Optimization models and techniques for radiation treatment planning applied to Leksell Gamma Knife® Perfexion™

by

Hamid R. Ghaffari

A thesis submitted in conformity with the requirements for the degree of Doctor of Philosophy
Graduate Department of Mechanical and Industrial Engineering
University of Toronto

Copyright © 2012 by Hamid R. Ghaffari
Abstract

Optimization models and techniques for radiation treatment planning applied to Leksell Gamma Knife® Perfexion™

Hamid R. Ghaffari
Doctor of Philosophy
Graduate Department of Mechanical and Industrial Engineering
University of Toronto
2012

Radiation treatment planning is a process through which a certain plan is devised in order to irradiate tumors or lesions to a prescribed dose without posing surrounding organs to the risk of receiving radiation. A plan comprises a series of shots at different positions with different shapes. The inverse planning approach which we propose utilizes certain optimization techniques and builds mathematical models to come up with the right location and shape, for each shot, automating the whole process. The models which we developed for Perfexion™ unit (Elekta, Stockholm, Sweden), in essence, have come to the assistance of oncologists in automatically locating isocentres and defining sector durations. Sector duration optimization (SDO) and sector duration and isocentre location optimization (SDIO) are the two classes of these models. The SDO models, which are, in fact, variations of equivalent uniform dose optimization model, are solved by two nonlinear optimization techniques, namely Gradient Projection and our home-developed Interior Point Constraint Generation. In order to solve SDIO model, a commercial optimization solver has been employed. This study undertakes to solve the isocentre selection and sector duration optimization. Moreover, inverse planning is evaluated, using clinical data, throughout the study. The results show that automated inverse planning contributes to the quality of radiation treatment planning in an unprecedentedly optimal fashion, and significantly reduces computation time and treatment time.
Dedication

I dedicate this thesis to my lovely wife, Behnaz Tat, whom I owe every step of the way toward this achievement. I also dedicate thesis to my wonderful daughter, Kimia, and son, Kasra, who patiently supported me through the way.
Acknowledgements

I would like to express my deepest appreciation and thanks to my supervisor and wonderful mentor, Dionne M. Aleman, who patiently guided me throughout the research with her insightful ideas and directions.

I also want to thank my committee members: Roy H. Kwon, Timothy C.Y. Chan, David A. Jaffray, Michael Sharpe, and Gino J. Lim for reviewing my thesis and providing great suggestions and comments. I am sincerely thankful to Professor Tamás Terlaky and Professor Michael W. Carter who kindly answered my questions along the way.

I am also grateful to my friend and collaborator Mohammad R. Oskoorouchi, Kimia Ghabadi, and Mark Ruschin for their support and help in preparing papers and publications.

Finally and most importantly, my utmost appreciation to my wife, Behnaz Tat, for her continuous support, constant love, non-stoping sacrifices, and stable understanding without which none of this would have happened.
Contents

1 Introduction and background ........................................... 1
   1.1 Leksell Gamma Knife® Perfexion™ treatment planning .......... 3
   1.2 Semi-infinite linear optimization ................................ 7
   1.3 Treatment plan evaluation ....................................... 9
   1.4 Contributions .................................................. 12

2 Sector duration optimization with pre-defined isocentres .......... 17
   2.1 Introduction .................................................. 17
   2.2 Sector-duration optimization models ............................ 17
   2.3 Sector-duration optimization solution methods .................. 20
      2.3.1 Gradient projection ....................................... 20
      2.3.2 Interior-point constraint generation ...................... 20
      2.3.3 Commercial solver MOSEK ................................ 23
   2.4 Results ....................................................... 23
      2.4.1 Numerical results ......................................... 23
      2.4.2 Clinical results ........................................... 29
   2.5 Conclusions .................................................... 33

3 Simultaneous sector duration and isocentre optimization ......... 36
   3.1 Introduction .................................................. 36
   3.2 A tractable mixed-integer model for SDIO ........................ 38
List of Tables

1.1 Clinical cases’ information ........................................... 10

2.1 Plan quality summary of GP ........................................... 31

3.1 Effect of limiting the number of isocentres in SDIO ................. 41
3.2 Plan quality comparison of isocentre limit scenarios ............... 42
3.3 Plan quality summary in the no-bounds scenario .................... 45
3.4 Plan quality summary in the loose-bounds scenario ................ 46
3.5 Plan quality summary with tight-bounds scenario .................. 48

4.1 Effect of changing $t_{\text{MIN}}$ on the quality of the plans .......... 59
4.2 BOT reduction in SDIO with no penalty on number of isocentres ... 61

5.1 Computational results of interior point constraint generation .... 82
List of Figures

1.1 Leksell Gamma Knife® and Perfexion™ units ................................................. 2
1.2 CT image for Case 1 .................................................................................. 11
2.1 SDO objective function ........................................................................ 19
2.2 Dose-volume histogram and isodose lines for Case 1 using SDO .......................... 25
2.3 Treatment plan quality by the GP algorithm ........................................... 27
2.4 Treatment plan quality by the IPCG algorithm ......................................... 28
2.5 Treatment plan quality by MOSEK ............................................................. 30
2.6 Treatment plan metrics for Case 1 ............................................................. 32
3.1 Dose-volume histogram and isodose lines for Case 1 using SDIO .................. 44
3.2 Voxel sampling rate effects on treatment quality ........................................ 49
4.1 Visualization of BOT penalty functions in SDO-BOT ............................... 55
4.2 Beam-on time penalty term comparison ................................................. 58
5.1 The IPCG algorithm visualization .............................................................. 72
5.2 Convergence behaviour of IPCG ............................................................. 85
5.3 Gradient projection convergence ............................................................. 86
A.1 Sub-gradient based oracle ....................................................................... 95
Chapter 1

Introduction and background

Nearly two-thirds of all cancer patients in the U.S. will receive radiation therapy during their illness [18]. Head and neck cancer accounts for about 3% to 5% of all cancers in the United States [18]. In 2012, an estimated 52,610 people will develop head and neck cancer, and an estimated 11,500 deaths will occur [18]. In 1968, Elekta (Stockholm, Sweden) developed Leksell Gamma Knife® (LGK), a device exclusively for minimally invasive radiosurgery of tumors and lesions in the head and neck. LGK surgery alone has been used on more than half a million people worldwide, and is used to treat 50,000 patients every year [24].

LGK (Figure 1.1a) consists of 201 radioactive sources of Cobalt-60 arrayed in a hemisphere with central channels through which gamma rays are focused to a common focal point called an isocentre. The LGK is provided with a set of four collimator helmets, providing collimator sizes of 4, 8, 14, and 18 mm (Figure 1.1b). A collimator size indicates the diameter of the radiation beams. A simultaneous collection of beams pointed at a particular isocentre is called a shot. Several shots are placed inside the target region at different isocentres, and sufficient radiation dose in those shots treats the lesions. The helmets must be manually changed to deliver the desired shot size throughout the delivery of the treatment plan.
Figure 1.1: LGK (a) collimator helmets (b) must be manually changed according to the required shot sizes. In PFX (c), collimators (d) moved automatically and independently to desired sizes for each of eight sectors. (www.elekta.com)
In the latest generation of LGK, Perfexion™ (PFX), automated multi-source collimators with 192 sources are placed on eight moveable sectors (Figure 1.1c). Each sector collimator can automatically move into four positions in order to generate beam rays of diameter 4, 8, or 16 mm, or the beam can be blocked (Figure 1.1d).

