Gamma Knife treatment planning with new degrees of freedom

EMIL NORELL
Gamma Knife treatment planning with new degrees of freedom

EMIL NORELL
Acknowledgements

I would like to begin by expressing my gratitude to my supervisor Dr. Stella Riad from Elekta, for generously sharing so much of her valuable time and experience with me. Also, sincere thanks to Dr. Axel Ringh, my supervisor at KTH, for always having had the patience to listen to my lengthy expositions for longer than he probably had to. Furthermore, warm thanks to Dr. Håkan Nordström for his genuine interest in the project and all the time he has invested since the start. Finally, I would like to thank Aurora Samper for her invaluable support all along.

Emil Norell, Stockholm, February 2019
Abstract

The Leksell Gamma Knife® is an instrument designed for high precision treatment of tumours and lesions located in the brain and upper spine. Today, the radioactive cobalt-60 sources can only move linearly along the radiation unit, but the machine could be modified to include rotational motion as well. We extend an existing linear programming approach to inverse planning for the Gamma Knife by examining the benefits from rotational degrees of freedom.

The improvements offered from rotations are limited, but easy to make use of. We investigate the model in four patient cases, and find that an upper bound on the improvement of the optimization cost function is between 4.5% and 7.0% depending on case. With a total of four angles distributed uniformly over a 45 degree interval, one can in each case achieve a solution that performs up to within 1% of this bound. The average maximal improvements in terms of clinical metrics are 0.5% selectivity and 1.9% gradient index, at the cost of 5.9% worse beam-on time. No statistically significant change in coverage is found.

A dynamic model based on column generation is proposed, which allows treatment during constant velocity angular motion and can achieve practically the same plan quality as the model with uniformly distributed angles at a significantly lower problem size. A similar algorithm can be designed to locate the most effective angles in a non-uniform manner that achieves better plans with fewer added rotational degrees of freedom.

Trade-offs between memory and solution times are used to successively reduce the RAM occupation by around 90% and make significantly larger models computationally feasible to solve. A voxel clustering approach with emphasis on surface voxels, adapted to the radiosurgical framework, can significantly reduce the problem size while still producing competitive plans.

Keywords: Stereotactic radiosurgery, Leksell Gamma Knife®, inverse planning, clustering, optimization, column generation
Sammanfattning

Strålkniven Leksell Gamma Knife® är ett instrument designat för högprecisionsbesträning av tumörer och lesioner i hjärnan och övre delen av ryggraden. Idag kan de radioaktiva källorna endast förflyttas linjärt under behandlingen, men maskinen skulle kunna modifieras för att även tillåta rotationsrörelser. Vi utvidgar ett ramverk för inversplanering, formulerat som ett linjär-programmeringsproblem, genom att undersöka fördelarna med nya rotationsfrihetsgrader.

Förbättringarna som rotationer möjliggör är begränsade, men kan relativt enkelt tas till vara. Vi undersöker de potentiella förändringarna i fyra patientfall och finner att den övre gränsen av förbättringarna för målfunktionsvärdet i optimeringsproblemet är mellan mellan 4.5% och 7.0% beroende på fall. Genom att tillåta rotation av källorna till fyra jämnt fördelade vinklar över 45 grader kan man i samtliga fall hitta en lösning som är inom 1% från det bästa målfunktionsvärdet. De genomsnittliga förbättringarna i form av kliniska metriker är 0.5% selektivitet och 1.9% gradient-index, dock på bekostnad av en försämring av bestrålningstiden med 5.9%. Ingen tydlig förändring av täckningen kunde påvisas.

En modell baserad på kolumngenerering, som tillåter behandling under rotation av kollimatorkroppen med konstant hastighet, föreslås. I denna modell kan praktiskt taget lika bra lösningar uppnås som för likformigt fördelade vinklar, men med betydligt mindre problemstorlek. En liknande algoritm kan lokaliserka de mest effektiva vinklarna och åstadkomma samma plankvalitet med färre, men olikformt fördelade, rotationsfrihetsgrader.

RAM-användningen kan reduceras med cirka 90% genom avvägningar mellan minne och beräkningsstider, vilket möjliggör lösning av probleminstanser som tidigare var beräkningsmässigt omöjliga. Klusteringmetoder av voxlar anpassade till strålkniven kan minska problemstorleken betydelsefullt medan de resulterande behandlingsplanerna är fortsatt konkurrenskraftiga.

**Titel:** Behandlingsplanering med nya frihetsgrader för Strålkniven

**Nyckelord:** Stereotaktisk radiokirurgi, strålkniven, inversplanering, klustring, optimering, kolumngenerering
Acronyms

CT  Computed Tomography.
DOF  Degree of Freedom.
DRK  Dose Rate Kernel.
DRM  Dose Rate Matrix.
GPU  Graphics Processing Unit.
IMRT  Intensity Modulated Radiotherapy.
LGK  Leksell Gamma Knife.
LGP  Leksell Gamma Plan.
LP  Linear Program.
MC  Monte Carlo.
MIP  Mixed Integer Problem.
MRI  Magnetic Resonance Imaging.
OAR  Organ at Risk.
RAM  Random Access Memory.
RNG  Random Number Generator.
RSS  Representative Subsampling.
RU  Radiation Unit.
SRS  Stereotactic Radiosurgery.
5 Results and Discussion

5.1 Computing the dose rate matrix ................................................. 30
5.2 Clustering in the inverse planner .............................................. 32
  5.2.1 The K-means performance ................................................. 32
  5.2.2 Treatment planning with clustered models ............................. 34
5.3 Comparison between the models with new degrees of freedom .......... 36
  5.3.1 Optimization cost function and solve times ......................... 36
  5.3.2 Distribution of beam-on time ............................................ 39
  5.3.3 Improvement of the clinical metrics ................................. 41
5.4 Evaluating the rotation invariance assumption with Monte Carlo simulations ... 43
  5.4.1 Comparison of dose distribution ....................................... 44
  5.4.2 Dose regions and $L_2$-error .......................................... 45

6 Conclusions and further work .................................................. 47

Bibliography ................................................................................. 49

Appendices .................................................................................... 51

A Complementary mathematics .................................................. 52
  A.1 Standard form of the underlying model ................................ 52
  A.2 The dual formulation .......................................................... 53
    A.2.1 Implications of freezing objectives .................................. 54
    A.2.2 The clustered model ..................................................... 54
    A.2.3 The dynamic model ..................................................... 55

B Additional results ..................................................................... 56
  B.1 Distribution of beam-on time ............................................... 56
## List of Figures

2.1 An example of a Pareto optimal set for a problem with two objectives.............. 4  
2.2 An example of how the weighting method can produce one single solution............. 5  
2.3 Cobalt-60 decay.................................................. 7  
2.4 A number of photon-atom interactions resulting in the ejection of an electron.... 7  
2.5 A collimator can be used to shape beams to the desired form (as illustrated in this picture).................................................................................. 7  
2.6 Direct and indirect action of radiation on a DNA molecule.................................. 8  
2.7 An overview of the latest LGK Icon. Images taken from [8]................................. 9  
2.8 The synergy between the sectors and the collimator body allows radiation to be directed into the isocenter. Images taken from [8]................................. 9  
2.9 The sectors can be moved to allow the radioactive sources to deliver radiation through each of the collimator holes. Images taken from [8]................................. 10  
2.10 In the step-and-shoot procedure, shots are only delivered from predetermined isocenter positions. They are chosen to allow the entire geometry of, e.g., the tumour to be covered with dose................................................................. 10  
2.11 An example of an LGK shot. Sectors can deliver through collimators of 4, 8 and 16 mm. Alternatively, they can be kept blocked 'B'. Image taken from [8]........ 10  
2.12 The radiation, illustrated with yellow and green isodose lines, must be delivered so that the tumour (red) receives a sufficiently high dose. This, while the dose to the OAR (purple) must be kept low. Image taken from [22]................................. 11  
2.13 The four relevant structures for the modelling................................................. 12  
2.14 An example of how a tumour volume can be discretized into voxels.................. 12  
2.15 The fill algorithm use template shots to cover the geometry from the border and inwards. Image taken from [4]......................................................... 13  
2.16 The dose absorbed in tissue can be illustrated with iso-dose surfaces that enclose the volumes receiving a dose greater or equal to a reference dose................. 15  
3.1 The coverage (green), selectivity (blue) and optimization time (purple) as functions of the fraction of voxels in the target. The dashed lines indicate the result when all voxels were used. Image taken from [22]................................. 17  
3.2 The DRM can be reordered and divided into four submatrices, and one of these is never used.......................................................... 18  
3.3 Voxels can be grouped into clusters to reduce the problem size......................... 19  
3.4 Rotating a sector to -22.5 and 22.5 degrees yield the same outcome because of the symmetry.................................................. 21  
3.5 An example of the order in which angles can be chosen to form a nested sequence. 21  
5.1 Prioritizing memory can significantly reduce the RAM usage, but the final evaluation time is prolonged.................................................. 31  
5.2 Prioritizing memory can significantly reduce the RAM usage, but the final evaluation time is prolonged.................................................. 32  
5.3 A cross section of a tumour belonging to GFR46 where every cluster starting guess have randomly been given a color assigned to each of the member voxels............ 32  
5.4 Matrix $L_2$-norm improvement as a function of fraction and K-means iterations... 33
List of Tables

5.1 A brief overview of the cases considered in the testing. .......................... 30
5.2 Calculation time of the DRM in seconds for different number of angles. .... 31
5.3 Comparison of clinical metrics between different variations of clustering and RSS. 34
5.4 Comparison of clinical metrics between different variations of clustering and RSS. 34
5.5 The K-means clustering times in seconds for every considered case. .......... 35
5.6 The optimization times in seconds for the three different models. .............. 39
5.7 Clinical metrics as a function of number of uniformly distributed angles in the model. 41
5.8 Clinical metrics as a function of number of uniformly distributed angles in the model. 41
5.9 Clinical metrics as a function of angles in the dynamic model. ................. 42
5.10 Clinical metrics as a function of angles in the dynamic model. ................. 42
5.11 Clinical metrics as a function of iterations in the non-uniform model. ....... 43
5.12 Clinical metrics as a function of iterations in the non-uniform model. ....... 43
Chapter 1

Introduction

Every year about 1300 primary tumours in the brain and central nervous systems are diagnosed in Sweden, and in 2016, brain tumours caused 591 deaths. The relative 5-year-survival-rate is 50% for men and 69% for women [1]. These tumours are mainly treated with open surgery complemented with radiation treatments or cytostatics (chemotherapy). Invasive brain surgery is however both complicated and risky. Moreover, it demands the intervention of several highly qualified surgeons and experts, whose availability and cost limit the number of operations that can be carried out. Radiation treatments on the contrary are non-invasive and have a high degree of automation.

The Leksell Gamma Knife® (LGK) is designed and manufactured by Elekta AB for the purpose of treating cancer tumours and lesions, located in the brain and upper spine, using ionizing radiation. Lars Leksell, neurosurgeon at Karolinska University Hospital, and Börje Larsson, professor in radiobiology at Uppsala University, invented the concept of radiosurgery in 1951 [13]. This laid the foundation for the LGK and high energy radiation treatments in the brain. The first LGK was installed in Stockholm 1968 [7] and Lars Leksell later became one of the founders of Elekta in 1972. Today, 330 clinical LGK centres in 54 countries treat 80 000 patients every year [6]. Better software and more capable machines result in both faster and higher quality treatments which in the long run can save more lives. The LGK operates with high precision, however, its full potential has not yet been explored although it has been around for more than 50 years. Sometimes small hardware modifications can make a big difference for the patient.

The radiation unit (RU) of the LGK contains 192 radioactive cobalt-60 sources emitting ionizing radiation directed into one focal point called an isocenter. The radioactive sources are distributed over 8 movable components referred to as sectors that allow radiation to be delivered through collimators of different sizes. Currently, the LGK operates only with the RU in a stationary position and relies only on translatory movements for positioning of the patient during treatments. However, the machine still has more degrees of freedom (DOFs), but these are currently not used. The possibility to control the rotation of the RU in a stationary or dynamic manner during treatment would introduce new DOFs that may be helpful when treating tumours in sensitive places, where risks are high of damaging adjacent organs or healthy tissue. The effects of these new rotational DOFs will be investigated in this thesis.

A linear programming approach to inverse planning for the LGK has been proposed in [22], and it is able to generate plans capable of competing with clinically acceptable plans constructed by forward planning. It is hypothesized that a significant further improvement in overall plan quality can be attained by including these new rotational DOF in the optimization model. In order to study the previously mentioned hypothesis, the outline of this thesis is as follows: Background material giving an introduction to the LGK and radiation treatments, along with the underlying model, is presented in Chapter 2. Next, Chapter 3 contains a number of tools for LGK modelling with new DOFs. First, Section 3.1 presents a number of approaches capable of reducing the problem size where, among other topics, clustering of voxels in an LGK framework is investigated. Second, an approach to incorporate the rotational DOF into the optimization, based on a rotation invariance
assumption, is presented in Section 3.2. Third, the potential of column generation algorithms in radiation treatments is explored in Section 3.3. Section 3.4 concludes the chapter by briefly outlining a method to freeze objectives in the optimization. Further, a number of new optimization models based on the methods previously established are proposed in Chapter 4. Next, Chapter 5 presents the results and discussion of the comparisons between the earlier mentioned models along with an evaluation of the rotation invariance assumption. Finally, Chapter 6 summarizes the findings and suggests appropriate areas for further work.
Chapter 2

Background

The creation of treatment plans is carried out in an optimization framework which must be understood to be able to properly study the impact of new DOFs in radiation treatments. To understand the optimization model, one must first have a basic knowledge of optimization in addition to a good grasp of how the LGK operates. This in turn requires a brief overview of the physics of radiation and implications in a biological context.

2.1 Optimization

In mathematical optimization one is concerned with finding a solution \( x \in \mathbb{R}^n \) within a designated feasible region \( F \subseteq \mathbb{R}^n \) so that some cost function \( f : \mathbb{R}^n \to \mathbb{R} \) is minimized with respect to said strategy \( x \). Such a program is generally expressed on the form

\[
(P) \quad \text{minimize} \quad f(x) \\
\text{subject to} \quad x \in F.
\]  

(2.1)

A strategy \( x \) refers to the possible courses of action that can be taken in the context, which has a known connection to the value function \( f \) that is to be controlled. Throughout this thesis, we always assume that the feasible region \( F \subseteq \mathbb{R}^n \) is nonempty. In our context, this region is designed so that the corresponding feasible solutions \( x \) fulfill certain rules, e.g. physical laws, within the problem.

Consider a very modest example: you need to dispose of 50 kg waste at the minimal cost. Company 1 charges 10 kr per kilo of waste disposal while company 2 only charges 5 kr/kg. However, company 1 accepts a maximum of 40 kg whereas company 2 only accepts up to 30 kg. Then the strategy variables \( x_1, x_2 \) denote how much waste in kg is sent to each company and the resulting program becomes

\[
(P) \quad \text{minimize} \quad 10 \cdot x_1 + 5 \cdot x_2 \\
\text{subject to} \quad x_1 + x_2 = 50 \\
0 \leq x_1 \leq 40 \\
0 \leq x_2 \leq 30.
\]

Definition 2.1.1. Optimality

1. A strategy \( \hat{x} \in F \) such that \( f(\hat{x}) \leq f(x), \forall x \in F \) is called a global optimal solution.

2. A solution \( \tilde{x} \) such that \( f(\tilde{x}) \leq f(x), \forall x \text{ s.t. } \|\tilde{x} - x\|_2 \leq \epsilon \), for some \( \epsilon > 0 \) is called a local optimal solution.

Note that the minimization is just a convention; if one desires to instead maximize \( f(x) \) then it is equivalent to minimizing \(- f(x)\). Not every program has an optimal solution, and if they have one it is in general not easy to find. However, certain optimization problems, so called convex optimization problems, are in general easier to solve.
**Definition 2.1.2** ([18]). Convexity

1. A set \( F \subset \mathbb{R}^n \) is convex if \( x, y \in F \) implies that \( \alpha x + (1 - \alpha) y \in F \) for all \( 0 \leq \alpha \leq 1 \).

2. A function \( f : F \to \mathbb{R} \) is convex if \( f(\alpha x + (1 - \alpha) y) \leq \alpha f(x) + (1 - \alpha) f(y) \) for all \( 0 \leq \alpha \leq 1 \), for all \( x, y \in F \).

(P) is a convex program if \( f \) is a convex function and \( F \) is a convex set as stated in Definition 2.1.2. A local optimal solution in a convex program is guaranteed to also be a global optimum [9].

Moreover, consider a program \((P^*)\) on the form (2.1) with cost function \( f^* \) and feasible region \( F^* \). If \((P^*)\) is such that \( f^*(x) \leq f(x) \) for all \( x \in F \) and \( F \subseteq F^* \), then it is a relaxation of \((P)\). The optimal cost function value to \((P^*)\) serves as a lower bound to the optimal value of \((P)\).

### 2.1.1 Multi-objective optimization

There are situations when several, often conflicting, objectives have to be optimized all at once. A multi-objective program is commonly expressed on the form

\[
(MOP) \quad \text{minimize}_{x} \quad \{ f_1(x), f_2(x), ..., f_M(x) \}
\]

subject to \( x \in F, \tag{2.2} \)

where \( M \) denotes the number of objectives. An objective vector is denoted \( \hat{z} := (z_1, z_2, ..., z_M) \in \mathcal{Z} \), where the feasible objective region is given by \( \mathcal{Z} := \{ (f_1(x), f_2(x), ..., f_M(x)) \in \mathbb{R}^M : x \in F \} \). The optimal solution with respect to different objectives are rarely the same. One is therefore in general concerned with finding a so called Pareto Optimal solution. The concept, which is defined below, is also illustrated in Figure 2.1.

