. Zhu, and J. V. Dyk. Refinements of the finite-size pencil beam model of three-dimensional photon dose calculation. *Med Phys*, 24(5):743–750, May 1997. (Cited on page 10.)

[49] W. Que. Comparison of algorithms for multileaf collimator field segmentation. *Med Phys*, 26(11):2390–2396, Nov 1999. (Cited on page 38.)

[50] H. E. Romeijn, R. K. Ahuja, J. F. Dempsey, and A. Kumar. A column generation approach to radiation therapy treatment planning using aperture modulation. *SIAM J on Optimization*, 15(3):838–862, 2005. ISSN 1052-6234. DOI: http://dx.doi.org/10.1137/040606612. (Cited on page 72.)

[51] C. Scholz, S. Nill, and U. Oelfke. Comparison of IMRT optimization based on a pencil beam and a superposition algorithm. *Med Phys*, 30(7):1909–1913, Jul 2003. (Cited on page 5.)

[52] J. Seco, P. M. Evans, and S. Webb. Analysis of the effects of the delivery technique on an IMRT plan: comparison for multiple static field, dynamic and NOMOS MIMiC collimation. *Phys Med Biol*, 46(12):3073–3087, Dec 2001. (Cited on page 28.)

[53] M. B. Sharpe, B. M. Miller, D. Yan, and J. W. Wong. Monitor unit settings for intensity modulated beams delivered using a step-and-shoot approach. *Med Phys*, 27(12):2719–2725, Dec 2000. (Cited on page 72.)

[54] D. M. Shepard, M. C. Ferris, G. H. Olivera, and T. R. Mackie. Optimizing the delivery of radiation therapy to cancer patients. *SIAM Rev.*, 41:721–744, December 1999. ISSN 0036-1445. DOI: 10.1137/S0036144598342032. URL http://portal.acm.org/citation.cfm?id=340312.340328. (Cited on page 6.)

[55] D. M. Shepard, M. A. Earl, X. A. Li, S. Naqvi, and C. Yu. Direct aperture optimization: a turnkey solution for step-and-shoot IMRT. *Med Phys*, 29(6):1007–1018, Jun 2002. (Cited on pages 28 and 72.)

[56] R. L. Siddon. Calculation of the radiological depth. *Med Phys*, 12 (1):84–87, 1985. (Cited on page 10.)

[57] R. A. Siochi. Minimizing static intensity modulation delivery time using an intensity solid paradigm. *Int J Radiat Oncol Biol Phys*, 43 (3):671–680, Feb 1999. (Cited on pages 7, 29, 34, 37, 38, 52, 54, 55, 61, 72, 73, and 78.)

[58] S. V. Spirou and C. S. Chui. Generation of arbitrary intensity profiles by dynamic jaws or multileaf collimators. *Med Phys*, 21 (7):1031–1041, Jul 1994. (Cited on page 4.)

[59] S. V. Spirou, N. Fournier-Bidoz, J. Yang, C. S. Chui, and C. C. Ling. Smoothing intensity-modulated beam profiles to improve the efficiency of delivery. *Med Phys*, 28(10):2105–2112, Oct 2001. (Cited on pages 52 and 72.)

[60] J. P. van Santvoort and B. J. Heijmen. Dynamic multileaf collimation without 'tongue-and-groove' underdosage effects. *Phys Med Biol*, 41(10):2091–2105, Oct 1996. (Cited on page 34.)

[61] C. Wang, S. Luan, G. Tang, D. Z. Chen, M. A. Earl, and C. X. Yu. Arc-modulated radiation therapy (amrt): a single-arc form of intensity-modulated arc therapy. *Phys Med Biol*, 53(22):6291–6303, Nov 2008. DOI: 10.1088/0031-9155/53/22/002. URL http://dx.doi.org/10.1088/0031-9155/53/22/002. (Cited on page 91.)

[62] S. Webb. Use of a quantitative index of beam modulation to characterize dose conformality: illustration by a comparison of full beamlet IMRT, few-segment IMRT (fsIMRT) and conformal unmodulated radiotherapy. *Phys Med Biol*, 48(14):2051, 2003. URL http://stacks.iop.org/0031-9155/48/i=14/a=301. (Cited on page 79.)

[63] S. Webb, D. J. Convery, and P. M. Evans. Inverse planning with constraints to generate smoothed intensity-modulated beams. *Phys Med Biol*, 43(10):2785–2794, Oct 1998. (Cited on page 72.)

[64] Q. Wu and R. Mohan. Algorithms and functionality of an intensity modulated radiotherapy optimization system. *Med Phys*, 27 (4):701–711, Apr 2000. (Cited on page 6.)

[65] Q. Wu, D. Djajaputra, M. Lauterbach, Y. Wu, and R. Mohan. A fast dose calculation method based on table lookup for IMRT optimization. *Phys Med Biol*, 48(12):N159–N166, Jun 2003. (Cited on pages 5 and 10.)

[66] Y. Wu, D. Yan, M. B. Sharpe, B. Miller, and J. W. Wong. Implementing multiple static field delivery for intensity modulated beams. *Med Phys*, 28(11):2188–2197, Nov 2001. (Cited on page 72.)

[67] P. Xia and L. J. Verhey. Multileaf collimator leaf sequencing algorithm for intensity modulated beams with multiple static segments. *Med Phys*, 25(8):1424–1434, Aug 1998. (Cited on pages 7, 37, 38, and 45.)

[68] Y. Yang and L. Xing. Incorporating leaf transmission and head scatter corrections into step-and-shoot leaf sequences for IMRT. *Int J Radiat Oncol Biol Phys*, 55(4):1121–1134, Mar 2003. (Cited on page 28.)

[69] G. Zhang, Z. Jiang, D. Shepard, M. Earl, and C. Yu. Effect of beamlet step-size on IMRT plan quality. *Med Phys*, 32(11):3448–3454, Nov 2005. (Cited on pages 5 and 16.)