PFX also provides automated couch movements and beam diameter changes, and is capable of delivering different beam sizes in different sectors. This new feature helps to lessen the extensive labor time of changing collimators for size-wise precision as well as adding a wide range of possible shot shapes instead of having only spherical ones. The technological advances in PFX make it possible to access any point from the top of the head to the cervical spine neck.

With PFX, the treatment of more complex and larger targets is now possible due to the reduction in manual effort and treatment time through machine movement automation. However, the major obstacle that trained treatment planners face is the presence of too many variables, which makes manual planning challenging. The following sections discuss these challenges and steps we take to overcome each of them.

1.1 Leksell Gamma Knife® Perfexion™ treatment planning

The neurosurgeons and the radiation oncologists plan the treatment by determining isocentres, the shot shapes, and the dose intensity (radiation duration at an isocentre). We refer to this duration as sector duration. In radiation treatment planning, a patient’s body is approximated as many cubes of tissue, called voxels (volumetric pixels). Voxels are typically about 1mm×1mm×1mm or smaller, depending on the image resolution. The desired amount of dose delivered to each voxel is determined by clinicians. The method to calculate the dose received to each voxel in a shot, called a dose engine, is provided by the manufacturer.
What makes the treatment planning challenging to the planners is the presence of many variables. These variables include the number and position of shots, collimator size for each of eight sectors per isocentre, and beam duration of each collimator size for each sector of each shot. The optimal solution for a given treatment is unlikely to be found manually. However, clinically satisfactory plans can be obtained through manual planning, called \textit{forward planning}, though the quality of the forward plans may vary between planners. Optimization techniques provide mathematical tools to obtain the optimal plan in reasonable time. Having an automated practical procedure to find the optimal plan helps treatment planers to achieve a standard level of quality.

The use of optimization techniques in radiation treatment planning has been an open and exciting arena of on-going research in the field. Intensive work done on \textit{intensity modulated radiation therapy} (IMRT) is a case in point (for example, see \cite{4, 5, 20, 47, 57, 75}). On the grounds of the wide range of applications of IMRT to treat tumors in different parts of the human body, it is no surprise that there has been significant bulk of work in this area. Similar approaches can be applied to automate treatment planning with LGK.

\cite{Ferris et al.} developed a computer program to reduce the threat of human error in setting radiation treatment plans \cite{1, 30, 48, 67}. They later experimented with two different approaches in modelling the problem: nonlinear programming and mixed integer programming (MIP) \cite{29}. They also modelled the dose distribution function for each shot width using the error function which forced them to solve their model many times to reduce the approximation error. To incorporate shot location, the authors introduced a mixed integer programming model by assigning a binary variable to each shot at each isocenter with only one collimator size. Therefore, the proposed MIP model needs an enormous number binary variables as well as huge amount of data which increases the complexity of the model exponentially. This data requirement ultimately resulted in the MIP model not being used in the authors’ treatment planning software. Also, the
authors initially started with spherical shot shapes and extended them to ellipsoidal ones; however, the LGK can deliver much more complex and irregular shot shapes—even though it is extremely labor intensive to create such a shape in LGK. The assumption of ellipsoidal shapes allowed the authors to find a good estimation of the isocentre locations by using the well-known ball packing problem. With the new flexibility of PFX, the need to investigate new large-scale optimization models and techniques to solve them efficiently is critical.

Many existing optimization models suffer from hardware limitations at delivering complex clinical treatments through earlier LGK models. For instance, changing the helmet from one size to another is a very time-consuming process. Therefore, implementing treatment plans with large numbers of shots on the LGK unit is practically impossible. Another limitation with LGK is the limited number of shot shapes. Since each of the 201 helmet holes must be plugged manually to avoid the curtain area, changing the shot shape is highly labor intensive. Therefore, the treatment plan team tends to limit the shots to an approximation of a sphere. These limitations have led the researchers to end up with very large optimization problems (mainly MIPs) which are computationally expensive to solve. As a result, generating plans with large numbers of shots (more than 20) was again practically hard to achieve.

Although all the previous studies provided interesting models and approaches to solve the LGK treatment planning problem, these works cannot be entirely applied to the new PFX machine because the PFX units afford more flexibility in the treatment. In fact, the PFX treatment planning problem can be modelled in a way very similar to intensity modulated radiation therapy treatment planning problems \[4, 5, 20, 21, 47, 52, 57, 75\]. In particular, the approaches taken in \[4, 21, 52\] are attractive due to the convexity of the formulations.

Similar to the beam-orientation optimization problem in IMRT, *isocentre optimization* (IO), is to find the optimal isocentre locations regarding the geometry of target and
adjacent structures. The problem is known to be NP-hard. A fast and clever approach to find a good set of isocentre such as skeletonization \cite{21, 28} or grassfire-sphere packing \cite{33} can be beneficial to the process of finding the optimal shot shapes or sector durations in PFX. Analogous to the fluence map optimization in IMRT, the purpose of the sector duration optimization (SDO), given a certain set of isocentres, is to achieve the prescribed target dose and healthy tissues sparing. Although we can use our proposed methods to solve the SDO problem, optimally locating the isocentres is not trivial. Physicians traditionally rely on their experience and judgment in determining the isocentres, which may lead to a non-optimal treatment plan. In this work, we focus on methods to model and solve the SDO problem.

The proposed SDO models and optimization techniques are evaluated by generating treatment plans for clinical cases. The details of our seven cases are presented in Section 1.3. The SDO models are tested for both radiosurgery and radiotherapy treatment scenarios following the clinical guidelines presented in Section 1.3. A gradient-projection algorithm, presented in Chapter 2, is first used to solve our test problems. To improve the computation time to solve such large-scale SDO problems, an interior-point constraint generation, presented in Chapter 5, is developed and implemented and generate both radiosurgery and radiotherapy treatment plans.

We also developed a new approach that automatically generates complete stereotactic radiosurgery treatment plans for Leksell Gamma Knife\textsuperscript{®} Perfexion\textsuperscript{™} that combine sector duration optimization and isocentre optimization (SDIO) in one large mixed-integer linear model. A voxel sampling approach is used to reduce the size of the model so that it can be solved in reasonable time. We also propose a heuristic method of bounding the number of isocentres needed for each target in a treatment plan in SDIO. The results show that each of our methods obtains quality treatment plans with a predictable amount of computational effort.

One of the components which effects the quality of a plan is the beam-on time (BOT)
or the time that the patient is under radiation. There are a few works done on reducing beam-on time [3, 10, 13] in IMRT but none for Perfexion™ or Leksell Gamma Knife®. Unlike in [3, 10, 13] where reducing the beam-on time is done through the leaf sequencing step, we developed an extension to the SDO model to incorporate the beam-on time directly into the optimization model.