**Definition 2.1.3** ([18]). Pareto optimality

1. A decision vector \( \hat{x} \in F \) is Pareto optimal if there does not exist another decision vector \( x \in F \) such that \( f_i(x) \leq f_i(\hat{x}) \) for all \( i = 1, ..., M \) and \( f_j(x) < f_j(\hat{x}) \) for at least one index \( j \).

2. An objective vector \( \hat{z} \in \mathcal{Z} \) is Pareto optimal if there does not exist another objective vector \( z \in \mathcal{Z} \) such that \( z_i \leq \hat{z}_i \) for all \( i = 1, ..., M \) and \( \hat{z}_j < z_j \) for at least one index \( j \); or equivalently, \( \hat{z} \) is Pareto optimal if the decision vector corresponding to it is Pareto optimal.

![Figure 2.1: An example of a Pareto optimal set for a problem with two objectives.](image)

One way to find a single solution to a multi-objective optimization problem is by considering a weighted sum of the objective functions in question, i.e., the weighting method. In this case one only has to solve a standard optimization problem with a single objective function. This gives the problem

\[
(WSP) \quad \text{minimize}_{x} \quad \omega_1 \cdot f_1(x) + \omega_2 \cdot f_2(x) + ... + \omega_M \cdot f_M(x)
\]

subject to \( x \in F, \ F \subseteq \mathbb{R}^n, \tag{2.3} \)

where \( \omega_1, ..., \omega_M \) are weights that have to be chosen appropriately. Figure 2.2 describes how a Pareto optimal solution is obtained with the weighting method. The feasible objective region \( \mathcal{Z} \)
is a convex set if $f_i(x), i = 1, ..., M$ are linear programs (LPs). This leads to that a solution to (WSP) is always Pareto optimal if the weights $\omega_1, ..., \omega_M \geq 0$ [18].

![Figure 2.2: An example of how the weighting method can produce one single solution.](image)

### 2.1.2 Linear programming

LPs are a subclass of all possible optimization problems where the cost function and feasible region are expressed by linear functions. The cost function is $f(x) = c^\top x = c_1 \cdot x_1 + c_2 \cdot x_2 + ... + c_n \cdot x_n$ and the feasible region has the form $\mathcal{F} = \{x \in \mathbb{R}^n : Ax = b, \quad x \geq 0, \quad A \in \mathbb{R}^{m \times n}\}$. Thus, a LP is conventionally written on the standard form

\begin{align}
\text{(LP) } \quad & \underset{x}{\text{minimize}} & & c^\top x \\
& \text{subject to} & & Ax = b \\
& & & x \geq 0.
\end{align}

(2.4)

It will be referred to as the *primal* program. The concept of *duality* asserts the existence of a corresponding *dual* program that can be written on the form

\begin{align}
\text{(LD) } \quad & \underset{\lambda}{\text{maximize}} & & b^\top \lambda \\
& \text{subject to} & & A^\top \lambda \leq c \\
& & & \lambda \text{ free}.
\end{align}

(2.5)

An interesting characteristic of LPs is the presence of *strong duality*. It states that if these two programs both have feasible solutions $x$ and $\lambda$, then there exist finite optimal solutions $\hat{x}$ and $\hat{\lambda}$ such that $c^\top \hat{x} = b^\top \hat{\lambda}$ [20].

### 2.1.3 The Simplex algorithm

The *Simplex algorithm* is an iterative solver of LPs that moves between the vertices of $\mathcal{F}$ and is capable of reaching a solution that is both primal and dual feasible, i.e. optimal, in almost every case\(^1\). Let $A \in \mathbb{R}^{m \times n}$, $m < n$ and assume that $A$ has rank $m$. One can then reorder the columns of $A$ and $x^\top$ such that $A = [A_{\beta} \quad A_{\nu}], \quad x = [x_{\beta}^\top \quad x_{\nu}^\top]^\top$, where $\beta$ is the *basic* set of $m$ variables and $\nu$ is the *non-basic* set of the remaining $n - m$ variables such that $\beta \cup \nu = \{1, ..., n\}$. A feasible solution to (LP) expressed as

$$A_{\beta}x_{\beta} = b, \quad A_{\nu}x_{\nu} = 0, \quad x \geq 0,$$

where $A_{\beta}$ has rank $m$, is then called a *basic feasible solution*, and it represents a vertex on the polytope defined by $Ax = b, \quad x \geq 0$. The fundamental theorem of linear programming guarantees that if there exists an optimal solution, then there exists an optimal basic feasible solution [9]. In

---

\(^1\)The simplex algorithm can get stuck in an infinite cycle in a few specific cases [9, Section 5.5].
every iteration of the simplex algorithm, the dual feasibility or the reduced cost \( r_\nu \) is found from evaluating

\[
r_\nu = c_\nu - A_\nu^T \lambda, \quad A_\beta^T \lambda = c_\beta,
\]

indicates whether the solution is optimal or not. Any non-basic element \( q \in \nu \) such that \( r_q < 0 \) indicates that the corresponding non-basic variable can improve the objective function if allowed to enter the basic set of variables, and thus if \( r_\nu < 0 \) then the solution is not optimal. The program can be interpreted as

\[
\text{(BIG)} \quad \underset{x, \beta}{\text{minimize}} \quad c_\beta^T x_\beta \\
\text{subject to} \quad A_\beta x_\beta = b \\
1 \leq \beta \leq n, \quad \beta \in \mathbb{Z}^m \\
x \geq 0,
\]

(2.6)

where \( \mathbb{Z} \) is the set of real integers. Here, the task to find the optimal set \( \beta \) is also a variable.

### 2.1.4 Column generation

**Column generation** is an extension to simplex that hinges on the fact that the non-basic set not necessarily have to be explicitly included in the model. This is, if one can evaluate the dual feasibility, i.e. reduced cost, separately. The program in (2.6) can be divided into a master program and a subprogram. Consider

\[
\text{(MAS)} \quad \underset{x_\beta}{\text{minimize}} \quad c_\beta^T x_\beta \\
\text{subject to} \quad A_\beta x_\beta = b \\
x \geq 0,
\]

Where \( \beta \) is assumed to be known. This master program becomes significantly smaller and operates similarly to (2.4). The dual feasibility constraint can be delegated to the subprogram according to

\[
\text{(SUB)} \quad \min_q c_q - \lambda^T A_q \\
\text{s.t.} \quad A_\beta^T \lambda = c_\beta \\
q \in \nu.
\]

The minimal reduced cost \( r_q \) is found and if it is greater than or equal to zero the solution of (MAS) is optimal to (BIG). Otherwise \( q \) is a new candidate for the basic set \( \beta \). For a more in-depth treatment of the topic, see, e.g., [9].

### 2.2 Radiation theory

Röntgen’s discovery of x-rays 1895 paved the way for usage of radiation in medical applications, and the potential of treating diseases with x-rays was discovered soon after [15]. The *computed tomography* (CT) scan, developed by Sir Godfrey Hounsfield in the late 1960s, uses the fact that different types of tissue absorb different amounts of x-ray beams, and it has become an invaluable noninvasive tool for producing cross sectional images of interesting areas of patients [3]. In this section, a brief overview over the fundamental physics and radiation biology relevant for the context is presented.

#### 2.2.1 Radiation physics

Radioactivity refers to the disintegration occurring in unstable substances in conjunction with a discharge of energy and particles. There are three types of radioactive decay; alpha (He\(^{2+}\)), beta
(e\(^-\)) and gamma (γ) radiation. For example, the radioactive cobalt-60 isotope decays by β-decay to the stable isotope Nickel-60, followed up with the emission of two γ-rays according to

\[
^60\text{Co} \rightarrow ^{60}\text{Ni} + e^- + \bar{\nu}_e + 2\gamma,
\]

where \(\bar{\nu}_e\) denotes an electron antineutrino [2]. The process is illustrated in Figure 2.3.

Figure 2.3: Cobalt-60 decay.

Photons typically interact with atoms via one of the following: the Photoelectric effect, Compton scattering, or Pair production. When photons interact with matter in these ways it can lead to ejection of an electron with the corresponding amount of kinetic energy, as illustrated in Figure 2.4. This leads to ionization of the atom and ejection of a charged particle with speed, hence the name ionizing radiation [16]. The electrons themselves are in turn capable of ionizing adjacent matter.

Figure 2.4: A number of photon-atom interactions resulting in the ejection of an electron.

It is impossible to control in what direction the radiation exits the nucleus, but one can create controlled beams of radiation by covering the radioactive source with a material impenetrable by radiation and leaving openings that correspond to the shape of the beam sought. A collimator is essentially a long hole in an otherwise solid material which is used to shape the outgoing beams of ionizing radiation in this way, see Figure 2.5. The effect is that only rays entering almost exactly parallel to the axis line can make its way through and come out at the other end.

Figure 2.5: A collimator can be used to shape beams to the desired form (as illustrated in this picture).
2.2.2 Radiobiology

Cancer is an abnormal cell proliferation that is not balanced by normally occurring cell death. These cells continuously grow and disrupt the functions of adjacent organs in the affected organism. A group of cancer cells is referred to as a tumour, which is either malignant or benign, depending on whether it is actively spreading or not. Malignant tumours can infiltrate nearby organs and sometimes spread to other parts of the body, i.e. metastasize, which makes them significantly harder to treat. It is either hereditary or environmental defects in the DNA of a cell that causes it to malfunction. Usually, these anomalies are detected by the cell, and if sufficient damage is present, it can be ordered to auto-terminate to avoid spread of defective genes to daughter cells [12].

![Figure 2.6: Direct and indirect action of radiation on a DNA molecule.](image)

The processes taking place in the cells during ionizing radiation treatment can usually be divided into direct and indirect action, where the most damage is caused by the latter. If the absorbed energy is adequate to eject electrons directly inside the DNA-helix molecule, then bond breaks are formed; This is called direct action. These breaks can be in the form of single or double breaks, where a single break can commonly be repaired by the cell, whereas a double strand break often results in cell death [12]. On the other hand, the indirect action ionizes adjacent water molecules, creating free radicals, that subsequently become capable of damaging the DNA chain, as seen in Figure 2.6. If there is a lack of oxygen, the indirect effect diminishes.

The goal of radiation treatment is to destroy sick cells using ionizing radiation. However, healthy cells are also susceptible to the damage, especially if they are close the tumour. In particular, some organs are extra sensitive to radiation. These will be referred to as organs at risk (OARs). In a radiation treatment planning process it is important to pay attention to OARs, in order to avoid severe irreversible damage to the patient. A typical example is the optic nerve that, if only slightly damaged, can render the patient blind. Moreover, cells sometimes show an astounding ability to repair themselves and circumvent sustained damage. This is also true for tumour cells, and in some cases the tumour might keep growing after treatment, or a secondary tumour may appear.

An important concept in radiobiology is dose, which measures the energy deposited into tissue as a result of exposure to ionizing radiation. It is commonly measured in Gray, where 1 Gy = 1 J/Kg. Technological and algorithmic advancements have resulted in increased popularity of the computationally intensive statistical Monte Carlo (MC) method for dose calculations. These methods are considered the most reliable and are able to simulate photon and electron transport in great detail [11].

2.3 The Leksell Gamma Knife

Stereotactic radiosurgery (SRS) is a noninvasive method for treating brain disorders by means of high dose irradiation. The inventor Lars Leksell once defined it as ”A single high dose fraction of radiation, stereotactically directed to an intracranial region of interest” [13]. SRS stresses the importance of the coordinate system in which the modelling and planning are taking place. The
The purpose of the LGK is automated SRS treatment of tumours and lesions in the brain and upper spine. Due to its high energy nature, mistakes can be fatal, and precision becomes paramount. In the majority of the body there are muscles, or other functions, that can potentially compromise the accuracy. For example when a patient breathes, large regions of the body is temporarily displaced, and often in an unpredictable manner. The skull, on the other hand, can more reliably be fixed during the treatment, and basically no internal motion occurs in the brain. Therefore, it can be considered stationary.

### 2.3.1 Components of the Gamma Knife

Prior to the treatment, the patient receives a frame attached to the skull. On the newer model, a tailor made mask can be used instead. In this way, the head of the patient can be fixed to the couch seen in Figure 2.7a. The couch re-positions the patient with translatory movements in three dimensions inside the RU.

![Image](image1.png)

(a) The couch and exterior.  
(b) The RU found inside the LGK.

Figure 2.7: An overview of the latest LGK Icon. Images taken from [8].

At the heart of the LGK we find the RU, shown in Figure 2.7b, which itself consists of a number of components. The collimator body, for the newer versions Icon and Perfexion, is shown in Figure 2.8a. It has the form of a hollowed cut cone with hundreds of collimator holes all directed into one focal point called the isocenter, as seen in Figure 2.8b. In a modern LGK the collimator sizes are 4, 8 and 16 mm in diameter at the focus. The collimator body is massive and consists of mostly lead and tungsten to prevent radiation leakage, making it heavy.

![Image](image2.png)

(a) The collimator body is a massive hollowed piece with hundreds of collimators.  
(b) The focal point of all collimator holes is called the isocenter.

Figure 2.8: The synergy between the sectors and the collimator body allows radiation to be directed into the isocenter. Images taken from [8].

Above the collimator holes, on the exterior sides of the collimator body, 192 radioactive cobalt-60 sources are distributed over eight components referred to as sectors. Each sector can be adjusted in such a way that the photons go through different sets of collimators, as shown in Figure 2.9. Note that all of the sectors can deliver radiation simultaneously. Each sector can also be positioned in an "off" position, where no radiation is delivered through any collimator. Note that rays emanating from a single collimator are not potent enough to cause harm within reasonable time, but the
energy deposited into the isocenter is significant [8]. The high gradients allow the high precision irradiation stipulated in SRS.

(a) The radioactive sources can be moved linearly during treatment.  
(b) A high amount of energy can be concentrated into the isocenter.

Figure 2.9: The sectors can be moved to allow the radioactive sources to deliver radiation through each of the collimator holes. Images taken from [8].

2.3.2 Gamma Knife treatment planning

The patient can be moved so that the isocenter of the LGK is positioned in one of a number of predetermined isocenter positions. These positions are typically placed inside the lesion of the patient. Beam delivery is only allowed while the patient is in a stationary position. The collimators are then blocked to allow the reposition to a new isocenter position. This is called the step-and-shoot procedure and a simplified illustration is found in Figure 2.10.

Figure 2.10: In the step-and-shoot procedure, shots are only delivered from predetermined isocenter positions. They are chosen to allow the entire geometry of, e.g., the tumour to be covered with dose.

In every isocenter position, radiation can be delivered through each set of collimators, from every sector independently. The result is referred to as an LGK "shot", and there is a high amount of shot variations, i.e., DOFs. In some situations it might be beneficial to have a number of sectors completely blocked during treatment in an isocenter position if, e.g., radiation would need to travel through an OAR. An example of such a collimator configuration is found in Figure 2.11.

Figure 2.11: An example of an LGK shot. Sectors can deliver through collimators of 4, 8 and 16 mm. Alternatively, they can be kept blocked 'B'. Image taken from [8].
A LGK treatment is carried out as outlined in Algorithm 1. As can be seen, a treatment plan is found by determining a number of control inputs, i.e., the isocenter positions $i$ and beam-on times $t_{csi}$. Prior to treatment planning the important structures, both tumour and OARs, need to be segmented. Given this information, the control inputs for the LGK must be carefully designed so that the optimal treatment is administered. The goal is to find the dose distribution in the patient that can be delivered by the LGK and best fits the patient in question, as illustrated in Figure 2.12. The treatment plans are generated from a software called Leksell GammaPlan® (LGP) [4].

![Figure 2.12](image-url)  
Figure 2.12: The radiation, illustrated with yellow and green isodose lines, must be delivered so that the tumour (red) receives a sufficiently high dose. This, while the dose to the OAR (purple) must be kept low. Image taken from [22].

### 2.4 Mathematical model

A popular technique for radiation treatment planning is called inverse planning. In this section the underlying linear inverse planner model proposed in [22] is briefly explained. The idea is that it is fairly easy to propose an ideal plan in terms of how much dose $D_T$ one would preferably deliver to the tumour, the prescription dose, and a critical dose, $D_O$, that must not be exceeded in the OAR. However, this plan cannot be feasibly reproduced because of physical restrictions. Thus, the new goal is to construct a physically feasible plan, which is as similar as possible to the ideal one. Note from Algorithm 1 that, given the isocenter positions, a plan is fully defined by determining the times $t_{csi}$ [min] delivering beams in isocenter position $i$, sector $s$ and collimator $c$. These delivery durations are the unknowns that this section aims to determine. The model procedure can be divided into four separate parts.

1. Placement of isocenter positions
2. Dose rate matrix calculation
3. Sector duration penalty optimization
4. Evaluation of clinical metrics
Let $X \subset \mathbb{R}^3$ denote the volume representing the head and upper spine of the patient and let $x \in X$ represent a coordinate in this volume. For the modelling, it is assumed that the necessary information is contained in four volumes, all subsets of $X$, referred to as *structures*. These four structures are illustrated in Figure 2.13. The *target*, denoted $X_T$, is as previously mentioned typically a tumour, or occasionally, a lesion in the brain. The *OAR*, denoted $X_O$, is the volume of adjacent sensitive organs. A new geometric structure named the *inner ring*, denoted $X_S$, has the form of a shell just outside the border of the target. Similarly, the *outer ring* $X_L$ is a larger shell enclosing both the target and the inner ring, that also intersects the OAR.

![Figure 2.13: The four relevant structures for the modelling.](image)

To simplify calculations, the volume of the head is discretized into $J$ cuboids called *voxels*, each with coordinate $x_j \in X$, as demonstrated in Figure 2.14. The dose $d_j$ is assumed to be uniform inside of the entire voxel. Let $\mathcal{J} := \{1, \ldots, J\}$ be an index set of the voxels and let $|\mathcal{J}| = J$ denote the number of elements in the set. We also define an index set for each structure below.

**Definition 2.4.1. Voxel structures index sets**

- Target: $\mathcal{T} := \{j \in \mathcal{J} : x_j \in X_T\}$, $T := |\mathcal{T}|$
- Inner ring: $\mathcal{S} := \{j \in \mathcal{J} : x_j \in X_S\}$, $S := |\mathcal{S}|$
- Outer ring: $\mathcal{L} := \{j \in \mathcal{J} : x_j \in X_L\}$, $L := |\mathcal{L}|$
- OAR: $\mathcal{O} := \{j \in \mathcal{J} : x_j \in (X_O \cap X_L)\}$, $O := |\mathcal{O}|$

Note that the intersecting set of voxel indices between the outer ring and the OAR will contain the voxels of the OAR that are closest to the isocenter positions, and thus these will receive more dose than the rest of the OAR volume. Hence, we consider only those. Finally, let $\mathcal{F} := \mathcal{T} \cup \mathcal{S} \cup \mathcal{L} \cup \mathcal{O}$ denote the set of voxels that belong to any structure, resulting in $|\mathcal{F}| <<< |\mathcal{J}|$.

![Figure 2.14: An example of how a tumour volume can be discretized into voxels.](image)
2.4.1 Placement of isocenter positions

Let \( \xi_i \in \mathbb{R}^3 \) denote the coordinate of isocenter position number \( i \) that, due to the high dose rate in its center, is traditionally always placed inside of the target, i.e., \( \xi_i \in X_T \). If the number and positions of the isocenters can be determined in advance, it turns out that the resulting optimization problem becomes convex, as will be shown later. Therefore, it is thus assumed that the number of isocenters \( I \), along with their positions \( \{\xi_i\}_{i=1}^{I} \), are known in advance.

![Template shots](image)

Figure 2.15: The fill algorithm use template shots to cover the geometry from the border and inwards. Image taken from [4].

The first task is to place a number of isocenter positions inside the target so that it can reliably be covered with dose. Placement can be done manually by a clinician, i.e. clinical isocenter positions, or for example by means of packing algorithms as shown in Figure 2.15. These algorithms use a number of predefined template shots to fill the geometry from the surface and inwards with as large shots as possible [4].

2.4.2 Dose rate matrix calculation

To calculate the dose in the volume, it will be assumed that the dose rate from the radioactive sources is time-invariant during the treatment. The variation of the activity during the duration of a treatment is negligible due to the fact that cobalt-60 has a half-life of 5.27 years [10]. Now, let \( \phi_{cs}(x, \xi) \) denote the dose accumulation rate in the point \( x \), resulting from irradiating trough sector \( s \) and collimator \( c \) in an isocenter position \( \xi \). The total dose \( d(x) \) in a point resulting from a shot in an isocenter, with all sectors and collimators, can then be expressed as

\[
d(x) = \sum_{c=1}^{3} \sum_{s=1}^{8} \phi_{cs}(x, \xi) \cdot t_{cs}.
\]

The dose rates \( \phi_{cs} \) are calculated with the help of the TMR-algorithm described in [5]. In this model the head is assumed to be homogeneous and entirely made out of water. As previously mentioned in Section 2.2.1, MC methods are the ideal choice for dose calculation, but they are typically too computationally expensive in a clinical context. To furthermore reduce the number of expensive computations, translation invariance is assumed. This means that one can reuse the dose distribution calculated in one central position \( \xi^* \in X_T \) for every isocenter in the tumour, just by repositioning according to

\[
\phi_{cs}(x, \xi_i) \approx k_{csi} \cdot \phi_{cs}(x + (\xi^* - \xi_i), \xi^*),
\]

where \( k_{csi} \) is a rescaling constant for adjustments. This scaling constant can be determined by just calculating the center dose for the desired isocenter according to

\[
k_{csi} := \phi_{cs}(\xi, \xi_i)/\phi_{cs}(\xi^*, \xi^*).
\]

The resulting dose calculation from (2.7) for every voxel, where \( d_j := d(x_j) \), can then be written as a simple matrix vector multiplication given by

\[
13
\]
\[
    d_j = \sum_{c=1}^{3} \sum_{s=1}^{8} \phi_{cs}(x_j, \xi_i) \cdot t_{csi} \quad \rightarrow \quad d = \Phi t. \tag{2.10}
\]

The matrix element \( \Phi_{j,csi} = \phi_{cs}(x_j, \xi_i) \) contains the dose rate from the known configuration, with the voxels \( j \) corresponding to the rows and the DOFs to the columns. The total number of DOFs becomes \( F = 3 \times 8 \times I \). All the known information is then contained inside the matrix \( \Phi \in \mathbb{R}^{J \times F} \) that can be calculated in advance; this matrix will be referred to as the dose rate matrix (DRM).

A specific column of the DRM is denoted \( \Phi_{csi} \).

### 2.4.3 Sector duration penalty optimization

Let \( f(d, t) : \mathbb{R}^{J+F} \rightarrow \mathbb{R} \) be a penalty based objective function that evaluates a distribution of dose \( d \) in the \( J \) voxels along with the times \( t \) spent in the \( F \) different DOFs. The function \( f \) is designed so that the optimal plan is found by minimizing it. The dose \( d(t) : \mathbb{R}^F \rightarrow \mathbb{R}^J \) is in turn a function of all the beam delivery times in \( t_{csi} \) as seen in (2.10), and these must satisfy \( t_{csi} \geq 0 \). Let \( \mathbb{D} \) be the feasible region for the dose that demands that the dose is acceptable inside voxels belonging to the OAR, i.e., \( \mathbb{D} := \{d \in \mathbb{R}^J : d_j \leq D_O, \quad \forall j \in \mathcal{O}\} \). The foundation of the sector duration optimization formulation then becomes

\[
\min_t \quad f(d, t) \\
\text{s.t.} \quad d = \Phi t \quad d \in \mathbb{D}, \quad t \geq 0. \tag{2.11}
\]

The goal is to give a high dose to the tumour voxels \( T \) and as little dose as possible to the rest of the voxels \( S, L, O \). Dose to target voxels, less than the prescription dose \( D_T \), are penalized with a function \( f_T \). Similarly, we design \( f_S, f_L \) to penalize doses in the inner and outer ring that are higher than \( D_S \) and \( D_L \), which are assumed to be known constants. We define the corresponding penalty functions as

\[
    f_T(d) := \frac{1}{D_T} \sum_{j \in T} \max\{D_T - d_j, 0\}, \\
    f_S(d) := \frac{1}{D_S} \sum_{j \in S} \max\{d_j - D_S, 0\}, \\
    f_L(d) := \frac{1}{D_L} \sum_{j \in L} \max\{d_j - D_L, 0\}. \tag{2.12}
\]

These piece-wise linear functions can be expressed in a linear programming framework, the details, along with the expression in standard form, are given in Appendix A.1. Note that every objective is normalized with the number of voxels in the structure and the magnitude of the dose. In addition, the treatment time is penalized. In every isocenter, it is the sector that has the longest total time that dictates how much time is spent in that position, thus the beam-on time (BOT) and its penalty function are defined as

\[
    \tau(t) := \sum_{i=1}^{I} \max_s \sum_{c=1}^{3} t_{csi} \quad \text{and} \quad f_B(t) := \phi_{cal} \cdot \frac{\tau(t)}{D_T}.
\]

The BOT rescaling constant \( \phi_{cal} \), with unit Gy/min, is the calibration dose rate measured at installation and then adjusted for as the cobalt-60 decays. The result is a multi-objective problem on the form (2.2) that can be handled with the weighting method, as described in (2.3), where the objective function can be expressed as

\[
    f(d, t) := \omega_T \cdot f_T(d) + \omega_S \cdot f_S(d) + \omega_L \cdot f_L(d) + \omega_B \cdot f_B(t),
\]

14
where $\omega_T, \omega_S, \omega_L, \omega_B$ are the weights that are assumed to be known. The resulting LP, found from rewriting (2.11), becomes

$$
(P) \quad \min_t \quad f(d, t)
$$

$$
s.t. \quad d_j = \sum_{c_{si}} \phi_{c_s}(x_{j}, \xi_{i}) \cdot t_{c_{si}}, \quad \forall j \in \mathcal{J}
$$

$$
d \in \mathcal{D}, \quad t \geq 0.
$$

Note that there is a lot to gain from solving this problem in a the dual instead, which can be found in Appendix A.2. For further details, see the original paper [22].

### 2.4.4 Evaluation of clinical metrics

When a plan has been generated from the optimization, it is of interest to evaluate the plan before executing it. The dose to all voxels must be computed, not only those contained in the structures $\mathcal{J}$. The evaluation in a clinical context is done in terms of metrics that are related to the objectives in the optimization, but they are not directly proportional to them. The reason for not using these metrics directly in the optimization problem is that they are non-convex.

(a) The desired isodose volume $X_T$ and the result of the plan $X_P$.

(b) The relation between the volume receiving half the prescription dose, $X_{P/2}$, and the full, $X_P$.

Figure 2.16: The dose absorbed in tissue can be illustrated with iso-dose surfaces that enclose the volumes receiving a dose greater or equal to a reference dose.

In this section we will give the definitions of the clinical metrics that will be used for the evaluation in this thesis. Let $X_P$ denote the planning isodose volume, the volume that is receiving a dose greater than or equal to $D_T$ as a result of the treatment plan, i.e., $X_P = \{x \in X : d(x) \geq D_T\}$. Consider its relation to the target volume $X_T$ in Figure 2.16a. In the ideal case one would of course like that $X_P = X_T$, meaning that no other tissue except the tumour receives a large dose. However this is, as earlier mentioned, practically impossible. Therefore, let $V(\cdot) : \mathbb{R}^3 \rightarrow \mathbb{R}$ denote the volume of a set and define

$$
\text{Coverage} := \frac{V(X_T \cap X_P)}{V(X_T)}
$$

that measures how large share of the target that receives the correct dose. Furthermore, it is also important to make sure that not too much dose is leaking out to healthy tissue, why we define

$$
\text{Selectivity} := \frac{V(X_T \cap X_P)}{V(X_P)}
$$

that describes the share of the volume receiving prescription dose that is actually the target. A good plan requires a balance between these metrics, which can be emphasized by considering the Paddick conformity index given by $PI := \text{Coverage} \times \text{Selectivity}$ [19]. The goal is that these three measures are as close to 1 as possible. Moreover, the dose should drop quickly outside the target, which would indicate that adjacent tissue is being spared. Let $X_{P/2}$ denote the volume receiving at least half of the prescription dose and define

$$
\text{Gradient index} := \frac{V(X_{P/2})}{V(X_P)}
$$
that measures how steeply the dose drops to half the prescription dose. This is illustrated in Figure 2.16b. The Gradient index (GI) is limited from below by 1, which is the best result. Finally, we also typically regard the BOT given in (2.13) as a clinical metric as well.

Note the discrepancy between the objectives in the optimization, from the previous section, and these clinical metrics: the relations are not one-to-one. There is a strong relation between, for example, the GI and the outer ring penalty $f_L$, but an improvement in one might in fact result in a deterioration of the other. Even though a clinician is only interested in the clinical metrics and dose distributions, one can argue that there are still intrinsic value in the objectives as they are defined. For example, the objective value corresponding to the inner ring, $f_S$, is proportional to the magnitude of energy that is delivered past the ideal level in that volume, due to its linearity.
Chapter 3

Method

An approach to incorporate the rotational DOF into the model from Section 2.4 is detailed along with a number of techniques to handle the inevitable problem size increase. Furthermore, column generation methods adapted to the LGK, with the aim to find the most meaningful DOF, are also treated.

3.1 Problem size reduction

Despite the partitioning of the voxels into the structures mentioned in Section 2.4.3, there are still a large amount of voxels in the problem. If the number of isocenters is high, the DRM can become very large. The incorporation of new DOF in the model will inevitably increase the model size even further, at some point rendering the problem computationally infeasible due to the lack of RAM. The problem size also dictates the computation times of all calculations, including the time it takes for the simplex algorithm to solve the problem instances, which is vital for the model’s tractability in practice.

3.1.1 Representative subsampling

Representative subsampling (RSS) is a method that only allows a fraction of the voxels in the structures into the optimization. It has been shown that far from all the voxels are necessary for the optimization to create good plans for the LGK [22]. Figure 3.1 makes it is obvious that the plan quality is rather unaffected even though a fraction of the voxels are used.

![Figure 3.1: The coverage (green), selectivity (blue) and optimization time (purple) as functions of the fraction of voxels in the target. The dashed lines indicate the result when all voxels were used. Image taken from [22].](image-url)
Data for optimization can therefore be sampled randomly with a uniform distribution from the voxel structures detailed in Section 2.4.3. This notably reduces the problem size, however it also introduces some uncertainties. Let the superscript * denote a sampled structure which is reduced, e.g., $T^* \subset T$. The result is that the sampled number of voxels $|J^*| << |J|$ become significantly fewer, meaning that the DRM becomes significantly smaller. If we only sample from the structures given in Definition 2.4, the set of voxels used in the model is limited to

$$\tilde{J}^* := T^* \cup S^* \cup L^* \cup O,$$

where it should be noted that the OAR structure is not reduced. If one had sampled from the OAR voxels, it would not be possible to guarantee to the same extent that they all receive a low enough dose.

### 3.1.2 Prioritization of memory

As shown in the previous section, only a reduced set of voxels $\tilde{J}^*$ are necessary for the optimization. Moreover, it can also be noted that a significant amount of the variables $t_{csi}$ are zero in the optimal treatment plan. The columns of the DRM corresponding to these DOF are then multiplied with zero, making a significant part of them unnecessary, as illustrated in Figure 3.2. The result is that a considerable subset of the DRM contains elements that are never used in the final evaluation, but are very expensive to compute. When the problem size grows additionally by adding DOFs, this subset is expected to grow accordingly. The full DRM can be reordered and divided into four parts

$$\Phi = \begin{bmatrix} \Phi_{1,1} & \Phi_{1,2} \\ \Phi_{2,1} & \Phi_{2,2} \end{bmatrix},$$

where $\Phi_{1,1}$ and $\Phi_{1,2}$ contain the rows for the subsampled voxels necessary for the optimization, i.e., the $j_{csi}$ such that $j \in J^*$. Moreover, the elements corresponding to the nonzero DOFs, denoted $t_+$, are contained inside the columns of $\Phi_{1,1}$ and $\Phi_{2,1}$. Thus, $\Phi_{2,2}$ can be neglected entirely, leading to significant savings in computation times and memory usage. The evident problem is of course that it is impossible to predict in advance for which DOFs $t_{csi} = 0$.

Initially, it is only necessary to compute the dose data for the sampled voxels, which are involved in the optimization. The remaining can be determined afterwards, when it is known what variables are nonzero. In this way one can get away with the exact same answer with fewer calculations, less RAM usage, and presumably a lower total computation time. However, from a clinical perspective it might not always be optimal to postpone these dose calculations until after the optimization. In practice, the DRM can be calculated as soon as the segmentation of the patient is done, and the isocenter positions have been chosen. This while the clinician still decides the model parameters, e.g., the prescription dose. When the optimization starts, these parameters are theoretically already decided and additional wait after that might more directly be delaying the actual treatment. However, the difference is expected to be minimal in practice.

A more detailed description of the memory prioritization procedure
1. Generate $[\Phi_{1,1} \Phi_{1,2}]$

2. Solve (P) in (2.14), and find $t = \begin{bmatrix} t_+ \\ 0 \end{bmatrix}$

3. Generate $\Phi_{2,1}$

4. Evaluate $d = \begin{bmatrix} \Phi_{1,1} \\ \Phi_{2,1} \end{bmatrix} t_+$

### 3.1.3 Clustering rows in the dose rate matrix

An approach based on clustering of voxels in an IMRT framework has been explored in [23] with promising results. In this section we replicate the method described in the article, but adapted to an LGK setting. The goal is to investigate the implications of clustering in the inverse planner and how competitive it is to RSS. The overarching idea is similar to that of RSS: there is a lot of redundant information in the data and far from all of it is necessary to obtain a good plan. The clustering algorithm differentiates itself by grouping, i.e. clustering, data in such a way that a minimal amount of important information is lost. An example of clustering is demonstrated in Figure 3.3. It is emphasized that clusters are represented by the data from all its members.

In other words, the dose in a cluster can, e.g., be chosen as the average of the member voxels. However, clustering comes at an extra cost of time and computation load. This, in contrast to RSS where the data is sampled randomly and the remaining information is discarded, which is a much quicker process.

![Figure 3.3: Voxels can be grouped into clusters to reduce the problem size.](image)

The clustering is done as follows: the voxels can be distributed over a predetermined set of clusters $K := \{1, ..., K\}$. The relation between clusters and voxels is described by a matrix $U \in \{0, 1\}^J \times K$, where $U_{jk} \in \{0, 1\}$ indicates whether voxel $j$ belongs to cluster $k$ or not. Define $C_k := \{j \in J : U_{jk} = 1\}$ to be the set of indices of the voxels that belong to cluster $k$. It is assumed that the dose in the cluster $d_k$ is uniform and it can generally be represented by any convex function of the member voxel doses, but most commonly it is defined as the average according to

$$d_k = \sum_{c} \Phi_{c,csi} \cdot t_{csi} = \frac{1}{|C_k|} \sum_{j \in C_k} \sum_{c} \Phi_{j,csi} \cdot t_{csi}. \quad (3.1)$$

The goal is to produce a reduced matrix $\hat{\Phi}$ such that $U \hat{\Phi} \approx \Phi$. Note that, due to the sparse nature of $U$, the average dose from (3.1) can also be expressed $\hat{\Phi} = (U^\top U)^{-1}U^\top \Phi$. One can formulate the clustering problem as an optimization problem in the variable $U$, namely to minimize the $L_2$ matrix norm of the mismatch. This problem has the form
\[ \min_U \| \Phi - U\hat{\Phi} \|_2 \]
\[
\text{s.t.} \quad \hat{\Phi} = (U^TU)^{-1}U^T\Phi \\
\sum_{k \in K} U_{jk} = 1, \quad \forall j \in J \\
U_{jk} \in \{0, 1\}, \quad \forall j \in J, \quad \forall k \in K.
\] (3.2)

This has to be done separately for the voxels in every structure \( T, S, L \) to ensure clusters are not made from voxels in different volumes. Note that the OAR structure \( O \) is not clustered. The problem in (3.2) is NP-hard and one typically has to resort to some heuristic to determine \( U \). In this case we use the K-means algorithm [23], described in Algorithm 4. Note that a starting guess of clusters is necessary to start the algorithm. A simple starting guess can be constructed by simply grouping neighboring voxels in the DRM with the assumption that they are then close in space.