1.2 Semi-infinite linear optimization

In order to solve the SDO problem, we develop an interior point constraint generation algorithm that can solve semi-infinite linear optimization problems, which can be used to describe SDO. Semi-infinite linear optimization (or programming) (SILO) is a well established mathematical concepts in literature. The theoretical aspects of SILO or, in general, infinite-programming, can be found in [7, 8, 36].

Roughly speaking, SILO deals with an optimization problem with a linear objective and linear constraints in which either the number of constraints or the dimension of the variable space, one at a time, is allowed to be infinite. The SILO class of problems essentially contains convex optimization; in particular, semi-definite optimization and second-order cone optimization (SOCO). Due to the complexity of the existing optimization algorithms [55], there have been few attempts to implement them in practice. Exponential algorithms have been developed to solve SILO problems, however, due to the complexity of the methods, no efficient implementation is available for the public. In this study, we develop and implement an efficient algorithm to solve SILO problems.

There are many algorithms in the literature based on cutting plane methods for SILO (see, for instance, [27, 49, 55, 79]). However, due to the complexity of these methods, no implementation of them made its way to the public. The method that we describe in this work is a variant of [55] with major differences both from the theoretical and implementation viewpoints. There are three main theoretical enhancements. First, our
algorithm adds violated constraints with no changes to the right hand side. In [55], when a violated constraint is identified, it is relaxed by changing its right hand side to make the current solution strictly feasible, which of course results in loss of information. We keep the violated constraints as deep as they are. Second, we extend the analysis to the case where multiple violated constraints are added simultaneously instead of adding one constraint at a time. Finally, at each iteration we update the barrier parameter together with updating the feasible region in the same step. All of these modifications contribute to the efficiency of the method as documented in Chapter 5.

Although there exist many efficient software packages based on polynomial interior-point methods for convex conic optimization (such as [72], SDPT3 [70], SeDuMi [66, 69], and CSDP [16]), and based on low-rank factorization (such as SDPLR [17]), we would still like to keep SOCO problems within our domain as we develop this algorithm. While today’s software packages perform extremely well on small to moderate size convex conic problems, they cannot efficiently handle large-scale problems arising in various real-life applications. For example, an optimization problem with a few thousand conic constraints of large size, say $10^4$, especially when dense, would be a challenging problem for classical primal-dual interior-point methods, and thus would require significant computation time to solve even by today’s state-of-the-art software packages.

We develop an interior-point constraint generation (IPCG) algorithm that can solve convex optimization problems efficiently, including large-scale SOCO. The IPCG algorithm, described in Chapter 5 is implemented to solve a SILO reformulation of SDO in Perfexion™ treatment planning. The results show that although IPCG has an exponential complexity, it converges comparably with the polynomial algorithms implemented in the literature.
1.3 Treatment plan evaluation

PFX can be used to deliver two types of radiation therapy: radiosurgery and radiotherapy. In stereotactic radiosurgery, the goal of the treatment is to deliver a very high dose to a target structure called the gross tumor volume (GTV). The clinical target volume (CTV), if any, is treated as GTV in our optimization models. The GTV can receive up to twice the prescribed dose, which allows some dose heterogeneity throughout the target. The dose distribution should be tightly contoured around the target structure so that surrounding healthy organs, called organs-at-risk (OARs), may not receive a high dose. A planning target volume (PTV) is usually the GTV plus some margin prescribed as therapeutic radiation dose. In radiosurgery treatments, due to the high dose intensity, the need for very steep dose fall-off outside the target structures is essential.

In radiotherapy, on the other hand, homogeneity of the target dose is of essence. Therefore, the PTV, which is naturally expected to contain healthy and tumorous tissues combined, replaces the GTV, which contains tumorous tissues only. Due to the fact that the healthy tissues recover faster than tumorous ones, in radiotherapy paradigm, a moderate and uniform dose in certain intervals is prescribed to the PTV give the healthy tissues a respite from radiation for them to recover after an early exposure.

Seven clinical cases provided by our collaborators from Princess Margaret Hospital (Toronto, ON) are used to evaluate the models and techniques developed throughout this study. The clinical characteristics of the cases are presented in Table 1.1. Each of these test cases represents a typical challenge that treatment planners face. For instance, Case 1 contains a very large tumor next to the brainstem (Figure 1.2) which highly affects the conformity of the radiosurgery treatment plans, while Case 2 has two large metastases as targets.
Table 1.1: Case information. AN=acoustic neuroma. MM=multiple metastases. BS=brainstem. Ch=Chiasm. Cl=Cochlea. Cnv=Cranial nerve V.

<table>
<thead>
<tr>
<th>Case</th>
<th>Indication</th>
<th>GTV volume (cm³)</th>
<th>GTV number of voxels</th>
<th>PTV volume (cm³)</th>
<th>PTV number of voxels</th>
<th>PTV Overlap %</th>
<th>Total number of voxels</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>AN</td>
<td>8.56</td>
<td>7,178</td>
<td>18.51</td>
<td>15,535</td>
<td>10.2</td>
<td>–</td>
</tr>
<tr>
<td>2a</td>
<td>MM</td>
<td>17.72</td>
<td>34,763</td>
<td>25.87</td>
<td>50,756</td>
<td>0</td>
<td>–</td>
</tr>
<tr>
<td>2b</td>
<td></td>
<td>11.71</td>
<td>22,973</td>
<td>17.57</td>
<td>34,467</td>
<td>0</td>
<td>–</td>
</tr>
<tr>
<td>3</td>
<td>AN</td>
<td>1.28</td>
<td>3,788</td>
<td>2.16</td>
<td>6,394</td>
<td>4.9</td>
<td>–</td>
</tr>
<tr>
<td>4a</td>
<td>MM</td>
<td>0.85</td>
<td>2,058</td>
<td>1.80</td>
<td>4,345</td>
<td>0</td>
<td>–</td>
</tr>
<tr>
<td>4b</td>
<td>MM</td>
<td>25.81</td>
<td>62,241</td>
<td>34.36</td>
<td>82,849</td>
<td>0</td>
<td>–</td>
</tr>
<tr>
<td>4c</td>
<td></td>
<td>5.66</td>
<td>13,637</td>
<td>8.75</td>
<td>21,108</td>
<td>0</td>
<td>–</td>
</tr>
<tr>
<td>5</td>
<td>AN</td>
<td>5.08</td>
<td>5,037</td>
<td>12.33</td>
<td>12,226</td>
<td>12.8</td>
<td>0.1</td>
</tr>
<tr>
<td>6</td>
<td>AN</td>
<td>13.06</td>
<td>13,159</td>
<td>56.49</td>
<td>23,693</td>
<td>10.9</td>
<td>–</td>
</tr>
<tr>
<td>7a</td>
<td>MM</td>
<td>0.19</td>
<td>328</td>
<td>0.50</td>
<td>860</td>
<td>37.7</td>
<td>–</td>
</tr>
<tr>
<td>7b</td>
<td>MM</td>
<td>2.71</td>
<td>4,617</td>
<td>4.37</td>
<td>7,64</td>
<td>23.0</td>
<td>–</td>
</tr>
<tr>
<td>Mean</td>
<td></td>
<td>–</td>
<td>8.42</td>
<td>15,434</td>
<td>16.61</td>
<td>12.4</td>
<td>0.1</td>
</tr>
<tr>
<td>St. Dev.</td>
<td></td>
<td>–</td>
<td>8.05</td>
<td>18,617</td>
<td>17.06</td>
<td>12.7</td>
<td>0.1</td>
</tr>
</tbody>
</table>
Radiosurgery clinical objectives