**Algorithm 2: Vectorized K-means [23]**

1. Calculate centroids \( \hat{\Phi} = (U^TU)^{-1}U^T\Phi \).
2. Compute uninitialized distance matrix
\[
D = -2\hat{\Phi}\Phi^T + 1\text{diag}(\hat{\Phi}^T\hat{\Phi})^T.
\]
3. \( U_{jk} = \begin{cases} 1, & k = \arg\min_k \{D_{jk}\} \\ 0, & \text{otherwise} \end{cases} \), \( j = 1, \ldots, J \), \( k = 1, \ldots, K \).
4. Repeat from 1 until stable or an iteration limit is reached.

### 3.2 Incorporating the rotational degree of freedom

The rotational DOF can rather seamlessly be incorporated into the model granted that one can use a rotation invariance assumption, similar to the already existing translation invariance assumption. Furthermore, it has to be decided what distribution and magnitudes of the angles to allow in the model.

#### 3.2.1 Rotation invariance assumption

Let \( \phi_{cs}(x, \xi, \theta) \) denote the dose rate for a given rotation \( \theta \) around the z-axis, the central axis of the collimator body, and consider a slight modification of the dose calculation in (2.7) given by
\[
d(x) = \sum_{c=1}^{3} \sum_{s=1}^{8} \phi_{cs}(x, \xi, \theta) \cdot t_{cs}.
\]

The rotations will be modeled as discrete nodes where delivery must be carried out from stationary positions, as an extra step in the step-and-shoot procedure. In other words, in every isocenter, the collimator body can be rotated to \( A \) different positions, each with an angle \( \theta_a \), for \( a = 1, \ldots, A \). The dose in every voxel can then be calculated as
\[
d_j = \sum_{c=1}^{3} \sum_{s=1}^{8} \sum_{a=1}^{A} \sum_{i=1}^{I} \phi(x_j, \xi_i, \theta_a) \cdot t_{csai} \quad \rightarrow \quad d = \Phi t,
\]
where \( \Phi_{j,csai} = \phi_{cs}(x_j, \xi_i, \theta_a) \) are the elements in a bigger dose rate matrix where the angle is also a setting in the configuration. Note that the total number of DOFs becomes \( F = 8 \times 3 \times A \times I \).

Now, the modelling assumption is that the dose distribution \( \phi \) is rotation invariant. In other words, when rotating the sources, the distribution form is preserved but rotated accordingly. In reality, rotating the machine might result in many beams passing trough a different amount of tissue which would then alter this form, but this effect is assumed to be negligible. The reasoning is similar to the one behind the translation invariance assumption from Section 2.4.2, which is now assumed to hold in combination with the rotation invariance. A modification to (2.8) results in
\[
\phi_{cs}(x, \xi_i, \theta_a) \approx k_{csai} \cdot \phi_{cs}(\xi^* + R(-\theta_a) \cdot (x - \xi_i), \xi^*),
\] (3.3)
where $R(\theta) \in \mathbb{R}^{3 \times 3}$ is the rotation matrix for a vector $x \in \mathbb{R}^3$ around the z-axis. The new compensatory factor, based on (2.9), can ideally be found from

$$k_{csai} = \frac{\phi_{cs}(\xi_i, \xi, \theta_a)}{\phi_{cs}(\xi^*_a, \xi^*, 0)},$$

where the center doses are assumed to be known. However, within the context of this thesis, it is assumed that $k_{csai} = k_{csi}$ for all $a$ according to (2.9).

### 3.2.2 Selection of angles

The next question is how far the collimator body should be able to rotate. Since there are 8 identical sectors there is no reason to rotate the collimator body more than or equal to a total of 45 degrees. This is illustrated in Figure 3.4. Moreover, as mentioned previously, rotation invariance is an assumption and is likely to become less accurate for bigger rotations. Therefore, by minimizing the maximal rotation, the effect of the assumption is assumed to be at its minimum. For that reason, only rotations between $-22.5$ and $22.5$ degrees are allowed.

![Figure 3.4](image)

Figure 3.4: Rotating a sector to -22.5 and 22.5 degrees yield the same outcome because of the symmetry.

It will be assumed that the total number of angles per isocenter $A$ is predetermined. Let the distribution of these angles

$$\Theta^A := \{\theta_a \in [-22.5, 22.5] : a = 1, ..., A\} \quad (3.4)$$

be designed in such a way that the set of angles forms a nested sequence, i.e., $\Theta^A \subset \Theta^{A+1}$. In this way, in every new bigger problem, the old strategies will always be available. In other words, the old feasible region of an optimization problem will always be contained inside the new, $F^A \subset F^{A+1}$, as will be shown further along. Note that this is not always achieved by dividing the interval uniformly over the number of angles.

![Figure 3.5](image)

Figure 3.5: An example of the order in which angles can be chosen to form a nested sequence.

An algorithm that produces a nested sequence of angles is proposed with the goal to place angles the largest possible distance from already existing ones, as shown in Figure 3.5. The result is thus a distribution that is uniform as long as $A = 2^n$, where $n$ is a positive integer. If nothing else is specified, this is the distribution of angles used in the rest of the thesis, and it is found from (3.4) with elements given by
\[ \theta_a = \begin{cases} 0, & a = 1 \\ -22.5, & a = 2 \\ +22.5 \cdot (1 - \frac{1}{2^n}) - (q - 1) \cdot \frac{22.5}{2^n}, & a = 2^n + (2q - 1) \\ -22.5 \cdot (1 - \frac{1}{2^n}) + (q - 1) \cdot \frac{22.5}{2^n}, & a = 2^n + 2q \end{cases} \] (3.5)

where \( q = 1, \ldots, 2^{n-1} \), \( n = \{ n \in \mathbb{Z}^+: 2^{n+1} \leq A \} \), and \( \mathbb{Z}^+ \) is the set of positive integers.

### 3.3 Column generation in the inverse planner

There is an infinite number of potential DOFs that could be incorporated into the model, but the improvement they offer can vary greatly. Column generation can be employed to locate the most significant DOFs before they are merged into the model. Consider an ideal formulation, of the form (2.6), according to

\[
\begin{align*}
(\text{BIG}) \quad \min_{t, \beta} & \quad f(d, t) \\
\text{s.t.} & \quad d = \Phi_{\beta} t_{\beta} \\
& \quad d \in D, \quad t \geq 0 \\
& \quad 1 \leq \beta \leq F, \quad \beta \in \mathbb{Z}^N.
\end{align*}
\] (3.6)

Let \( \Phi \in \mathbb{R}^{J \times F} \) be a matrix containing all imaginable DOFs and \( \beta \) is an index set referring to the columns of \( \Phi \) that is currently being used. The set \( \nu \) refers to all the potential, but unused, DOFs that could possibly be introduced into the model. Note that in the (BIG) model, the task of determining the best subset of DOFs to employ, \( \beta \), is also a variable in the optimization. The total number of imaginable DOFs, \( F \), is an enormous value and motivates the restriction on \( \beta \) to allow a maximum of \( N \) of the best DOFs to be employed. Typically, such problems are non-convex and hard to solve. Recall Section 2.1.4, and note that feasible solutions to (BIG) can be found, that are not necessarily optimal, by partitioning the problem into two separate programs. Consider the master program

\[
\begin{align*}
(\text{MAS}) \quad \min_t & \quad f(d, t) \\
\text{s.t.} & \quad d = \Phi_{\beta} t_{\beta} \\
& \quad d \in D, \quad t \geq 0,
\end{align*}
\]

where \( \beta \) is assumed to be given, which results in that the problem once again becomes linear. The idea is that after solving a problem (P), of the form (2.14), the set of Lagrange multipliers \( \lambda \) can be used evaluate to the dual infeasibility of the potential columns \( \nu \). The reduced costs \( r_\nu = c_\nu - \lambda^\top \Phi_\nu \) can be seen as a measure of the potential gain from including a specific column in the model. In this case the cost to introduce the column in the model is simply the BOT penalty given by

\[
c_q = \frac{\omega_B \cdot \Phi_{cal}}{D_{F}} \cdot F, \quad \forall q \in \nu. \tag{3.7}
\]

The task of the subprogram is then to produce the new DOF, \( \Phi_q \), that best serve the master program by minimizing the reduced cost

\[
\begin{align*}
(\text{SUB}) \quad \min_q & \quad c_q - \lambda^\top \Phi_q \\
\text{s.t.} & \quad q \in \nu.
\end{align*}
\]

The information gained from the subprogram can be used to update \( \beta \) for the next iteration. This is a very general formulation and in this thesis the emphasis will be put on the rotational DOFs. Hence, an inconvenience is that for every angle in an isocenter, there are actually a total of 24 columns, corresponding to every collimator and sector. From a practical standpoint it is wasteful not to make use of all sectors and collimators while treating in a position, since time can potentially be saved by allowing the others to deliver as well. Preferably all the 24 columns should be taken into account as a single unit. The following sections proposes different approaches to evaluate new DOFs from the subprograms.
3.3.1 Primitive Gamma Knife shots

The following approach relies on a simplification of the problem. Define a primitive shot to be a LGK shot where only one collimator per sector can be used, and where they all have to deliver for the same duration. Since the shot is rigid, the total dose rate becomes the sum of its member columns given by

\[ \tilde{\Phi}_{j,ai} := \sum_{cs} O_{cs} \cdot \phi_{cs}(x_j, \xi_i, \theta_a), \quad \sum_{c} O_{cs} \leq 1, \quad O_{cs} \in \{0, 1\}. \]  

(3.8)

Here, \( O_{cs} \) is a binary variable enforcing the rule that only one collimator per sector can be used. Note that a sector \( s \) can also be turned off, i.e., \( \sum_{c} O_{cs} = 0 \). Note that the DOFs are now compressed down to only one column per angle and isocenter position, instead of 24 as before. The reduced cost for this position becomes

\[ \tilde{r}_{ai} := c_{ai} - \lambda^T \tilde{\Phi}_{ai}, \]

where \( c_{ai} \) is equal to that of (3.7). To produce the primitive shot with the minimal reduced cost, the subprogram becomes an integer optimization problem where \( O_{cs} \) is sought according to

\[
\begin{align*}
\text{(SUB)}_{ai} \quad & \min c_{ai} - \sum_{j} \lambda_j \sum_{cs} O_{cs} \cdot \phi_{cs}(x_j, \xi_i, \theta_a) \\
\text{s.t.} \quad & \sum_{c} O_{cs} \leq 1, \quad \forall s \\
& O_{cs} \in \{0, 1\}, \quad \forall s, c
\end{align*}
\]

(3.9)

The (SUB) problems reduce to a few inexpensive dot product computations per angle \( a \) and isocenter \( i \) according to

\[
\begin{align*}
\text{(SUB)}_{ai} \quad & c_{ai} - \sum_{s} \max_{c} \{ \sum_{j} \lambda_j \cdot \phi_{cs}(x_j, \theta_a, \xi_i) \}
\end{align*}
\]

(3.10)

3.3.2 Candidate based approach

The following approach proposes a measure that can evaluate the potential benefit of the entire group of columns, corresponding to an angle and an isocenter position, without reducing them to primitive shots. Consider the solution in the previous section. There, only the best collimator in every sector is used, whereas in a regular shot one could of course use all of them if beneficial. A heuristic approach is to evaluate them all as one unit, where only the columns that can minimize the reduced cost are used. The latter is due to that any unfavorable DOF can simply be turned off in practice, anyway. Therefore, the generalized reduced cost for a single position is evaluated as

\[ r_{ai} := c_{ai} - \sum_{cs} \max \left\{ \sum_{j} \lambda_j \cdot \phi_{cs}(x_j, \theta, \xi_i), 0 \right\} = c_{ai} - \sum_{cs} [\lambda^T \Phi_{csai}]_. \]

Note that \( c_{ai} \) is a constant from (3.7) and vanishes when minimizing the reduced cost. Then, in the ideal case, the angle with the least reduced cost could be found by solving a problem according to

\[ \max_{\theta} \sum_{cs} \left[ \sum_{j} \lambda_j \cdot \phi_{cs}(x_j, \theta, \xi_i), 0 \right]_+ \]

s.t. \( \theta \in [-22.5, 22.5] \).

(3.11)

However, this problem is non-convex, and solving it exactly is out of scope for this thesis. A reliable, albeit more computationally expensive, approach to find the best DOF is to generate a great number \( Q \) of candidate DOFs for many different angles, distributed according to \( \Theta^Q \) in (3.4),
and then choose between these to include in the optimization. The best candidate angle for every isocenter could then be found by solving
\[
(SUB) \quad \argmax_a \sum cs \left[ \sum_j \lambda_j \cdot \phi cs(x_j, \theta_a, \xi) \right]_{+}
\]
s.t. \( \theta_a \in \Theta^Q \)

This procedure can then be reduced to a number of dot product computations
\[
(SUB) \quad \argmax_a \left\{ \sum cs \left[ \lambda^T \Phi csia \right]_{+} \right\}_{a=1,...,Q}.
\]

The result is that the (SUB) program can, for every isocenter \( i \), propose which candidate has the highest potential of improving the solution.

### 3.4 Freezing separate objectives

The results of a multi-objective optimization problem might sometimes be hard to interpret all at once. To facilitate the evaluation of the problem it is possible to freeze one or several objectives at a satisfying point and then adjust the weights for the remaining objectives to explore what can be achieved. When comparing different instances of optimization problems to each other, it may also be helpful to be able to keep some objectives constant. After a program \((P)\) of the form (2.14) is solved, the feasible objective vector \( \hat{z} = (f_T, f_S, f_L, f_B) \) is known. This objective vector combination is necessarily inside the feasible objective region \( Z \), and in particular, it is on the Pareto surface, as established in Section 2.1.1. Define a constant \( V_L := f_L \), and add a constraint to \((P)\) according to
\[
\frac{1}{D_L} \sum_{j \in L} \max\{d_j - D_L, 0\} = V_L.
\]

In this way a specific subset of the Pareto surface can be explored, as stated in Lemma 3.4.1. This can be done with any of the objectives \( f_T, f_S, f_L, f_B \), and also for several at once, if desired. The implications for the dual program, and the fact that the problem size generally grows notably, is explored in Appendix A.2.1.

**Lemma 3.4.1.** Consider a multi-objective program \((MOP)\) as in (2.2), where \( M = 4 \) and \( \hat{z} = (\hat{z}_1, \hat{z}_2, \hat{z}_3, \hat{z}_4) \) is a Pareto optimal objective vector. Consider adding the constraint \( f_4(x) = \hat{z}_4 \) and denote the new program \((MOP^*)\). Any Pareto optimal solution to this new problem is also Pareto optimal to the old.

**Proof.** The constraint maintains \( f_4 \) constant, and can thus be considered to no longer be an objective. This is equivalent to that \((MOP^*)\) is also of the form (2.2) where there are \( M = 3 \) objectives. The feasible region is \( F^* \subset F \), and since the constraint in (3.13) is linear, \( F^* \) is also convex. The \((MOP^*)\) can be solved with the weighted sum approach in (2.3). If the weights are chosen so that \( \omega_1, \omega_2, \omega_3 \geq 0 \), a Pareto optimal solution \( x^* \) is found with respect to \((MOP^*)\). Furthermore, since \( f_4(x^*) \) is constant for any solution to \((MOP^*)\), the solution \( x^* \) is also Pareto optimal to \((MOP)\). \( \square \)
Chapter 4

Model

We propose a number of different models using the methods from Chapter 3. First, the clustering methods applied to the LP produces a model that aims to compete with the RSS approach. Regarding the modelling of rotational DOF, three variations are detailed. A firm foundation is laid by a uniform model where angle nodes are spread out evenly over the interval, and equally for all isocenters. The dynamic model adds the possibility of beam delivery while moving between these nodes. Finally, the last model proposes a method that can suggest the most beneficial angles iteratively in a non-uniform fashion.

4.1 Voxel clustering

Instead of subsampling from the structures, we employ the K-means algorithm from Section 3.1.3 for the voxels of every structure independently. This gives a number of cluster sets, e.g., $\hat{T} = \{k \in \mathcal{K} : \mathcal{C}_k \subseteq \mathcal{T}\}$ where $\hat{T}$ is the total number of clusters in the structure, which are assumed to be known for every structure in beforehand. The cluster indices for $\hat{S}$ and $\hat{L}$ can be defined in a similar fashion. Note that the BOT penalty is identical to before in (2.13), but for the rest of the objective functions to be correct, one has to slightly modify the penalty functions in (2.12) so that every cluster is weighted with the number of voxels it contains according to

$$
\hat{f}_T = \frac{1}{D_T} \sum_{k \in \hat{T}} |\mathcal{C}_k| \cdot \max(D_T - d_k, 0),
$$

$$
\hat{f}_S = \frac{1}{D_S} \sum_{k \in \hat{S}} |\mathcal{C}_k| \cdot \max(d_k - D_S, 0),
$$

$$
\hat{f}_L = \frac{1}{D_L} \sum_{k \in \hat{L}} |\mathcal{C}_k| \cdot \max(d_k - D_L, 0).
$$

(4.1)

Given the high dose gradient nature of the LGK, alternate approaches to clustering with emphasis on surface voxels will be investigated. Let $\partial X_T$ denote the set defined by the surface of the target and define $\partial \mathcal{T} = \{j \in \mathcal{J} : x_j \in \partial X_T\}$ to be the indices of these voxels. They are typically significant for coverage, since isocenters are always positioned inside the volume $X_T$. The interior of the target, $X_T \setminus \partial X_T$, has the voxels $\mathcal{T} \setminus \partial \mathcal{T}$ that are more probable to indirectly be given a sufficient dose. There are presumably significantly higher dose gradients close to the surface and thus a finer mesh of clusters might be necessary there to generate good plans. Given the clustered matrices from Algorithm 4, and the objective functions in (4.1), the clustered program can be expressed

$$(C) \quad \min_t \quad \hat{f}(d, t)
$$

s.t. $d_k = \frac{1}{|\mathcal{C}_k|} \sum_{j \in \mathcal{C}_k} \phi_{cs}(x_j, \xi_i) \cdot t_{esi}, \quad \forall k \in \mathcal{K}$

$$
d \in \mathbb{D}, \quad t \geq 0.
$$

(4.2)
The resulting dual form is described in Appendix A.2.2. We consider a number of different cluster approaches

- **A: Normal uniform clustering**
  Only consider the voxels from the structures \( \bar{J} \), and cluster the target, inner ring and outer ring independently with different fractions.

- **B: Surface independent clustering**
  In addition, cluster the target surface voxels \( \partial T \) independently of the interior voxels \( T \setminus \partial T \) with different fractions.

- **C: Surface exclusive clustering**
  Discard the interior voxels \( T \setminus \partial T \) entirely and cluster the surface voxels \( \partial T \) exclusively for the target structure. The non-target structures are clustered identically to approach A.

### 4.2 Uniform distribution of angles

Rotations to a uniform distribution of discrete angle nodes can be introduced to the model by adding the modifications proposed in Section 3.2 to (P). This gives the optimization problem

\[
(R) \quad \min_t \ f(d,t) \\
\text{s.t.} \quad d_j = \sum_{csai} \phi_{cs}(x_j, \xi_i, \theta_a) \cdot t_{csai}, \quad \forall j \in \bar{J}^* \\
\theta_a \in \Theta^A, \quad \forall a \\
d \in \mathbb{D}, \quad t \geq 0.
\]  

(4.3)

The set of angles is a nested sequence \( \Theta^A \) as described in Section 3.2.2. Note that only the subsampled voxels are used as discussed in Section 3.1.1. The new BOT, similar to (2.13), becomes

\[
\tau_{ai}(t) = \sum_{a=1}^{A} \sum_{i=1}^{I} \max_s \sum_{c=1}^{3} t_{csai}.
\]

This means that every angle will be treated as a shot in the sense that it is the sector with the longest beam-on time for each angle individually that counts. The angles can thus be seen as "subshots" of sorts. Moreover, Lemma 4.2.1 shows that the nested sequences of angles defined in (3.5) guarantee improvements in (R).

**Lemma 4.2.1.** Consider Programs (R) and \( \tilde{R} \) with angles \( \Theta \) and \( \tilde{\Theta} \), respectively. If the angles are chosen such that \( \Theta \subset \tilde{\Theta} \), then \( \tilde{R} \) is a relaxation of \( R \).

**Proof.** Assume that \( \Theta \subset \tilde{\Theta} \), and add constraints \( t_{icsa} = 0, \quad \forall a : \theta_a \in \tilde{\Theta} \setminus \Theta \) to \( \tilde{R} \). This new problem is equivalent to \( \tilde{R} \), which means that \( \tilde{R} \) is a relaxation of \( R \).

### 4.3 Dynamic treatment while rotating

Consider a program \( R \) with a number stationary angles that are uniformly distributed according to (3.4). It is assumed that the RU is able to deliver radiation while rotating between each of its nodes. Assume that this can be done while rotating with constant velocity from a stationary node to its closest neighbor. Since the nodes are uniformly distributed, the angle between every nodes is \( \Delta \theta = 45/A \), and in this section it will be assumed that \( A = 2^n, n \in \mathbb{Z}^+ \). Let the dose rate be denoted \( \tilde{\phi}_{cs}(x, \xi, \theta_a) \), where it rotates from the stationary position at angle \( \theta_a \) and arrives in \( \theta_a + \Delta \theta \). Let \( \tilde{t}_a \) denote the duration of the movement between the nodes. The resulting dose rate must then ideally be such that

\[
\tilde{\phi}_{cs}(x, \xi, \theta_a) \cdot \tilde{t}_a := \int_0^{\tilde{t}_a} \phi_{cs}(x, \xi, \theta_a + \hat{\theta}_a t) \cdot dt, \quad \hat{\theta}_a = \frac{\Delta \theta}{\tilde{t}_a}.
\]
This integral will be approximated according to
\[
\tilde{\phi}_{cs}(x, \xi, \theta_a) \approx \sum_{k=1}^{2^n - 1} \frac{1}{2^n - 1} \cdot \phi_{cs}(x, \xi, \theta_a + \frac{\Delta \theta}{2^n}, k),
\]  
(4.4)
where \(2^n - 1\) denotes the number of subdivisions used for this Riemann sum approximation [20]. Note that the number and position of the subdivisions are designed identically to the selection of angles in Figure 3.5. Furthermore, for simplicity assume that the dynamic shots are also primitive, according to (3.8), so that no collimator changes can be done during the movement. By allowing the new dynamic DOFs into a problem on the form (4.3), the result is
\[
\begin{align*}
\text{(MIP)} \quad \min_{t, \tilde{t}, O} & \quad f(d, t, \tilde{t}) \\
\text{s.t.} & \quad d_j = \sum_{csai} \phi_{ac}(x_j, \xi_i, \theta_a) \cdot t_{csai} + \sum_{ai} \tilde{\phi}_{j, ai} \cdot t_{ai}, \quad \forall j \in \tilde{J}^* \\
& \quad \tilde{\phi}_{j, ai} = \sum_{cs} O_{csai} \cdot \tilde{\phi}_{cs}(x_j, \xi_i, \theta_a), \quad \forall j \in \tilde{J}^* \\
& \quad \sum_{csai} O_{csai} \leq 1, \quad \forall s \\
& \quad O_{csai} \in \{0, 1\} \\
& \quad \theta_a \in \Theta^A, \quad \forall a \\
& \quad d \in D, \quad t, \tilde{t} \geq 0.
\end{align*}
\]  
(4.5)
Here, \(f(d, t, \tilde{t})\) is identical to \(f(d, t)\) except for that the BOT penalty in (2.13) is slightly changed. The BOT with the presence of primitive shots \(t_{ai}\) simply becomes
\[
\tau(t, \tilde{t}) = \sum_{ai} \left( t_{ai} + \max_s \sum_c t_{csi} \right).
\]
The problem in (4.5) is as stated a mixed integer problem (MIP) and is in general difficult and time consuming to solve. A heuristic solution to (MIP) can be found by employing the methods presented in Section 3.3.1 and let the binary variables \(O_{csai}\) be determined separately by a subprogram on the form (3.9). In this way, the master program becomes fully linear according to
\[
\begin{align*}
\text{(D)} \quad \min_{t, \tilde{t}} & \quad f(d, t, \tilde{t}) \\
\text{s.t.} & \quad d_j = \sum_{csai} \phi_{cs}(x_j, \xi_i, \theta_a) \cdot t_{csai} + \sum_{ai} \tilde{\phi}_{j, ai} \cdot t_{ai}, \quad \forall j \in \tilde{J}^* \\
& \quad \tilde{\phi}_{j, ai} = \sum_{cs} O_{csai} \cdot \tilde{\phi}_{cs}(x_j, \xi_i, \theta_a), \quad \forall j \in \tilde{J}^* \\
& \quad \theta_a \in \Theta^A, \quad \forall a \\
& \quad d \in D, \quad t, \tilde{t} \geq 0.
\end{align*}
\]  
(4.6)
Here, the primitive shots, defined by the binary variables \(O_{csai}\), are assumed to be constants acquired from (3.10). The dual formulation is found in Appendix A.2.3. Assume that, in the interest of solving time, all the columns corresponding to dynamic DOFs can be introduced at once into the model. Now, this is probable to generate a lot of “overlap”, in the sense a few of the new DOFs might dominate the others. The solution can be fine-tuned however, without increasing the problem size, by iteratively proposing new primitive shots for positions that are not used and resolving (D). Lemma 4.3.1 states that any unused primitive DOF, \(t_{ai} = 0\), can be re-evaluated to potentially improve the solution. The full procedure is shown in Algorithm 3.
Algorithm 3: Generating a dynamic plan

1. Start with $O_{csai} = 0$ and $\tilde{t}_{ai} = 0$ for all $c, s, a, i$.
2. Solve (D) and acquire the Lagrange multipliers $\lambda$.
3. For any $\tilde{t}_{ai} = 0$ determine $O_{csai} = \begin{cases} 1, & c = \text{argmax} \left\{ \sum_j \lambda_j \cdot \tilde{\phi}_{cs}(x_j, \xi_i, \theta_a) \right\}, \forall c, s. \\ 0, & \text{otherwise} \end{cases}$
4. Evaluate $\tilde{r}_{ai} = c_{ai} - \sum_{csj} \lambda_j \cdot O_{csai} \cdot \tilde{\phi}_{cs}(x_j, \xi_i, \theta_a)$.
5. Stop if $\tilde{r}_{ai} \geq 0 \forall a, i$. Otherwise, repeat from 2.

Lemma 4.3.1. For any variable $\tilde{t}_{ai} = 0$ in the (D) program, the corresponding column can be discarded, and a new primitive shot can be generated that guarantees an equal or better solution.

Proof. If the reduced cost for a primitive DOF $\tilde{r}_{ai} > 0$ then $\tilde{t}_{ai} = 0$ is not a basic variable. If a new column can be produced with $O_{csai}$ such that $\tilde{r}_{ai} < 0$, then the old solution is not optimal to (D).

4.4 Non-uniform distribution of angles

It is interesting to investigate if one can select angles in a more educated manner, which is more problem specific. In other words, allowing the angles $\theta_{ai}$ to vary over every isocenter so that the distribution of angles is not necessary uniform and can vary between isocenters. Ideally, we would like to solve a program according to

\begin{align*}
\text{(NLP)} \quad & \min_{t, \theta} f(d, t) \\
\text{s.t.} \quad & d_j = \sum_{csai} \phi_{cs}(x_j, \theta_{ai}, \xi_i) \cdot t_{csai}, \quad \forall j \in \mathcal{J}^* \\
& \theta_{ai} \in [-22.5, 22.5], \quad \forall a, i \\
& d \in \mathcal{D}, \quad t \geq 0.
\end{align*}

However, the problem is non-convex and solutions will instead be proposed with the column generation methods from Section 3.3.2. The aim is to maximally improve the objective function with the lowest number of DOFs in the model. Consider the program

\begin{align*}
\text{(G)} \quad & \min_t f(d, t) \\
\text{s.t.} \quad & d_j = \sum_{csai} \phi_{cs}(x_j, \xi_i, \theta_{ai}) \cdot t_{csai}, \quad \forall j \in \mathcal{J}^* \\
& \theta_{ai} \in \Theta_{\beta_i}^Q, \quad \forall a, i \\
& d \in \mathcal{D}, \quad t \geq 0.
\end{align*}

Here, $\Theta^Q$ is the set of the $Q$ uniformly distributed candidate angles. Note that $\beta_i$ is an index set, such that $|\beta_i| = A$, that indicates which of these candidates that are allowed to be used in the isocenter $i$. It can be determined by solving a subprogram on the form (3.12). Moreover, we assume that the isocenters are disjoint and can therefore be treated as if they were separate problems. Thus, several DOFs can be introduced at once, one in each isocenter. Algorithm 4 describes how the procedure is iterated to improve the objective, while at the same time increasing the problem size, until the candidates run out. Note that if the algorithm is iterated until all candidates are included, the result will be equivalent to that of a uniform model, but at a much higher computational cost. Therefore, it is of interest to define a maximum of iterations that is less than $Q$. One way is to examine the improvements of the objective function, and break when a certain level of diminishing returns is reached.
<table>
<thead>
<tr>
<th>Algorithm 4: Non-uniform selection of angles</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Start with $\beta_i$ so that $\Theta_{\beta_i} = {0}, \ \forall i$.</td>
</tr>
<tr>
<td>2 Solve (G) and acquire the Lagrange multipliers $\lambda_i$.</td>
</tr>
<tr>
<td>3 Update $\beta_i$ by incorporating $a = \arg\max_a \left{ \sum_{cs} \Phi_c^\top cs a_i \lambda_i \right}_{a=1, \ldots, Q}, \ \forall i$.</td>
</tr>
<tr>
<td>4 Repeat from 2 until an iteration limit is reached.</td>
</tr>
</tbody>
</table>
Chapter 5

Results and Discussion

The models proposed in the previous section have been tested on a number of different patient cases briefly summarized in Table 5.1. The selected patient cases have one target with at least one OAR close to it. We use clinical isocenter positions.

<table>
<thead>
<tr>
<th>Patient case</th>
<th>Feature</th>
<th>GFR41</th>
<th>GFR46</th>
<th>SBR39</th>
<th>SBR41</th>
<th>SBR61</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Nr. Voxels</td>
<td>803</td>
<td>122677</td>
<td>19231</td>
<td>47451</td>
<td>95145</td>
</tr>
<tr>
<td></td>
<td>Voxel res. [mm]</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Nr. Isocenters</td>
<td>28</td>
<td>15</td>
<td>17</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>Nr. OAR</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>2</td>
<td>6</td>
</tr>
</tbody>
</table>

Table 5.1: A brief overview of the cases considered in the testing.

All computations have been carried out in MATLAB [17] with a 64-bit Ubuntu 16.04 LTS with 16 GB RAM and an Intel Core i7-4790K CPU @ 4.00 GHz x 8. If nothing else is specified the RSS fraction will be 5% for target, inner, outer ring and 100% for OAR. Recall that the RSS is stochastic, so the values in every table are averages resulting from 100 different seeds for the random number generator (RNG). Note that these seeds are reused for all the tests. Similarly, the weights in the weighted sum optimization program are $\omega_T = 1.0$, $\omega_S = 0.2$, $\omega_L = 0.04$ and $\omega_B = 0.05/24$.

5.1 Computing the dose rate matrix

The calculation and assembly of the DRM is by far the most time consuming task in the entire procedure. In these tests the dose distributions $\phi_{cs}(x_j,\xi^*)$ from shots in a central position $\xi^*$ have been precalculated and are stored locally. The DRM have been calculated using the rotation invariance from (3.3) and the angles from a nested sequence proposed in (3.5). The total time necessary to load the data from the hard drive, perform necessary translations and rotations, interpolate to the voxel grid, and assemble the DRM can be found in Table 5.2 as a function of angles $A$ in the model. As can be seen, the computation times are practically proportional to the number of angular positions in the model. Note that many of these processes can be carried out in parallel and the GPU can therefore be used to lower the computational time when calculating the DRM. The result is that the time frame would be different in a clinical setting.
Table 5.2: Calculation time of the DRM in seconds for different number of angles.

<table>
<thead>
<tr>
<th>Angles</th>
<th>SBR39 [s]</th>
<th>SBR41 [s]</th>
<th>GFR46 [s]</th>
<th>SBR61 [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9.90</td>
<td>21.1</td>
<td>11.2</td>
<td>133</td>
</tr>
<tr>
<td>2</td>
<td>23.4</td>
<td>43.8</td>
<td>24.1</td>
<td>258</td>
</tr>
<tr>
<td>4</td>
<td>47.6</td>
<td>90.8</td>
<td>47.7</td>
<td>523</td>
</tr>
<tr>
<td>8</td>
<td>97.0</td>
<td>192</td>
<td>104</td>
<td>1070</td>
</tr>
</tbody>
</table>

The problem size can become demanding for a 16 GB RAM after around 10-15 angles, depending on case. However, by prioritizing memory according to Section 3.1.2, the memory size of the DRM can be significantly reduced, with the drawback that the evaluation phase becomes prolonged as shown in Figure 5.1. The RSS fractions in those tests are 5% of the voxels in the target, 2% of the voxels in the ring and no sampling in the OAR. The amount of memory saved depends on the relative structure size. For GFR41, which has a matrix with about 540 million elements normally, one can only use 3 angles before the RAM fails during normal circumstances. However, with the method in question, one could run instances of optimization problems of up to 18 angles before failure. The hypothesis is that prioritizing memory in this way can make any type of new DOF more tractable.

![Figure 5.1](image)

(a) A memory reduction of 87.3%.
(b) A memory reduction of 90.4%.

Figure 5.1: Prioritizing memory can significantly reduce the RAM usage, but the final evaluation time is prolonged.

Even though the method was primarily conceived for the purpose of relieving the memory requirements, one can see a significant decrease in total computation time as well in some cases. The final evaluation times, however, are always higher in the reduced case due to the extra step involved, but it seems to level out relatively quickly as seen in Figure 5.2. The evaluation time is extended with up to about 7% of the DRM calculation time. A great deal of information is disregarded in favor of reduction in used RAM and the evident drawbacks are that one cannot change the structures nor the RSS fraction without rerunning the generation, which is otherwise relatively easy if one possesses the full DRM.
Figure 5.2: Prioritizing memory can significantly reduce the RAM usage, but the final evaluation time is prolonged.