The following institutional clinical objectives for PFX radiosurgery treatment planning were considered: (1) 100% of prescription dose (Rx) or higher must be received by 98% of the target structure ($V_{100} \geq 98\%$); (2) the maximum dose to $1\text{mm}^3$ of brainstem must not exceed 15Gy; (3) the maximum dose to optic structures must not exceed 8Gy to $1\text{mm}^3$; and (4) the Classic conformity index ($\text{CI}_{\text{Classic}}$) should be less than 1.5. Additional treatment plan quality metrics including the Paddick conformity index ($\text{CI}_{\text{Paddick}}$) [63] and beam-on time, radiation time, were also recorded. These conformity indices are calculated using the following formulas:

$$
\text{CI}_{\text{Classic}} = \frac{\text{PIV}}{\text{TV}}
$$

$$
\text{CI}_{\text{Paddick}} = \frac{\text{TV}_{\text{PIV}}}{\text{TV}} \times \frac{\text{TV}_{\text{PIV}}}{\text{PIV}}
$$

where PIV (prescription isodose volume) is the treated volume, that is, the volume enclosed by a given isodose line (e.g., 50%, 95%, etc.), TV is the target volume, and $\text{TV}_{\text{PIV}}$ is the target volume covered by prescription isodose volume. For PIV, we use 100% isodose lines as the prescribe dose.
Due to a limitation imposed in the machine design, for SDO treatment plans, any shots with duration less than 10s were removed from the treatment plan for clinical deliverability. For our typical cases, the results show that removing shots with less than 10s duration does not effect the quality of the plans significantly. To ensure target dose coverage of $V_{100} \geq 98\%$, a simple heuristic scaling procedure was developed and used as a post process in our inverse planning approach. Such normalization is common in forward planning, and allows for more fair comparisons between generated plans.

**Radiotherapy clinical objectives**

Unlike radiosurgery plans, radiotherapy plans require much higher target prescription doses (to be delivered over a pre-determined number of fractions), as well as homogeneity of target dose, whereas radiosurgery treatments are primarily concerned with conformity. Radiotherapy plans also include a PTV in addition to the GTV, resulting in a larger overall treatment volume. Further, in radiosurgery, a common target prescription dose is 12Gy with a maximum allowable dose of 15Gy to the brainstem (a 25% difference in dose), while in radiotherapy, a common GTV and PTV prescription dose is 50Gy with a maximum allowable dose of 54Gy to the brainstem (an 8% difference in dose, requiring high conformity as well as homogeneity throughout the PTV). This smaller difference is additionally complicated by the frequent overlap of the brainstem and PTV.

### 1.4 Contributions

We developed the first optimization models to obtain Perfexion™ treatments, and provided several variations. We also developed an interior-point constraint generation algorithm to solve large-scale convex problems which outperforms MOSEK, a commercial non-linear optimization solver. The developed IPCG algorithm can handle many more isocentres in SDO than MOSEK or gradient projection [62]. We developed the first ex-
licit beam-on time consideration in external beam radiation therapy optimization. Each of our methods yields quality treatment plans in clinically acceptable computation time. The developed inverse planning approach improved treatment quality over manual planning in most cases. Our computerized treatment planning procedure has the potential to improve standardization in treatment planning.

The following contributions are made to the literature either as book chapters, journal publications, or conference presentations.

**Book Chapters**


**Peer-reviewed journals**


Peer-reviewed conference proceedings


Conferences presentations


Other presentations


Chapter 2

Sector duration optimization with pre-defined isocentres

2.1 Introduction

The radiation delivery at a given isocentre location is determined by the amount of time each sector delivers radiation at each of the three available collimator sizes (4mm, 8mm, or 16mm). Beam weights resulting in zero duration is equivalent to a blocked sector, that is, no radiation delivered from that sector. Our model is similar to the fluence map optimization (FMO) problem in IMRT, where, for a fixed set of beams, the durations (intensities) of each beamlet in each beam must be optimized. In our model, we use a convex penalty function approach based on the FMO models presented previously for IMRT [4, 5, 6, 20, 47, 57, 75], with increased flexibility in the penalty functions.

2.2 Sector-duration optimization models

Let $\Theta$ be the set of isocentre locations determined by the grassfire and sphere-packing (GSP) algorithm presented in [33]. Let $\mathcal{B}$ be the set of eight sectors of the Perfexion$^{\text{TM}}$ machine and $\mathcal{C}$ be the set of collimator sizes. Note that, although the order of shots and
sectors and collimator size selection is important in efficiency of the treatment plan, in this study, we assume $\Theta$, $\mathcal{B}$ and $\mathcal{C}$ are not ordered sets. Let $t_{Ibc}$ represent the duration of radiation delivery to isocentre location $I \in \Theta$ from sector $b \in \mathcal{B}$ at collimator size $c \in \mathcal{C}$. To optimize the time of delivery for each combination of sectors and collimators (a total of 24 combinations per isocentre location), a method to calculate dose as a function of $t_{Ibc}$ is required. Consider a set of structures $\mathcal{S}$ in the treatment plan. Typically, this set consists of all target structures, a ring around each target as a healthy structure to improve conformity, and all or a subset of OARs. Each structure $s \in \mathcal{S}$ has $v_s$ voxels, and each voxel $j$ in structure $s$ receives dose $z_{js}$. Dose is defined as

$$z_{js} = \sum_{I \in \Theta} \sum_{b \in \mathcal{B}} \sum_{c \in \mathcal{C}} D_{Ibcjs} t_{Ibc} \quad \forall j = 1, \ldots, v_s, \ s \in \mathcal{S} \quad (2.1)$$

where $D_{Ibcjs}$ is the dose deposited per unit time (i.e., dose rate) at isocentre $I$ by sector $b$ at size $c$ to voxel $j$ in structure $s$. The $D_{Ibcjs}$ distribution is obtained by a dose engine implemented in MATLAB (by the vendor, Elekta), which models the clinical treatment planning dose algorithm (TMR1(1)).

In our penalty based model, each voxel is assigned a penalty related to the amount of under- or over-dosage it receives. The penalties are weighted according to the structure to which the voxel belongs so that some structures can be given priority over other structures. The penalties for underdosing may be different from the penalties for overdosing so that the optimization model has a preference for certain structure dose.

The penalty function for the dose received by voxel $j$ in structure $s$ is defined as

$$F_s(z_{js}) = \frac{1}{v_s} \left[ w_s (z_{js} - T_s)_+^2 + w_s (T_s - z_{js})_+^2 \right], \quad (2.2)$$

where $(\cdot)_+ = \max\{\cdot, 0\}$; $w_s$, $w_s$ are weights of overdosing and underdosing structure $s \in \mathcal{S}$, respectively; and $T_s$, $T_s$ are threshold doses for structure $s \in \mathcal{S}$ indicating at what point the dose in the voxel is penalized. This separation of lower and upper  

---

1The latest water-based dose calculation algorithm by Elekta.
thresholds increases the flexibility of our model by allowing for a range of dose at which no penalty is assigned. The penalty function is normalized according to the structure size $v_s$ so that structures are not given extra significance in the optimization model based on size. For our numerical results, we used only one set of weighting parameters for all the clinical cases. Figure 2.1 illustrates the convexity and non-smoothness of $F_s$.

The SDO model is simply to minimize the total penalty in the treatment plan:

\[
\text{minimize} \quad \sum_{s \in S} \sum_{j=1}^{v_s} F_s(z_{js}) \\
\text{subject to} \quad z_{js} = \sum_{I \in \Theta} \sum_{b \in B} \sum_{c \in C} D_{Ibcjs} t_{Ibc} \quad s \in S, \; j = 1, \ldots, v_s \quad \text{(SDO)}
\]

\[t \geq 0,\]

where $t$ is the vector of all $t_{Ibc}$'s. Since penalty functions $F_s$ are convex, this model is a
convex problem with only non-negativity constraints.

### 2.3 Sector-duration optimization solution methods

#### 2.3.1 Gradient projection

To solve the SDO problem, we employ a gradient projection (GP) algorithm with an Armijo line search \[45, 58\] (see Algorithm 1). In general, the GP method is used to find a local optimum of a differentiable function over a feasible set. For a minimization problem such as SDO, GP starts using any initial feasible point (we used zero-vector in our results). The algorithm then moves the current point in the direction of the negative gradient (steepest descent) of the objective function to find a better point. The algorithm moves some distance (step length \(\lambda\)) along that direction to obtain a new point. It is possible that in this step, we leave the feasible region; in the case of SDO, that means that some \(t_{Ibc}\) become negative. Feasibility is restored by taking the projection of this point to the feasible set (point \(t(\lambda)\)). We used an Armijo condition to find step length \(\lambda\), which is

\[
f(t(\lambda)) - f(t^k) \leq \frac{-\alpha}{\lambda} \|t(\lambda) - t^k\|^2
\]

The algorithm continues until the relative improvement in the objective is less than \(\epsilon\) from one iteration to the next. Empirically, \(\epsilon = 10^{-4}\) in our results.

#### 2.3.2 Interior-point constraint generation

Although GP can solve SDO to find quality plans, the algorithm is known to oscillate near the optimal solution, which results in very slow convergence to achieve the required rate of accuracy. Since functions \(F_s\) are convex functions, Problem SDO is a convex problem with only non-negativity constraints and therefore can be cast as a SILO problem. An
Algorithm 1 Gradient projection algorithm

1: \( k \leftarrow 1, t^k \leftarrow (0, \ldots, 0), \epsilon \leftarrow 10^{-4}, \alpha \leftarrow 10^{-4} \)

2: while improvement > \( \epsilon \) do

3: find \( f(t^k) \) and \( \nabla f(t^k) \)

4: \( t(\lambda) \leftarrow \) projection of \( t^k - \lambda (\nabla f(t^k)/\|\nabla f(t^k)\|) \) into the feasible set

5: find \( \lambda \) such that \( f(t(\lambda)) - f(t^k) \leq \frac{\epsilon}{\lambda} \|t(\lambda) - t^k\|^2 \) (Armijo rule)

6: \( t^{k+1} \leftarrow t(\lambda) \)

7: \( k \leftarrow k + 1 \)

8: end while

9: return \( t^k \)

alternative formulation of \[\text{SDO}\] as an SILO is

\[
\begin{align*}
\text{minimize} & \quad \delta \\
\text{subject to} & \quad g(\hat{t}) + \nabla g(\hat{t})^\top (\hat{t} - t) \leq \delta \quad \forall \hat{t} \in \mathbb{R}^K \\
& \quad t_{Ibc} \geq 0
\end{align*}
\]

where \( K = |\Theta| \cdot |\mathcal{B}| \cdot |\mathcal{C}| \) and \( g : \mathbb{R}^K \rightarrow \mathbb{R} \) is

\[
g(\hat{t}) = \sum_{s \in \mathcal{S}} \sum_{j=1}^{v_s} F_s \left( \sum_{I \in \Theta} \sum_{b \in \mathcal{B}} \sum_{c \in \mathcal{C}} D_{Ibcjs} \hat{t}_{Ibc} \right)
\]

Theoretically, it is possible to approximate the \[\text{SDO-SILO}\] with a semi-definite optimization problem \[11, 73\] which is known to be a polynomially solvable problem. We show that with our implementation of IPCG there is no need to approximate \[\text{SDO-SILO}\] in order to achieve tractability.

Commercial software packages can be used to solve \[\text{SDO-SILO}\]. However, these solvers can solve only classical optimization problems. On the other hand, \[\text{SDO-SILO}\] can be reformulated as a classical optimization problem, specifically \[\text{SOCO}\]. But since this problem contains many piecewise convex quadratic functions, the price of this reformulation
is tens of thousands of additional variables and constraints. The following SOCO formulation has the same optimal solution as SDO-SILO:

\[
\begin{align*}
\min & \quad \delta \\
\text{subject to} & \quad \sqrt{\frac{v_s}{w_s}} (z_{js} - T_s) \leq \bar{y}_{js} \quad s \in S, j = 1, \ldots, v_s \\
& \quad \sqrt{\frac{v_s}{w_s}} (T_s - z_{js}) \leq y_{js} \quad s \in S, j = 1, \ldots, v_s \\
& \quad \sqrt{\sum_{s \in S} \sum_{j=1}^{v_s} (\bar{y}_{js}^2 + y_{js}^2)} \leq \tilde{\delta} \\
& \quad z_{js} = \sum_{I \in \Theta} \sum_{b \in B} \sum_{c \in C} D_{Ibcjs} t_{Ibc} \quad s \in S, j = 1, \ldots, v_s \\
& \quad \bar{y}, y, t \geq 0,
\end{align*}
\]

(SDO-SOCO)

where \( \bar{y} \) and \( y \) are the vectors of all \( \bar{y}_{js} \) and \( y_{js} \), respectively.

Problem SDO-SOCO is an SOCO with one second-order cone constraint of dimension \( 2 \sum_{s \in S} v_s + 1 \) and \( 2 \sum_{s \in S} v_s \) linear constraints. We use MOSEK to solve this problem. Note that it is possible to break the second-order cone constraint into constraints of smaller dimension. Such reformulation, however, significantly increases the number of conic constraints in SDO-SOCO, which may not be ideal for MOSEK.