Note that the memory saved by this method can be used in combination with the column generation approaches proposed in Section 3.3 for improved performance. As discussed earlier, there is a lot to gain from expanding the number of candidates or subdivisions in these algorithms. With this method one can calculate a large amount of candidates since only the sampled voxels are needed in the optimization step. Then, only the columns that enter the model actually have to be completely determined, making the memory requirements significantly less challenging.

5.2 Clustering in the inverse planner

The process can be divided into two parts, one where the goal is to group the voxels and produce a reduced matrix with minimal loss of information, and another where the said matrix is employed in the inverse planner optimization to produce new plans. The clustering is carried out using K-means, as outlined in Algorithm 4, and the plans are generated using the program (C) in (4.2) from Section 4.1. The relation clusters-to-voxels is 5% in every structure and case, unless stated otherwise.

5.2.1 The K-means performance

The starting guess for the K-means algorithm is found by simply grouping voxels close to each other in the matrix with the optimistic assumption that they are in that way close in space as seen in Figure 5.3a. A starting guess where 50% of the clusters are distributed to the surface voxels is illustrated in Figure 5.3b, making the resolution significantly higher in that area. A starting guess randomly assigning voxels to clusters was tested with very poor results in comparison.

Figure 5.3: A cross section of a tumour belonging to GFR46 where every cluster starting guess have randomly been given a color assigned to each of the member voxels.
The K-means algorithm typically terminates after about 40-50 iterations, but by investigating the matrix $L_2$-norm as a function of iterations it is evident that a value very close to the final is reached with much fewer iterations than necessary for termination. This seems to be the case regardless of fraction between number of clusters and voxels, even though the norms are notably better for higher fractions. Often, such a value can be reached after only about 5-10 iterations as can be seen in Figure 5.4. Thus, to save time a iteration limit of 15 iterations is imposed. This iteration limit is very conservative to ensure that the norm is minimal. It could be possible to save time by evaluating the norm in every iteration and break the algorithm earlier. That however, would in turn prolong the time for every iteration, and would require an exact limit for when the norm is considered to have converged.

![Figure 5.4: Matrix $L_2$-norm improvement as a function of fraction and K-means iterations.](image)

The results of clustering the tumour in GFR46 is shown in Figure 5.5a, which seems to be independent of the starting guess. It does not seem possible to encourage a denser clustering around the surface by designing the starting guess. If the surface and core voxels are deliberately solved separately however, one arrives in a result according to Figure 5.5b. These starting guesses for the K-means algorithm are arguably quite crude, the voxels seem to be connected but are far from the desired shape. Anyway, Figure 5.5a suggests that few of the characteristics of the starting guesses remain in the final clusters. However, there might be benefits to consider a more geometrically advanced method to construct these starting guesses.

![Figure 5.5: A cross section of a tumour belonging to GFR46 where each voxel is colored according to the cluster it belongs to.](image)

Note that in the end the motive is to generate good plans, which does not necessarily require the matrix norm of the mismatch between the clustered and original DRM to be minimal. It functions as a measure of similarity between the matrices, but some information might be much more valuable than other from a treatment planning perspective. By clustering surface and core voxels separately, one might arrive in a worse matrix norm than if clustered normally. However, it is hypothesized that the final plans can be made better with such distributions. Something similar can be said when discarding core voxels. Another way of encouraging separation of surface voxels is by instead considering a weighted $L_2$-norm, where surface elements are valued significantly higher.
5.2.2 Treatment planning with clustered models

In Table 5.3 and 5.4 the clinical metrics are compared between the three clustering methods from Section 4.1 and RSS. 'Normal' refers to the standard full clustering approach described in approach A. 'Indep' refers to the approach B, where surface voxels are clustered independently of core voxels and the distribution of clusters are 50-50% in surface and core. 'Surface' refers to a model where all voxels belonging to the core of the target are discarded and all the clusters are only distributed over the surface, as described in approach C. It can be concluded that it is difficult to correctly reproduce the coverage using the normal clustering approach, resulting in a lower PI in every case. Outstanding selectivity is achieved but at the cost of a significant deterioration of the coverage. Figure 5.5a seems to indicate that the clusters are often relatively big on the surface, and sometimes extend notably into the interior of the tumour. It is evident that the gradients are too large within these clusters and the mean dose in a cluster becomes a poor representation of the member voxels. Currently, the clusters are represented with the average of the member voxels, but by considering a different weighted average of the member voxels there might be much to gain. For example, by emphasizing the voxels with lower doses, one would achieve a more "pessimistic" representation of the clusters that potentially could lead to an improvement in coverage.

<table>
<thead>
<tr>
<th>Method</th>
<th>SBR39</th>
<th>SBR61</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cov</td>
<td>Sel</td>
<td>GI</td>
</tr>
<tr>
<td>Norm</td>
<td>0.800</td>
<td>0.955</td>
</tr>
<tr>
<td>Indep</td>
<td>0.871</td>
<td>0.945</td>
</tr>
<tr>
<td>Surf</td>
<td>0.903</td>
<td>0.932</td>
</tr>
<tr>
<td>RSS</td>
<td>0.923</td>
<td>0.937</td>
</tr>
</tbody>
</table>

Table 5.3: Comparison of clinical metrics between different variations of clustering and RSS.

In general, it is difficult to see a trend in the clinical metrics as seen in Table 5.3 and 5.4. Notably, the clustering methods seem to produce better GI in practically every case, and the best coverage is achieved by the surface exclusive method in three out of four cases. In terms of BOT, there seems to be no evident trend, except for that normal clustering in general produces slightly shorter plans, which might explain why the coverage is poor. It is possible that a higher fraction of clusters in relation to voxels is needed before the difference become more distinguishable.

<table>
<thead>
<tr>
<th>Method</th>
<th>SBR41</th>
<th>GFR46</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cov</td>
<td>Sel</td>
<td>GI</td>
</tr>
<tr>
<td>Norm</td>
<td>0.828</td>
<td>0.994</td>
</tr>
<tr>
<td>Indep</td>
<td>0.930</td>
<td>0.982</td>
</tr>
<tr>
<td>Surf</td>
<td>0.948</td>
<td>0.978</td>
</tr>
<tr>
<td>RSS</td>
<td>0.937</td>
<td>0.982</td>
</tr>
</tbody>
</table>

Table 5.4: Comparison of clinical metrics between different variations of clustering and RSS.

The biggest drawback with the method is probably the longer computation times, which are correlated with the size of the problem both in terms of voxels and DOF. The total clustering time can vary significantly between cases but are relatively independent of clustering method, as seen in Table 5.5. It is evident that the clustering time is very sensitive to the size of the matrix, but the fact that the cluster time for GFR46 was so elevated despite having few isocenters seems to suggest that the number of voxels is more definitive. Note that by allowing the K-means algorithm to stop earlier, a lot of time could be saved. However, from a practical point of view, clustering can commence as soon as the isocenter positions are placed and the DRM is computed. As such, it can possibly be computed in parallel while determining, e.g., prescription dose magnitude, and is very likely finished before any optimization can be started. Furthermore, different structures can
be clustered independently and thus potentially in parallel as well. Moreover, it might be possible to combine RSS and clustering to further reduce the computation times.

<table>
<thead>
<tr>
<th>Method</th>
<th>SBR39</th>
<th>SBR41</th>
<th>GFR46</th>
<th>SBR61</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>3.35</td>
<td>32.9</td>
<td>129</td>
<td>189</td>
</tr>
<tr>
<td>Indep</td>
<td>3.60</td>
<td>32.0</td>
<td>126</td>
<td>179</td>
</tr>
<tr>
<td>Surface</td>
<td>3.66</td>
<td>31.2</td>
<td>123</td>
<td>179</td>
</tr>
</tbody>
</table>

Table 5.5: The K-means clustering times in seconds for every considered case.

An attempt at illustrating the PI as a function of the fraction clusters-to-voxels is shown in Figure 5.6. The tests were performed while subjecting the objectives corresponding to GI and BOT, i.e. \( f_L \) and \( f_B \), to constraints maintaining them constant, as described in Section 3.4. 'Full' refers to the result from a plan run with no RSS and is thus a reference. The results from 20 RSS runs with different RNG seeds are also included. The fractions on the x-axis refers to the target only and the sampling fractions were held constant at 5% in the inner and outer ring structures over the tests.

One interesting fact is that the surface voxels have such a high significance in the model. It is surprising that a model where exclusively surface voxels are used in the target performs better than the other models, with respect to the PI, while being slightly faster to compute. This model is completely blind to the interior voxels, which could potentially be a problem for larger tumours. Occasionally, the clustering methods with surface emphasis even produces a better PI than the full model where every single voxel is used. Note however that the freezing of the outer ring objective might give slightly different implications when comparing RSS and clustering, which might influence the results. The BOT on the other hand can be reliably frozen, since it can be expressed linearly. At the moment, the thickness of the surface is only one voxel and one can assume that there might be more to gain from experimenting with that thickness to include even more of the most important voxels. Alternatively, to explore clustering methods with the explicit goal to generate higher resolution towards the surface, that not necessarily uses the DRM.
The advantage is that plans are deterministic instead of stochastic as a function of the RSS, at least up to the starting guess for the k-means heuristic. As seen in Figure 5.7 the variance of the plan quality for RSS sometimes grows notably for lower fractions. Clustering results also seem to become significantly worse for too small fractions where RSS outperforms them greatly, however it is arguable how relevant those fractions are in practice. It is notable that the RSS seems to stay more true to the full model in general, while the clustered model seems to behave less predictably. In other words, RSS give a relatively static result close to the reference value, while clustering results seem to fluctuate considerably. It can be concluded that RSS competes relatively well with clustering which is a reaffirmation that the current model is viable.

5.3 Comparison between the models with new degrees of freedom

The performance in terms of cost function value, solving times and clinical metrics will be compared for the models incorporating the new rotational DOF proposed in Chapter 4. The uniform distribution of angles $\Theta^A$ is found from (3.4) and (3.5). Three models will be investigated:

- **Uniform**: The model resulting from (4.3), with $A$ uniformly distributed stationary angles will be referred to as ($R_A$).

- **Dynamic**: The model resulting from the combination of (4.6) together with Algorithm 3 in Section 4.3 will be referred to as ($D_A$). For $A$ uniformly distributed stationary angles and another $A$ additional intermediary trajectories for dynamic shots. The approximation of the dynamic dose rates are approximated from (4.4) with $n = 1$ subdivisions, which is equivalent to the dose rate for an angle in the middle of the trajectory.

- **Non-uniform**: The model resulting from (4.7), Section 4.4, and $A$ iterations of the Algorithm 4 will be referred to as ($G^Q_A$). It can choose from $Q$ uniformly distributed candidate angles, but note that the first problem is always solved with the zero degree angle.

5.3.1 Optimization cost function and solve times

The importance of the four different objectives and their relation to each other are illustrated in Figure 5.8 and 5.9 as functions of number of angles. The gains are principally won from the inner ring objective in three out of four cases, suggesting improvement of selectivity. All the cases have a similar behaviour and distribution of objective values except SBR61 where coverage is receiving an overwhelming share of the penalties. This is presumably since it is unable to comply with all the constraints imposed by the adjacent OAR without sacrificing a significant portion of plan quality.
In general, it seems as if the gain from adding rotations to the program is limited and exhausted quickly. The objective function stagnates between 93.0% and 95.5% of the original value depending on case, even for a high number of angles in a uniform model. Recall Lemma 4.2.1 that clarifies that a uniform model with many angles constitutes a lower bound for the improvements offered by the rotational DOF for all the models. For all cases, the uniform model comes within 1% percentage of this lower bound after 4 angles and within 0.1% after 8. The uniform selection of angles was designed to generate angles with the largest distance from already existing ones, which might explain the steep drop of value from new DOFs since the columns become less unique the more angles are added.

When comparing the cost function of the three models as shown in Figure 5.10 and 5.11, it is evident that the dynamic model almost always starts off best. It must be kept in mind that the dynamic model has slightly more DOFs than the other cases, in the form of primitive shots, so it is expected that it has a better start. It is however interesting that it stays competitive despite heavily relying on the primitive shots. Furthermore, the non-uniform model seems to improve the objective function notably more with the exact same problem size. In particular, the non-uniform model can achieve an objective value within one percentage of the lower bound with only one additional angle.
Figure 5.10: Comparison of the average objective function improvement between models for 100 different RNG seeds.

The approximation employed for the dynamic DOFs is relatively crude, and it is reasonable to assume that more subdivisions in the Riemann approximation of the integral determining the dynamic dose distribution might improve the results. This would make the dynamic DOFs more unique, in the sense that a dynamic shot should distribute the dose over a larger area than a normal shot. However, as concluded in Lemma 4.2.1, the dynamic program \((D_A)\) is still bound from below by \((R_{2A})\), given that there is only \(n = 1\) subdivisions. The objective value of the dynamic model is very close to the lower bound and one can conclude that not much more is to be gained anyway. It is however possible that the result could become worse.

Figure 5.11: Comparison of the average objective function improvement between models for 100 different RNG seeds.

Note that the optimization times grow almost proportionally to the number of angles, as seen in Table 5.6. The optimization times include the assembly of the system matrices along with eventual column generation operations. The times to solve the optimization problems are typically the most important, and they are contained within a minute, which is typically considered acceptable in a clinical context. The worst case, SBR61, could due to memory restrictions not run the dynamic method for \(A = 8\) angles. The dynamic and non-uniform models always take longer time due to solving the LP several times along with a number of extra dot product calculations. Empirically, the dynamic model iterates between 3-5 times before convergence, which slightly improves the objective value but extends the computation time. However, the majority of the improvement is achieved after one iteration and it would be possible to terminate the algorithm early.
Table 5.6: The optimization times in seconds for the three different models.

<table>
<thead>
<tr>
<th></th>
<th>SBR39</th>
<th>SBR61</th>
<th>SBR41</th>
<th>GFR46</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Uni</td>
<td>Dyn</td>
<td>Non</td>
<td>Uni</td>
</tr>
<tr>
<td>1</td>
<td>0.47</td>
<td>1.51</td>
<td>0.83</td>
<td>5.12</td>
</tr>
<tr>
<td>2</td>
<td>0.85</td>
<td>2.66</td>
<td>1.78</td>
<td>9.92</td>
</tr>
<tr>
<td>4</td>
<td>1.68</td>
<td>4.86</td>
<td>3.16</td>
<td>20.5</td>
</tr>
<tr>
<td>8</td>
<td>3.49</td>
<td>10.4</td>
<td>4.97</td>
<td>47.1</td>
</tr>
</tbody>
</table>

Note that both for angle generation and angular motion, in interest of time, one includes several columns at once to the model in contrast to traditional simplex and column generation. To get the best out of the methods one would ideally want to solve a problem after the inclusion of at least every angle position, in this way one would have the maximal information to make the most educated estimation of what DOFs should enter the model next. In a context where one can reuse old solutions as starting guesses, in a “warm start”, this approach could be feasible due to the presumably short solving times for subsequent programs. In the current tests, the problem must be solved from the beginning every time, involving a separate phase-one problem to find a feasible starting guess. As mentioned, the current program is solved in the dual, whereas a solution is only primal feasible to a program with new DOFs. This would demand that subsequent runs have to be solved in the primal program which might be significantly more demanding.

5.3.2 Distribution of beam-on time

The distribution of the BOT over the positions in the uniform (R₈) and the dynamic (D₄) models are illustrated in Figures 5.12 to 5.15. Note that the ∼ indicates a primitive dynamic shot in the right figures. Also, note that the distribution is very sparse, suggesting that only a fraction of the introduced angles are actually employed in the plan. This indicates that a very similar plan can be achieved for a smaller problem by finding these most significant angles in every isocenter. The sparsity also implies that there are a high number of unused DOFs that can potentially save a lot of time and memory in combination with method to reduce RAM usage in Section 3.1.2.

Figure 5.12: The distribution of BOT for case SBR39 for the uniform (left) and dynamic (right) model. The angles proposed by one iteration of the non-uniform model is also marked (red).

Note that there is only a slight difference between the BOT distribution between the dynamic and static uniform model, seen from comparing the left and right sides of the figures in this section. It seems to suggest that a very similar plan can be constructed with the help of primitive shots, which are much less complex than normal, and indicates further investigation of dynamic planning would be of interest.
Figure 5.13: The distribution of BOT for case SBR61 for the uniform (left) and dynamic (right) model. The angles proposed by one iteration of the non-uniform model is also marked (red).

The specific angle values proposed by one iteration of the non-uniform model ($G^1_8$) are visualized in red on top the BOT distributions. Figures illustrating another iteration ($G^3_8$) can be found in Appendix B. It is evident that the proposed angles in general come very close to the angles chosen in the full programs where all angles are available. It also seems common that the original zero degree position is seldom used and it might be reasonable to believe that the removal of those columns would still produce a very similar solution, while simultaneously reducing the problem size by a factor 2. It might be interesting to explore the possibility to remove obsolete DOFs in tandem with the introduction of new promising ones.

Figure 5.14: The distribution of BOT for case SBR41 for the uniform (left) and dynamic (right) model. The angles proposed by one iteration of the non-uniform model is also marked (red).

Note that in the dynamic case, time spent in an intermediary trajectory $\sim$ is the time the machine has to traverse the interval and one can imagine that when $\tilde{t} \approx 0 \rightarrow \tilde{\theta} \approx \infty$. The case when $\tilde{t} = 0$ is simple, since no time is spent there means no dose is delivered and thus in practice it can simply be considered as a normal movement between shots, where the sectors are in a blocked position. Furthermore, the fact is that an inherit problem with the LGK is that when the sector duration is too low, the command will not be carried out during treatment, but since the duration is so low the losses are usually minimal. This leads to the realization problem, where the plans suggested by the optimization must be adapted to the specific machine conditions. Note that between the iterations in Algorithm 3, one can manually make sure that only DOFs with sufficient time are allowed. Currently, a new primitive shot is proposed if it is receiving zero time, but it would, e.g., be possible to increase that cutoff to the minimum value for a shot to be delivered.