SDO model can also be reformulated as a quadratic optimization problem as follows:

\[
\begin{align*}
\text{minimize} & \quad \sum_{s \in S} \sum_{j=1}^{v_s} (\bar{y}_{js}^2 + y_{js}^2) \\
\text{subject to} & \quad \sqrt{\frac{v_s}{w_s}} (z_{js} - T_s) \leq \bar{y}_{js} \quad s \in S, j = 1, \ldots, v_s \\
& \quad \sqrt{\frac{v_s}{w_s}} (T_s - z_{js}) \leq y_{js} \quad s \in S, j = 1, \ldots, v_s \\
& \quad z_{js} = \sum_{I \in \Theta} \sum_{b \in B} \sum_{c \in C} D_{Ibcjs} t_{Ibc} \quad s \in S, j = 1, \ldots, v_s \\
& \quad \bar{y}, y, t \geq 0.
\end{align*}
\]

(SDO-QP)

The above Quadratic Programming (QP) model can be efficiently solved by many polynomial algorithms, such as, interior-point methods, active-set algorithms, and more. However, the cost is the increases on the number of isocentres.
2.3.3 Commercial solver MOSEK

MOSEK \cite{72} is one of the widely used tools for solving mathematical optimization problems. Established in 1997, MOSEK is designed to solve linear optimization, quadratic optimization, conic optimization and mixed integer optimization. MOSEK Version 6 is used for our computational results presented in the Results section to solve SDO. The default interior-point method implemented within the solver is used for our experiments.

2.4 Results

This section presents numerical and clinical results of solving SDO treatment planning variants presented in previous sections. Numerical results include the numerical properties of the models and the algorithms used to solve them. The clinical results include the quality of the treatment plans generated by proposed inverse planning models.

The following experiments are based on real patient data provided by the Department of Radiation Oncology at the Princess Margaret Hospital (PMH), Toronto, Ontario, Canada under Health Canada’s Research Ethics Board approval. In this section, all the test problems were done on a desktop computer using Intel\textsuperscript{®} Core\textsuperscript{TM}2 Quad CPU 2.66 GHz processor with 4 GB RAM. All implementations are done using MATLAB 2008b (MathWorks Inc.).

2.4.1 Numerical results

The SDO model is the most simple one comparing to SDO-SOCO, SDO-SILO, and SDO-QP. The number of decision variables is $24 \times \text{Card}(\Theta)$ and they all must be non-negative which forms the only set of constraints. We can exclude the equality constraints in SDO from our counting as they can be easily removed by substitution. Therefore, it can handle a very large number of isocentres.

On the other extreme, SDO-QP has the most number of constraints. SDO-SOCO has...
one more variable). The number of variables in this model is $2 \sum_{s \in S} v_s$, that is, twice as many as the total number of voxels in all structures (equality constraints are excluded). Therefore, the number of variables can significantly grow depend on the size of the voxels. In our experiment, the number of voxels goes up to 203,541 in Case 4. The number of constraints in SDO-QP is $2 \sum_{s \in S} v_s$ plus the non-negativity constraints for each decision variable (SDO-SOCO has one more constraint). Consequently, SDO-QP (and SDO-SOCO) can be quite large and, comparing to SDO, they cannot solve treatment plans with large number of isocentres. The large number of variables and constraints is resulted from resolving the non-smoothness of the SDO objective in SDO-QP and SDO-SOCO.

In SDO-SILO, the number of decision variables is the same as of SDO model. However, the number of constraints is infinitely many but all linear. With the help of IPCG we can sequentially select a few when is needed. Our experiment results (see Chapter 5) shows that a few hundreds of them are usually selected. Therefore the number of isocentres that this model can handle is almost the same as SDO but significantly faster with a guaranty of approximate optimality. The beam-on time, however, is significantly high in the final treatment plan generated by IPCG. The primary driver behind the high BOT is that since some of the components of the duration vector $t$ in SDO-SILO do not change throughout the IPCG procedure (objective function gradient has components equal to zero at the final solution), the initial solution affects the final plan beam-on time. Gradient projection introduces better BOT since it starts with an all-zero vector so that it has more zero components compared to the IPCG that starts somewhere in the middle of the boundary box. An iterative purification technique [41] is used to introduce more zeros in the final plan by IPCG. Theoretical aspects of the IPCG algorithm are presented in Chapter 5.

In terms of algorithm performance, the conformity indices tended to rapidly improve as the number of isocentres increased, but then they reached a plateau (see Figure 2.6), indicating there was a point at which additional isocentres do not improve conformity. As
Figure 2.2: Dose-volume histogram and isodose lines for a sample case (Case 1) using 25 isocentres. Top: Dose-volume histogram for the target (solid line) and OARs (dashed lines). The $V_{100}$ is also indicated with vertical and horizontal dashed lines. Bottom: Cross-sectional views of the target and brainstem showing the conformal prescription isodose line (100% Rx) as well as 50% of the prescription isodose.
indicated in Figure 2.6, beam-on time varied with the number of isocentres and exhibited no clear relationship, but for the most optimal plans was on average 33 minutes longer than manual plans (range: -17min to +91min) when normalized to a calibration dose rate of 3.5Gy/min.

Figures 2.3–2.5 show the isodose curve (top) and DVH (bottom) for the Case 1 plans obtained by the GP algorithm applied to SDO, IPCG algorithm applied to SDO-SILO and MOSEK applied to SDO-SOCO, respectively. The figures show the final dose distribution on the 10 structures when we randomly select 35 isocentres within the GTV. Note that the PTV does not have any clinical use in radiosurgery treatment planning and we used it for only numerical purposes. The numerical results for a complete set of isocentre selection is presented in Chapter 5. The dots represent the selected isocentres projected to the plane of the slice. The colours of the dots show the level of projected isocentre compare to the projection plane. The blue dots are above, the red dots are under, and the black dots are on the plane. The solid curve in the isodose curve shows the area that receives at least 100% of the prescribed dose. This area includes the actual GTV. The dashed curve shows the area that receives at least 50% of the prescribed dose. This area includes the PTV and a part of the brainstem. The dashed vertical line in the DVH shows the prescription dose, which is 12Gy in this example, and the dash-dotted horizontal line is placed at the 100% of the structure volume. Since IPCG and MOSEK use an interior-based approach, the final solutions are relatively closer to the one of IPCG compare to the solution generated by GP. It is hardly possible to distinguish the final plan of generated by MOSEK from the final plan of IPCG.

The dose volume histograms in these figures show that all three models are equally good in the sense that less than 1% of the GTV receives less than 100% of the prescribed dose, and less than 1% of the GTV receives more than 150% of the prescribed dose. Also, except for the brainstem, all other critical structures receive less than 25% of the prescribed dose. These are clinically acceptable treatment plans, which shows that all
Figure 2.3: Treatment plan generated by the GP algorithm shows similar quality as the plan generated by IPCG and MOSEK.
Figure 2.4: Treatment plan generated by the IPCG algorithm shows similar quality as the plan generated by GP and MOSEK, but significantly faster (see Chapter 5).
three models lead to good quality treatment plans. However, IPCG can handle the largest number of isocentres and perform the fastest, as detailed in Chapter 5. Due to the randomness of the selected isocentres, not surprisingly, the isodose curves in all three approaches are not quite conformal to the target boundary. In next section we show that using GSP significantly improves the conformity of the plan.

2.4.2 Clinical results

For the clinical cases presented in this section, the grassfire-based algorithm presented in [33] is used to generate isocentres needed for each case. Adequate coverage was achieved by the algorithm as indicated by a $V_{100}$ greater than 98% in all 11 targets. The mean difference in $V_{100}$ between the forward and the inverse plans was 0.2% (range: -2.0% to 2.4%) (see Table 2.1). In all plans (except in Case 7), the clinical objective for brainstem sparing was achieved, and for the inverse plans for acoustic cases, the brainstem dose was lesser than or equal to the forward plans. When normalizing the resulting inverse plans to have the same coverage ($V_{100}$) as the forward plans, the mean difference in dose to 1mm$^3$ of brainstem was -0.24Gy (range: -2.40Gy to 2.02Gy) in favour of the inverse plans. The mean difference in conformity index between inverse and forward plans was -0.12 (range: -0.27 to +0.03) and +0.08 (range: 0.00 to +0.17) for classic and Paddick definitions, respectively, both favouring the inverse plans. The least conformal plan (Case 6) occurred due to a large and irregularly shaped target adjacent to the brainstem.

The dosimetric results, dose volume histogram (DVH) and isodose lines, for a typical case (Case 1) is presented in Figure 2.2 for the plan with 25 isocentres. In the cross-sectional images, the dashed and solid lines represent the 50% and 100% isodose (relative to the Rx dose) lines, respectively, which illustrate the conformity of the treatment.
Figure 2.5: Treatment plan generated by the MOSEK algorithm shows similar quality as the plan generated by GP and IPCG.
Table 2.1: Plan quality summary using GP algorithm. Fwd: forward plans. Inv: inverse plans. *: number of isocentres in inverse plans obtained by GSP algorithm.

<table>
<thead>
<tr>
<th>Case</th>
<th>Isocentre*</th>
<th>CI_{Paddick} Fwd</th>
<th>CI_{Paddick} Inv</th>
<th>CI_{Classic} Fwd</th>
<th>CI_{Classic} Inv</th>
<th>Brainstem dose (Gy) Fwd</th>
<th>Brainstem dose (Gy) Inv</th>
<th>BOT (min) Fwd</th>
<th>BOT (min) Inv</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>25</td>
<td>0.85 0.90</td>
<td>1.14 1.07</td>
<td>14.4 13.1</td>
<td>32.4 45.8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2a</td>
<td>40</td>
<td>0.84 0.90</td>
<td>1.17 1.09</td>
<td></td>
<td>3.0 5.0</td>
<td>41.12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2b</td>
<td>45</td>
<td>0.80 0.93</td>
<td>1.23 1.05</td>
<td>3.0 5.0</td>
<td>45.81</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>35</td>
<td>0.81 0.82</td>
<td>1.15 1.18</td>
<td>14.6 12.2</td>
<td>34.3 17.98</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4a</td>
<td>3</td>
<td>0.77 0.94</td>
<td>1.30 1.03</td>
<td></td>
<td>10.34</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4b</td>
<td>40</td>
<td>0.83 0.93</td>
<td>1.18 1.05</td>
<td>1.8 2.6</td>
<td>25.2 72.88</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4c</td>
<td>20</td>
<td>0.82 0.94</td>
<td>1.21 1.03</td>
<td></td>
<td>33.14</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>20</td>
<td>0.82 0.87</td>
<td>1.20 1.11</td>
<td>14.2 13.3</td>
<td>24.1 60.30</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>50</td>
<td>0.69 0.73</td>
<td>1.40 1.31</td>
<td>14.9 14.9</td>
<td>60.8 92.37</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7a</td>
<td>10</td>
<td>0.67 0.81</td>
<td>1.38 1.18</td>
<td>16.9 16.9</td>
<td>60.2 20.91</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7b</td>
<td>35</td>
<td>0.91 0.91</td>
<td>1.07 1.05</td>
<td></td>
<td>65.11</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure 2.6: Treatment plan metrics for a sample case (Case 1) as a function of number of isocentres. BOT (h) represents the beam-on time of the plans in hours. The figure shows introducing more isocentres does no guarantee conformity improvement or beam-on increment.
2.5 Conclusions

We have shown that mathematical frameworks successful in IMRT optimization can be applied to Leksell Gamma Knife® Perfexion™ inverse planning. These models are flexible enough that despite the large number of voxels in Perfexion™ treatments, solutions can be obtained in a clinically viable amount of time using a standard GP optimization algorithm. We implemented our home-developed IPCG algorithm to solve the sector duration optimization model and showed that our algorithm outperforms MOSEK as well as the GP algorithm.

The results of this study indicate that our approach to the inverse problem for radiosurgery on PFX yields conformal treatment plans that satisfy the clinical objectives. These automatically generated plans are capable of being delivered on the treatment unit although the beam-on times are, on average, longer than the manually-created (clinical) plans. Nevertheless, the framework for posing the problem is created and can be used to guide treatment planners to explore the tradeoffs between delivery efficiency and dose conformity.

The amount of published literature on inverse planning for Leksell Gamma Knife® is limited, mainly due to the time and effort needed to deliver the resulting complex treatment plans generated by optimization approaches [28, 29, 53, 77, 78]. Although some of the results [28, 29] indicate that highly tailored dose-distributions can be automatically generated, the delivery of these plans would involve multiple collimator changes as well as labor-intensive manual plugging of individual channels to fully realize the outcome of the inverse planning. Our approach is specifically designed to exploit PFX’s automated collimator size changes and couch positioning. The inverse plans obtained by our approach show favourable conformity compared to previous works [28, 29]. In addition, OAR-sparing is explicitly considered in our approach, while the focus of other studies [28, 29, 53, 77, 78] was only delivering dose to target structures.

We make use of convex quadratic optimization, solved using GP, IPCG, and MOSEK.
Previous approaches \cite{28, 29} focused on nonlinear programming and nonlinear mixed-integer programming optimization which are data-intensive and computationally intractable for large structures, but appear to perform faster than our GP method for small structures (on average almost three times faster for roughly the same number of voxels). We use a geometric GSP approach to rapidly ($< 3s$) obtain isocentre positions, and then use the SDO to optimize the shape and weighting of the resulting shots, thereby combining the efficiency of a purely geometric isocentre selection with the efficiency of a convex quadratic beam-weighting optimization. It should be mentioned that our SDO model does not limit the shot shapes to be spherical \cite{29}, or even elliptical \cite{28}. The SDO formulation can handle all possible combinations of sectors with different collimator sizes at the same time, thereby allowing for increased flexibility and customization in the treatment plans.

Another issue which we address is the incorporation of organ-at-risk sparing into the optimization model. Previous Leksell Gamma Knife$\text®$ optimization methods have considered only optimization of the dose distribution for the target structure \cite{28, 29, 29, 29, 53, 77} to keep the size and complexity of the problem manageable. Also, methodologies \cite{76, 77} that are only based on geometric approaches such as skeletonization and grassfire can not easily incorporate OARs in the plans. Our inclusion of OARs results in a substantially larger model: as many as 130,000 voxels, compared to 30,000 voxels in a previous study \cite{29}. Similarly, the model in the present study can consider treatments for multiple targets simultaneously. This feature lets us observe the effects of multiple targets on each other. In plans with multiple targets, shots placed in one target deliver some dose to other targets as well. Therefore, if the shot durations for targets are optimized separately for each target, the mutual effect of the shots on other targets will not be incorporated and optimized. In our SDO model, all the targets in a plan are optimized at the same time, and the dose spillage from one target to the other is accounted for in the optimization model.
Although our SDO model described is flexible and computationally feasible, a few limitations and areas for improvement exist.