Figure 5.15: The distribution of BOT for case GFR46 for the uniform (left) and dynamic (right) model. The angles proposed by one iteration of the non-uniform model is also marked (red).
5.3.3 Improvement of the clinical metrics

The resulting clinical metrics, defined in Section 2.4.4, for the different models will be shown in the form of tables where the best value in every column is displayed in bold.

The uniform model

The clinical metrics along with the relative change of objective values, 'Obj', are displayed in tables 5.7 and 5.8. They seem to suggest that there are a number of patterns in the clinical metrics when introducing the rotational DOFs. The coverage seems to be approximately the same, while the BOT is in general longer. The GI and selectivity are always improved, albeit sometimes at insignificant levels. It is remarkable that despite the improvements in the objective function, the clinical metrics do not seem to show an equally convincing behaviour. The differences are sometimes in other cases so small that they are better explained with statistical error than true improvement.

This is partly explained by the non-continuous nature of the metrics and the discrepancy between them and the objective functions. It must be kept in mind that the optimization is carried out with respect to the piece-wise linear functions in (2.12), which does not guarantee improvement of the clinical metrics.

<table>
<thead>
<tr>
<th>A</th>
<th>Cov</th>
<th>Sel</th>
<th>GI</th>
<th>BOT</th>
<th>Obj</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.923</td>
<td>0.937</td>
<td>3.23</td>
<td>83.4</td>
<td>1.00</td>
</tr>
<tr>
<td>2</td>
<td>0.925</td>
<td>0.940</td>
<td>3.18</td>
<td>87.8</td>
<td>0.969</td>
</tr>
<tr>
<td>4</td>
<td>0.925</td>
<td>0.941</td>
<td>3.17</td>
<td>90.4</td>
<td>0.954</td>
</tr>
<tr>
<td>8</td>
<td>0.925</td>
<td>0.941</td>
<td>3.17</td>
<td>90.7</td>
<td>0.950</td>
</tr>
</tbody>
</table>

Table 5.7: Clinical metrics as a function of number of uniformly distributed angles in the model.

Table 5.7 indicates that SBR61 achieved the most notable changes, increasing the selectivity with one percent at the cost of 45 minutes of BOT however. As noted before, the coverage is remarkably poor in that case, and the program is willing to make significant sacrifices in BOT for even small improvements of this metric. Another remarkable behaviour is displayed by case SBR41, which seems to have found a way to deliver practically the same plan at a lower BOT cost. Moreover, the patient case GFR46 seems to be receiving the lowest impact on the clinical metrics despite showing one of the better improvements in terms of objective function. Note that for this case, the coverage and selectivity are the best of all the cases, and improving them further is difficult. However, the conformity between the objective function and the clinical metrics in general seems poor.

<table>
<thead>
<tr>
<th>A</th>
<th>Cov</th>
<th>Sel</th>
<th>GI</th>
<th>BOT</th>
<th>Obj</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.937</td>
<td>0.982</td>
<td>2.67</td>
<td>126</td>
<td>1.000</td>
</tr>
<tr>
<td>2</td>
<td>0.936</td>
<td>0.983</td>
<td>2.65</td>
<td>120</td>
<td>0.955</td>
</tr>
<tr>
<td>4</td>
<td>0.935</td>
<td>0.983</td>
<td>2.64</td>
<td>119</td>
<td>0.937</td>
</tr>
<tr>
<td>8</td>
<td>0.935</td>
<td>0.983</td>
<td>2.64</td>
<td>118</td>
<td>0.930</td>
</tr>
</tbody>
</table>

Table 5.8: Clinical metrics as a function of number of uniformly distributed angles in the model.

RSS is the source of the statistical uncertainties in the model, and it might be reasonable to suggest that it limits what can be achieved in terms of clinical metrics. If the optimization is performed with 5% of the voxels, these might not be representative for the other 95%. The issue is that the possibilities of improving coverage approaching 1 is continuously smaller, since most of the under-dosed voxels will be discarded by the RSS. It is notable for example that plans for difficult cases like SBR61 have a high variance and are very sensitive to the RSS fraction. A more surface
oriented sampling will probably improve these results. Since all shots are placed on the inside of the target, it is reasonable to suggest that the interior voxels receive approximately an equal or greater dose than the surface voxels. Therefore, when building the dose to reach and cover the surface voxels, it is probable that the central voxels receive a sufficiently high dose indirectly, even if they are not sampled.

In all of the patient cases we use the clinical isocenter positions, not generated by the packing algorithms, which might have a considerable influence on the outcome. It would be interesting to further investigate the possibility to position the isocenters with the rotation in mind. Consider for example the packing algorithm in Figure 2.15, one can imagine introducing new template shots corresponding to the rotated shots. These isocenter positions would be more suitable for investigation of the improvements offered by rotations. In this way, one should also presumably be able to cover the target more reliably due to having more template shots to choose from.

The dynamic model

The clinical metrics achieved from the dynamic model are displayed in Table 5.9 and 5.10. The BOT spent treating with dynamic shots are denoted 'Dyn', measured in minutes as well. Note that a significant amount of radiation time is delivered by the dynamic primitive shots despite being less flexible and equally expensive in terms of BOT penalty. A rough foundation can be laid by the simply configured shots in order to later fill in the details with stationary more flexible shots. One can hypothesize that there is a lot to gain from employing a similar method in movements between isocenters instead of angles, where there are presumably more to be gained. The movement between angles is relatively small in comparison to the distance between isocenters, and much more unique DOFs could possibly be found by treating while moving between these isocenter positions.

<table>
<thead>
<tr>
<th>A</th>
<th>Cov</th>
<th>Sel</th>
<th>GI</th>
<th>BOT</th>
<th>Dyn</th>
<th>Obj</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.925</td>
<td>0.939</td>
<td>3.20</td>
<td>84.6</td>
<td>23.1</td>
<td>0.979</td>
</tr>
<tr>
<td>2</td>
<td><strong>0.926</strong></td>
<td><strong>0.941</strong></td>
<td>3.17</td>
<td>89.2</td>
<td>32.6</td>
<td>0.957</td>
</tr>
<tr>
<td>4</td>
<td>0.925</td>
<td><strong>0.941</strong></td>
<td>3.17</td>
<td>90.3</td>
<td>29.9</td>
<td>0.951</td>
</tr>
<tr>
<td>8</td>
<td>0.925</td>
<td><strong>0.941</strong></td>
<td>3.17</td>
<td>90.8</td>
<td>37.6</td>
<td><strong>0.948</strong></td>
</tr>
</tbody>
</table>

Table 5.9: Clinical metrics as a function of angles in the dynamic model.

Note that the total BOT is very similar to the uniform model, but here it is notable that a significant portion is delivered from the dynamic setting. One could also argue that time in motion is not as bad as stationary time, since that would otherwise not be used in the model and is thus a missed opportunity. To emphasize this, the time penalty could be lower for these dynamic shots. This would then intuitively result in that even more of the treatment would be carried out in a dynamic setting, which would further reduce total treatment time. On the other hand it could possibly be "cheating" to lower the cost of these DOFs since the real beam delivery time is still the same.

<table>
<thead>
<tr>
<th>A</th>
<th>Cov</th>
<th>Sel</th>
<th>GI</th>
<th>BOT</th>
<th>Dyn</th>
<th>Obj</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>0.936</strong></td>
<td>0.982</td>
<td>2.66</td>
<td>121</td>
<td>35.7</td>
<td>0.968</td>
</tr>
<tr>
<td>2</td>
<td>0.935</td>
<td><strong>0.983</strong></td>
<td>2.64</td>
<td><strong>118</strong></td>
<td>41.3</td>
<td>0.940</td>
</tr>
<tr>
<td>4</td>
<td>0.935</td>
<td><strong>0.983</strong></td>
<td>2.64</td>
<td><strong>118</strong></td>
<td>41.8</td>
<td>0.931</td>
</tr>
<tr>
<td>8</td>
<td>0.935</td>
<td><strong>0.983</strong></td>
<td>2.64</td>
<td><strong>118</strong></td>
<td>45.5</td>
<td><strong>0.927</strong></td>
</tr>
</tbody>
</table>

Table 5.10: Clinical metrics as a function of angles in the dynamic model.

The integer problem (MIP) in (4.5) is in general solvable, where branch and bound methods will
eventually reach a global optimum, albeit at a high cost in computation time. Another approach would be to instead solve a problem where the integer constraint on the variables are relaxed to a linear problem. However, the provided solution of those variables are seldom integer, and thus, significant rounding errors might occur. The presented column generation approach program (D) from Section 3.3.1 seems to provide competitive solutions in a relatively quick fashion, although there is no guarantee that the solution is a global optimum to the (MIP).

The non-uniform model

The clinical metrics as a function of the iterations of Algorithm 4.7 can be found in Table 5.11. Note that the first iteration is equivalent to the uniform model for one angle. All tests have been carried out with \( Q = 16 \) candidates except for SBR61 which could only manage \( Q = 8 \) because of memory restrictions. Note that the memory prioritization method was not employed, but for future future work it is expected to considerably reduce the memory requirements. It can be argued that the non-uniform approach manages to reach the final objective values of the uniform model, but with fewer iterations. For example, the difference between 4 and 8 iterations seem to be insignificant for almost all cases. As earlier remarked with regard to the objective functions, the clinical metrics seem to be very close to the final values already after one iteration.

<table>
<thead>
<tr>
<th></th>
<th>SBR39</th>
<th></th>
<th>SBR61</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cov</td>
<td>Sel</td>
<td>GI</td>
<td>BOT</td>
<td>Obj</td>
<td>Cov</td>
<td>Sel</td>
</tr>
<tr>
<td>1</td>
<td>0.923</td>
<td>0.937</td>
<td>3.23</td>
<td>83.4</td>
<td>1.00</td>
<td>0.835</td>
<td>0.910</td>
</tr>
<tr>
<td>2</td>
<td>0.925</td>
<td>0.940</td>
<td>3.18</td>
<td>88.9</td>
<td>0.958</td>
<td>0.837</td>
<td>0.918</td>
</tr>
<tr>
<td>4</td>
<td>0.925</td>
<td>0.941</td>
<td>3.17</td>
<td>89.9</td>
<td>0.951</td>
<td>0.837</td>
<td>0.920</td>
</tr>
<tr>
<td>8</td>
<td>0.925</td>
<td>0.941</td>
<td>3.17</td>
<td>90.5</td>
<td>0.949</td>
<td>0.838</td>
<td>0.921</td>
</tr>
</tbody>
</table>

Table 5.11: Clinical metrics as a function of iterations in the non-uniform model.

Consider the nonconvex program in (3.11), the candidate approach is a rough approach to find a good solution for it. Since the problem is constrained and is of only one variable, it might be possible to find a local optima in a more efficient manner than the current. However, this requires that \( \phi_{cs}(x, \theta, \xi) \) and its gradients with respect to \( \theta \) can be evaluated efficiently. Moreover, it is important to emphasize that the more columns that can be used as candidates the better. With more columns it is more probable that the chosen positions are closer to the actual optimum of (3.11), and become even more helpful in subsequent iterations. The drawback is however slightly extended computation times and memory requirements, but this can as previously mentioned be reduced significantly by employing the methods in 3.1.2.

<table>
<thead>
<tr>
<th></th>
<th>SBR41</th>
<th></th>
<th>GFR46</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cov</td>
<td>Sel</td>
<td>GI</td>
<td>BOT</td>
<td>Obj</td>
<td>Cov</td>
<td>Sel</td>
</tr>
<tr>
<td>1</td>
<td>0.937</td>
<td>0.982</td>
<td>2.67</td>
<td>126</td>
<td>1.00</td>
<td>0.953</td>
<td>0.980</td>
</tr>
<tr>
<td>2</td>
<td>0.935</td>
<td>0.983</td>
<td>2.65</td>
<td>119</td>
<td>0.941</td>
<td>0.952</td>
<td>0.981</td>
</tr>
<tr>
<td>4</td>
<td>0.935</td>
<td>0.983</td>
<td>2.64</td>
<td>118</td>
<td>0.931</td>
<td>0.952</td>
<td>0.981</td>
</tr>
<tr>
<td>8</td>
<td>0.935</td>
<td>0.983</td>
<td>2.64</td>
<td>118</td>
<td>0.929</td>
<td>0.952</td>
<td>0.981</td>
</tr>
</tbody>
</table>

Table 5.12: Clinical metrics as a function of iterations in the non-uniform model.

5.4 Evaluating the rotation invariance assumption with Monte Carlo simulations

The most accurate dose estimates available are, as previously mentioned, generated with MC methods where the photons and electrons are simulated in great detail. However, in the interest of time these simulations are not feasible in a real world situation since they typically require a lot
of computational effort. Thus, in LGP and especially in the optimization, the dose distributions are generated by an algorithm that among other simplifications consider the head as a uniform body of water [5]. This approximation is considered acceptable for the purpose even though it naturally introduces a significant discrepancy in the model. Thus, we want to estimate whether the error introduced by the rotational invariance is significant in relation to the known current approximation errors. The assumption is expected to become worse the farther from the center of the head the dose is computed, where the rotational symmetry is smaller. By comparing the approximated distribution from LGP to an accurate MC simulation using a water phantom, the deviations can be examined and evaluated.

In the following tests, we use the patient case referred to as GFR08 where the small tumour is located close to the right ear, a non-central position in the head. The patient geometry was manually segmented and sculpted from a CT image of the head. A single isocenter was manually positioned and we considered the distribution resulting from the 4 mm collimator in sector 1, the other sectors were blocked. The MC simulations were carried out, using the in-house software Pegasos, based on the established MC code Penelope [21]. We use 1 billion photons, $10^3$ eV and $10^5$ eV absorption energies for photons and electrons respectively. The head was modeled as water, while the surroundings medium was assumed to be vacuum. The result was read from a dose box with size $2.525 \times 2.525 \times 2.525$ cm centered in the isocenter and divided into $101 \times 101 \times 101$ bins, resulting in a voxel size of $0.25 \times 0.25 \times 0.25$ mm. Monte Carlo simulations were performed for 9 rotation angles uniformly distributed over the interval $[-22.5, 22.5]$.

5.4.1 Comparison of dose distribution

The dose distributions are illustrated in Figure 5.16 for a plane intersecting the z-coordinate closest to the isocenter position. The rotation invariance was exercised by rotating the coordinate system of the lower resolution LGP data, and then interpolating the high resolution MC dose, for the angle in question, onto these grid points. In practice, this LGP data has to be rotated and interpolated to the fixed voxel grid, a third coordinate system, which of course introduces additional errors. However, to evaluate the assumption it is argued that it is desirable to reduce the number of interpolations and their inherent errors. The outputs from the dose calculations result in different physical units depending on method. The output from the Monte Carlo simulation is in deposited dose in eV/g per primary particle while the dose rates kernel are in Gy/min. However, in this case there is a trivial scaling factor relating the different results. Thus, a normalization constant was chosen such that the dose rate in the center bin of the MC calculation is equal to the dose rate of the isocenter of the LGP calculations.

![Figure 5.16: Comparison between MC and LGP distributions for 0 and 22.5 degrees rotation.](image)

(a) 0 degree rotation  
(b) 22.5 degree rotation

The LGP and MC dose distributions for zero and maximal rotation are shown in Figure 5.16. Overall, LGP produces a good approximation, and there are practically no discernible differences
between Figure 5.16a and 5.16b. It is evident that there are some deviations between the LGP and the MC distribution from the start, which is normal since the LGP algorithm is a deliberate trade-off between quality and computation times. Note that there is little symmetry close to the surface of the head. This implies that when rotating the collimator body, it is likely that small rotations indicate that many beams have to travel through a varying amount of tissue, which would make the approximation worse. Furthermore, there are of course statistical errors that may manifest themselves to different degrees in the tests, given the modelling errors in LGP and MC, and additional errors introduced from the normalization constant selected.

5.4.2 Dose regions and $L_2$-error

It is of interest to more thoroughly investigate the behaviour of the errors introduced with the rotations and the rotation invariance assumption. It can be argued that, for the approximations to be acceptable, the $L_2$-norm of the error between the LGP and MC distributions should be approximately constant when rotating. This is assumed to be equivalent to that the discrepancies contributed from the rotations are small. Let $n$ denote the number of grid points from the LGP distribution that are contained inside the dose box defined for MC, used for the interpolation in the previous section. Let $d_{LGP}^k$ and $d_{MC}^k$ be the dose in grid point $k = 1, \ldots, n$ for respective method, and define the $L_2$-error as

$$
\|d_{LGP} - d_{MC}\|_2 = \left[\sum_{k=1}^{n} |d_{LGP}^k - d_{MC}^k|^2\right]^{1/2}.
$$

The tolerance on dose distributions can vary between different regions. Since the gradients are very high in the LGK, the dose drops to a tiny fraction of the center dose quickly. These lower doses are in general much less significant for the treatment. Therefore, errors in that low-dose region, denoted $d_{low}$, are not as important. For higher doses it is much more important that the calculations are precise. Above all, the mid-dose region $d_{mid}$, with the strongest gradients, need to be accurate, since this part of the dose will ideally be positioned perfectly on the tumour surface for the optimal outcome. With this reasoning, the high-dose region denoted $d_{high}$, containing the maximal dose $d_{max}$, is slightly less important. This is also in line with how the cost function in the optimization problem is formulated, since the high-dose region is always contained inside the target where we typically do not have an upper bound on dose. The exact dose intervals for every region are as follows.

**Definition 5.4.1. Dose regions**

- **Low:** $d_{low} = \{ d \in \mathbb{R}^+ : 0.05 \cdot d_{max} < d \leq 0.30 \cdot d_{max}\}$
- **Mid:** $d_{mid} = \{ d \in \mathbb{R}^+ : 0.30 \cdot d_{max} < d \leq 0.70 \cdot d_{max}\}$
- **High:** $d_{high} = \{ d \in \mathbb{R}^+ : 0.70 \cdot d_{max} < d \leq d_{max}\}$

The quotient between the $L_2$-error and the $L_2$-norm for MC, in the grid points contained in every zone, is illustrated as a function of the angles in Figure 5.17. The most significant misfit seems to be found in the low-dose region while the error seems to be kept relatively constant for both the mid and high-dose region over all the angles. One can argue that the $L_p$-norm is unfit for the task of comparing dose distributions. Small misalignment can generate huge errors, especially in the high gradient regions. To that it should be noted that the chosen collimator, 4 mm, has the highest gradients of the three. There are measures like the $\gamma$-index [14] that are better designed for the task, however that particular method constitutes an optimization problem on its own. We settle for the $L_2$ measure of error to limit the scope of this thesis.
Figure 5.17: $L_2$-error behaviour as a function of rotations.

Figure 5.17 indicates that the original $L_2$-error, i.e. for no rotation, is arguably relatively large, but that it does not grow significantly after rotation in the mid and high-dose regions. Some of the error might be attributed to differences in the segmentation of the head between the manually constructed MC test versus the LGP approach. The low-dose region fit seems to be more questionable, which might be attributed to that the statistical uncertainty is higher for lower doses in MC. The figure curiously suggest that the fit becomes better for -22.5 degrees for low and mid dose regions than in the original 0 degree case. This might indicate a difference in isocenter positions between LGP and MC or a slight error in transformation between coordinate systems. It must however be kept in mind that the manual methods as employed are prone to some small errors.
Chapter 6

Conclusions and further work

The potential gain from introducing rotational DOFs into the LGK inverse planner is limited to between 4.5% and 7% of the penalty cost function, depending on the case. It can practically be fully achieved by allowing the possibility to rotate -22.5, -11.25, 0 and 11.25 degrees around the central axis. In terms of clinical metrics, the improvements are most notably visible in GI and selectivity, often at the cost of BOT. Coverage on the other hand seems to be mostly unchanged. Experimentation with new “daring” isocenter placement, e.g. positions close to an OAR or the tumour surface, might make the rotational DOF more impactful. The results also indicate that the usage of the new DOFs is generally sparse, indicating that the majority of the improvements can be achieved with a limited increase in problem size. This might also open up for future exploration of the possibility to gradually remove obsolete DOFs to leave space for new more promising ones. Furthermore, the new DOFs might become more capable of improving the plans if the sampling of voxels were made with a higher emphasis on the surface of the target.

A large part of the gain from rotations can be attained with very low complexity dynamic models as proposed in this thesis. Empirically, the objective value in the dynamic model with only 2 angle nodes is approximately within one percent of the lower bound for all cases. It is possible to take advantage of otherwise untapped time by delivering a significant part of the dose while moving between positions. We conclude that there is a lot to gain by allowing even rudimentary shots while repositioning, suggesting that this method could give interesting results for other dynamic models delivering while travelling between isocenters. In addition, these primitive shots are cheaper than normal shots by a factor of 24, meaning that they are do not increase the problem size notably in comparison. Moreover, warm starting the optimization by using an old solution as a starting guess in simplex could significantly reduce the time to solve the problems. This could significantly reduce the computational load for the column generation approaches that rely heavily on being able to solve the optimization problems several times. Note however, that the program might have to be solved in the primal for the solutions to be feasible starting guesses, which could change the problem significantly.

Column generation shows much potential in the inverse planner. The non-uniform approach improves the model with less complex solutions, which leads to lower problem sizes. In general, an objective value within one percent of the lower bound is achieved with only one additional angle per isocenter. One can hypothesize that as a part of future work, this method can be combined with the approach of the dynamic model to possibly produce even smaller and less complex plans, that can capitalize on the benefits provided by rotations. Moreover, it would be interesting to attempt to solve the non-convex optimization problem stated in (3.11). Unfortunately, the problem is non-smooth and non-convex, but on the other hand, there is only one variable in the problem, which is constrained. One can argue that a local optima might suffice for the context, especially if several angles will be generated. Furthermore, recall that the isocenter positioning stage of the problem is done separately from the optimization, which is necessary for convexity, but might be limiting the plan quality that can be achieved. Column generation opens up the possibilities to incorporate the isocenter positioning into the optimization by employing a similar approach to
that of the non-uniform model. In this way one could gradually add new isocenters to the model via subprograms, which will presumably be more adapted to improve the objective function, and subsequently the treatment plan.

Trade-offs prioritizing memory can be made to significantly reduce the memory requirements for models with many DOFs. The memory usage can at least be reduced with 87% in the worst case, but extends the evaluation time of clinical metrics with up to 7% of the DRM calculation time for the worst scenario. In some cases it seems however that the total time is significantly reduced, but it must be stressed that the time frame in practice will likely be different. These results are also expected to benefit any types of DOF, not only rotational, and might make otherwise challenging additions much more tractable. For example, models with more isocenter positions or dynamic movement between isocenters instead of angles.

Traditional K-means clustering performs poorly in the LGK framework by being unable to deal with the high gradients and thus produces plans with lacking coverage. Methods emphasizing surface voxels on the other hand give promising results, and in general, provides similar or better plans than RSS. Moreover, an advantage is that the clustering results are "deterministic", albeit at the cost of a significant increase in computation time in comparison to RSS for the same size reduction. The clustering time seems to be very sensitive to the size of the DRM. One can furthermore experiment with using starting guesses and methods designed with more emphasis on distinguishing surface voxels. Consider representing the dose in a cluster with a weighted average of the voxels, with emphasis on the surface voxels. In a similar fashion one could consider a weighted norm in the K-means algorithm, instead of an ordinary $L^2$-norm. There might also be a lot to gain from investigating other cluster methods than K-means, that are more suitable for the common occurrence of steep gradients in the LGK.

It is concluded that the rotation invariance assumption is reasonable for the context of the optimization. The growth of the $L^2$-norm of the error between the approximation and the corresponding MC results is small enough to warrant the conclusion. For a more in depth analysis, it is suggested that the assumptions regarding both the rotation and translation invariance of the dose distributions should be tested simultaneously, since it is plausible that they aggravate each other. Moreover, it is suggested that the assumptions should be evaluated with other metrics, e.g., the $\gamma$-index that might be more appropriate for the context.
Bibliography


Appendices
Appendix A

Complementary mathematics

A more in-depth description of the central optimization problems employed in the testing of the methods presented in this thesis will be carried out in the following sections.

A.1 Standard form of the underlying model

Consider the piece-wise linear penalty functions from (2.12). This form can be achieved with the help of auxiliary variables $y^+, y^- \in \mathbb{R}^J$. Considering only $f_T(d)$, the fact that the objective function will be minimized in the end allows us to rewrite the penalties according to

$$\min \sum_{j \in T} \max \{D_T - d_j, 0\} = \left\{ \begin{array}{c} \min_{t, y^+, y^-} \ \sum_j y^-_j \\ s.t. \ \frac{d_j}{y^-_j} = D_T - \Phi_j t + y^-_j \\ y^+_j, y^-_j \geq 0 \end{array} \right\} \forall j \in T^*.$$

One can express $f_S$ and $f_L$ similarly. A similar trick can be carried out using auxiliary variables $\tau, r$ to express the BOT in a linear programming framework according to

$$\min \sum_{i=1}^I \max_s \sum_{c=1}^3 t_{isc} = \left\{ \begin{array}{c} \min_{t, \tau, r} \ \sum_i \tau_i \\ s.t. \ \tau_i - r_{is} = \sum_c t_{csi}, \ \forall s, i \\ \tau_i, r_{is} \geq 0 \end{array} \right\}.$$ 

Now, these constraints can be written $B\tau - Ir = Ct$ given that

$$C = \begin{bmatrix} I_8 & I_8 & I_8 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & I_8 & I_8 & I_8 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & I_8 & I_8 & I_8 \end{bmatrix}, \quad B = \begin{bmatrix} 1_{8 \times 1} & 0 & 0 & \cdots & 0 \\ 0 & 1_{8 \times 1} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1_{8 \times 1} \end{bmatrix}.$$

Where $I$ is the identity matrix and $1$ the matrix of ones. The full linear problem from (2.14) can then be expressed.
Now, divide the DRM into one sub-matrix per structure such that, i.e., \( \Phi_T \) problem can then be written

\[
\min \frac{\omega_T}{D_T T} \sum_{j \in T} y_j^+ + \frac{\omega_S}{D_S S} \sum_{j \in S} y_j^+ + \frac{\omega_L}{D_L L} \sum_{j \in L} y_j^+ + \frac{\omega_R \cdot \tilde{\varphi}_{cal}}{D_T I} \sum_i \tau_i
\]

subject to

\[
\begin{align*}
 y_j^+ &= D_T - \Phi_j t + y_j^-, \quad \forall j \in T^* \\
y_j^+ &= D_S - \Phi_j t + y_j^-, \quad \forall j \in S^* \\
y_j^+ &= D_L - \Phi_j t + y_j^-, \quad \forall j \in L^* \\
\Phi_j t - s_j &= D_O, \quad \forall j \in O \\
\tau_i - r_i &= \sum_c l_{cal}, \quad \forall s, i \\
t, y^+, y^-, \tau, r, s &\geq 0.
\end{align*}
\]

Now, divide the DRM into one sub-matrix per structure such that, i.e., \( \Phi_T \) \( \in \mathbb{R}^{T \times F} \) is a matrix with only the rows corresponding to voxels \( j \in T^* \). The resulting components in the standard form expressed in (2.4) thus becomes

\[
\begin{align*}
 c^T &= \begin{bmatrix} 0 & 0 & \frac{\omega_T}{D_T T} & \frac{\omega_L}{D_L L} & 0 & \frac{\omega_S}{D_S S} & 0 & 0 & 0 & \frac{\omega_R \cdot \tilde{\varphi}_{cal}}{D_T I} \end{bmatrix} \\
x^T &= \begin{bmatrix} t & y_T^+ & y_T^- & y_S^+ & y_S^- & s & r & \tau \end{bmatrix} \\
A &= \begin{bmatrix} \Phi_T & -I & I & 0 & 0 & 0 & 0 & 0 & 0 \\
\Phi_L & 0 & 0 & -I & I & 0 & 0 & 0 & 0 \\
\Phi_S & 0 & 0 & 0 & 0 & -I & I & 0 & 0 \\
C & 0 & 0 & 0 & 0 & 0 & 0 & I & -B \end{bmatrix}, \quad b = \begin{bmatrix} D_T \\
D_L \\
D_S \\
D_O \end{bmatrix}.
\end{align*}
\]

A.2 The dual formulation

Consider the dual formulation in (2.5). Note that it is fully defined by (A.1), and that for the constraints in the dual there are several rows with only one single variable. This leads to that a large number of rows can be removed and replaced by simple bounds according to

\[
\begin{bmatrix}
-I & 0 & 0 & 0 & 0 \\
I & 0 & 0 & 0 & 0 \\
0 & -I & 0 & 0 & 0 \\
0 & I & 0 & 0 & 0 \\
0 & 0 & -I & 0 & 0 \\
0 & 0 & I & 0 & 0 \\
0 & 0 & 0 & I & 0 \\
0 & 0 & 0 & 0 & I \\
\end{bmatrix} \lambda \leq \begin{bmatrix}
0 \\
\frac{\omega_T}{D_T T} \\
\frac{\omega_L}{D_L L} \\
\frac{\omega_S}{D_S S} \\
\frac{\omega_R \cdot \tilde{\varphi}_{cal}}{D_T I} \\
0 \\
0 \\
0 \\
0 \\
\end{bmatrix} \rightarrow \begin{cases}
0 \leq \lambda_T \leq \frac{\omega_T}{D_T T} \\
0 \leq \lambda_L \leq \frac{\omega_L}{D_L L} \\
0 \leq \lambda_S \leq \frac{\omega_S}{D_S S} \\
0 \leq \lambda_O \leq \infty \\
0 \leq \lambda_B \leq \infty
\end{cases}.
\]

Further, note that many constants can be canceled out by a change of notation to

\[
\begin{align*}
\gamma_T, \gamma_L, \gamma_S, \gamma_O, \gamma_B &:= \begin{bmatrix} D_T \lambda_T, -D_L \lambda_L, -D_S \lambda_S, -D_O \lambda_O, -\frac{D_T I \cdot \tilde{\varphi}_{cal} \lambda_B}{} \end{bmatrix}, \\
\Psi_T, \Psi_L, \Psi_S, \Psi_O, C_s &:= \begin{bmatrix} D_T \Phi_T, -D_L \Phi_L, -D_S \Phi_S, -D_O \Phi_O, -\frac{D_T I \cdot \tilde{\varphi}_{cal} C}{} \end{bmatrix}.
\end{align*}
\]

Where \( \gamma \) is the new dual variable, and \( \Psi \) the re-scaled DRM matrices. Now, the full re-scaled dual problem can then be written
$$\max_\gamma \begin{bmatrix} 1 & -1 & -1 & -1 & 0 \end{bmatrix} \cdot \gamma$$

subject to

$$\begin{bmatrix} \Psi_T^T & -\Psi_L^T & -\Psi_S^T & -\Psi_O^T & -C_S^T \end{bmatrix} \cdot \gamma \geq \begin{bmatrix} 0 \end{bmatrix}$$

(A.2)

A.2.1 Implications of freezing objectives

Given that (2.14) has been solved, a Pareto optimal solution \((f_T, f_S, f_L, f_B)\) is available. Assume that it is desired to apply constraints mentioned in (3.13) for both \(f_T\) and \(f_L\) simultaneously with constants \(V_T := f_T\) and \(V_L := f_L\). These constraints can be written in vector form according to

$$\frac{\omega_T}{D_T} 1^T y_T = V_T \quad \text{and} \quad \frac{\omega_L}{D_L} 1^T y_L = V_L.$$ 

The standard form formulation is identical to (A.1) except for

$$A = \begin{bmatrix} \Phi_T & -I & I & 0 & 0 & 0 & 0 & 0 & 0 & 0 \ 
\Phi_L & 0 & 0 & -I & I & 0 & 0 & 0 & 0 & 0 \\
\Phi_S & 0 & 0 & 0 & 0 & -I & I & 0 & 0 & 0 \\
\Phi_O & 0 & 0 & 0 & 0 & 0 & 0 & I & 0 & 0 \\
C & 0 & 0 & 0 & 0 & 0 & 0 & 0 & I & -B \\
0 & 0 & 1^T & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1^T & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad b = \begin{bmatrix} D_T \\
D_L \\
D_S \\
D_O \\
0 \\
D_T V_T \\
D_L V_L \end{bmatrix}.$$ 

The result is a new dual variable for each objective frozen, along with a number of constraints in (A.2). After a number of operations similar to in the previous section, the resulting new dual problem becomes

$$\max_\gamma \begin{bmatrix} 1 & -1 & -1 & -1 & 0 \end{bmatrix} \cdot \gamma$$

subject to

$$\begin{bmatrix} \Psi_T^T & -\Psi_L^T & -\Psi_S^T & -\Psi_O^T & -C_S^T \end{bmatrix} \cdot \gamma \geq \begin{bmatrix} 0 \end{bmatrix}$$

Here, the new additions are colored in blue. Note that big additions are introduced in the dual matrix with these constraints, resulting in increased RAM usage and solving times. The new appended matrices are very sparse fortunately.

A.2.2 The clustered model

The clustering problem looks almost identical except for the fact that the values in the rows of the matrices, e.g. \(\hat{\Psi}_T\) and \(\hat{\gamma}\), represent clusters instead of voxels and let the vectors \(C_T, C_L, C_S\) contain the number of member voxels per cluster in their corresponding structure. The new optimization problem is given by
A.2.3 The dynamic model

In the dynamic model, the new primitive DOF $\tilde{\Psi}$ can be appended to the matrix directly due to the BOT being equal to the delivery time, per definition of primitive shots. The dual form becomes

$$\begin{align*}
\max_{\gamma} \quad & \begin{bmatrix} 1 & -1 & -1 & -1 & 0 \end{bmatrix} \cdot \gamma \\
\text{s.t.} \quad & \begin{bmatrix} \Psi^T_T & -\Psi^T_L & -\Psi^T_S & -\Psi^T_O & -C^TS^T \end{bmatrix} \cdot \gamma \geq \begin{bmatrix} 0 \\
\end{bmatrix} \\
& \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \end{bmatrix} \leq \gamma \leq \begin{bmatrix} \infty \omega_T \infty \omega_S \infty \omega_L \infty \infty \end{bmatrix}.
\end{align*}$$
Appendix B

Additional results

B.1 Distribution of beam-on time

The result from a second iteration of the non-uniform model is illustrated in Figures B.1 and B.2. It is reasonable to propose that if any significant angle is not identified by the first iteration, it is generally found in the second iteration. This seems to be in accordance with the results in Figures 5.11 and 5.10, where it is evident that the vast majority of the potential, in terms of objective value, is achieved after only two or three iterations for the non-uniform case.

Figure B.1: The distribution of BOT resulting from the uniform model for case SBR39 (left) and for SBR61 (right). The angles proposed by the non-uniform model is marked for one (red) and two iterations (green).

Recall the assumption of disjoint isocenters in Section 4.4. For the bigger models there are many unused isocenters, meaning that no BOT is spent in any of its angles. It becomes evident that the assumption does not quite hold in these cases, the algorithm is forced to propose angles in isocenters that are insignificant for the final plan. The algorithm produces the columns with the best reduced cost in the isocenter, but it cannot ensure that it will necessarily be competitive to the other DOFs.

Figure B.2: The distribution of BOT resulting from the uniform model for case GFR46 (left) and for SBR41 (right). The angles proposed by the non-uniform model is marked for one (red) and two iterations (green).