After finding the sector durations with our SDO model, the sectors are combined together into deliverable shots supported by the treatment unit. In this process any shot of duration 10 seconds or less is removed to reflect the treatment unit limitations and minimize the effect of shutter dose. Although we report beam-on time in our results, it is not explicitly considered in the optimization model. The obtained beam-on times for our plans are acceptable, but the rather large number of shots can be inconvenient if the shots are manually entered into PFX. Also, the number of shots for each plan depends on the optimization solution from SDO and therefore is not predictable. Incorporating beam-on time in the SDO model is examined in Chapter 4. The total computational time for our approach is rather high (although acceptable) due to the GP method used for SDO.

We have shown that mathematical frameworks successful in IMRT optimization can be applied to Leksell Gamma Knife® Perfexion™ inverse planning. These models are flexible enough that despite the large number of voxels in PFX treatments, solutions can be obtained in a clinically viable amount of time.
Chapter 3

Simultaneous sector duration and isocentre optimization

3.1 Introduction

Unlike previous Perfexion™ optimization approaches that addressed isocentre and sector duration optimization sequentially as presented in Chapter 2, in this chapter we optimize both simultaneously. We employ a mixed-integer linear programming (MILP) model to optimize the candidate isocentres to use and their associated sector durations (i.e., shot shapes and intensities). Hard limits on the allowable number of isocentres per target can be specified, and we present an intuitive method of automatically determining reasonable bounds based on target shape and radiation shot volume. The model emphasizes selecting fewer isocentres as a proxy to managing beam-on time. In order to obtain a tractable optimization problem, we develop isocentre and voxel sampling techniques to reduce the number of constraints.

Each isocentre represents an integer variable that must assume a value of 0 (not used) or 1 (used) in the optimization model. For each isocentre, the amount of time to deliver radiation from each of the eight banks for each of the three collimator sizes
requires \( 8 \times 3 = 24 \) continuous variables. However, there are generally \( \sim 67,000 \) target voxels, so allowing every target voxel to be included into the set of candidate isocentres results in \( \sim 67,000 \) integer variables, \( \sim 1,608,000 \) continuous variables, and \( \sim 134,000 \) linear constraints that provide upper and lower bounds on dose to each target voxel. An MILP model of this size requires \( \sim 5 \) TB of memory and days of computation time. We therefore reduce the number of isocentres and voxel doses that must be considered in the optimization model using sampling techniques.

Rather than consider each target voxel as a potential isocentre, we intelligently select a subset of these voxels to be candidate isocentres. The candidate pool of isocentres is selected using a fast grassfire and sphere-packing algorithm \[33\] tuned to select a user-provided number of isocentres. The number of isocentres selected can be as many as the computer memory limit will allow. Typically, we select 100 candidate isocentres per target.

In the optimization model, dose to target voxels must be constrained to be within user-specified lower and upper limits, resulting in two constraints for every voxel. As previously stated, the extremely large number of voxels yields an unsupportably large number of constraints, each of which requires computationally expensive dose calculations. To reduce the number of constraints, rather than constrain the dose delivered to every target voxel, we instead only explicitly constrain the dose to a subset of voxels (e.g., every other voxel) with the expectation that if the dose to one voxel is controlled, dose to immediately adjacent voxels will also be indirectly controlled.

Similarly, to improve the computation time of the optimization, instead of minimizing the dose to all OAR voxels, we instead only explicitly minimize dose to a subset of OAR voxels. Specifically, we only consider minimizing dose to healthy voxels in a ring around the target; the ring used in this study is the PTV minus GTV for each target, although we only treat the GTV in stereotactic radiosurgery plans. Intuitively, if dose to voxels immediately surrounding the target is minimized, then, assuming the isocentres are
within the GTV, dose to voxels further away will also be minimized.

3.2 A tractable mixed-integer model for SDIO

The simultaneous sector-duration and isocentre optimization (SDIO) model is described as follows. Let $\Theta_s$ be the set of candidate isocentres in target $s \in \mathcal{T}$, the set of all target structures. The grassfire and sphere-packing isocentre selection method \[33\] is performed individually for each target $s \in \mathcal{T}$ to obtain $\Theta_s$. For simplicity, define $\bar{\Theta} = \cup_{s \in \mathcal{T}} \Theta_s$. In the optimization, let variable $\lambda_I \in \{0, 1\}$ indicate whether or not isocentre $I \in \bar{\Theta}$ is selected for use in the treatment.

The SDIO optimization model minimizes dose to the healthy voxels in a ring surrounding the target while forcing target voxels to receive a dose in the range of $[T_s, \bar{T}_s]$ for all targets $s \in \mathcal{T}$. To implement the voxel sampling techniques, rather than incorporate each voxel $j$ in every structure in the constraints and objective function, only voxel subset $V_s$ for structure $s \in \mathcal{S} \cup \mathcal{T}$ is considered, where $\mathcal{S}$ is the set of OAR structures. Note that the definition of $V_s$ is generalizable to any sampling strategy, including the ring structure used to minimize dose outside the target.

As large beam-on times are a consequence of mathematically-driven treatment plans, we design our model in such a way as to prefer treatments with fewer numbers of isocentres, where the number of isocentres is treated as a surrogate for beam-on time. To incorporate this preference for fewer isocentres, the total number of selected isocentres ($\sum_{I \in \bar{\Theta}} \lambda_I$) is also minimized, and adjusted with weighting parameter $\beta$ to control how much emphasis to place on isocentre reduction.
Chapter 3. Simultaneous sector duration and isocentre optimization

The SDIO optimization model is then

\[
\text{minimize } \sum_{I \in \Theta} \sum_{b \in B} \sum_{c \in C} \left( \sum_{s \in S} \left| V_s \right| \sum_{j \in V_s} D_{Ibcj} \right) t_{Ibc} + \frac{\beta}{|\Theta|} \sum_{I \in \Theta} \lambda_I
\]

subject to

\[
T_s \leq \sum_{I \in \Theta_s} \sum_{b \in B} \sum_{c \in C} D_{Ibcj} t_{Ibc} \leq T_s \quad s \in \mathcal{T}, \ j \in V_s
\]

\[
K_s \leq \sum_{I \in \Theta_s} \lambda_I \leq \bar{K}_s \quad s \in \mathcal{T}
\]

\[
\lambda_I t_{\text{MIN}} \leq t_{Ibc} \leq \lambda_I t_{\text{MAX}} \quad I \in \Theta, \ b \in B, \ c \in C
\]

\[
\lambda_I \in \{0, 1\} \quad I \in \Theta,
\]

where \(K_s\) and \(\bar{K}_s\) are the minimum and maximum numbers of isocentres in each target structure \(s \in \mathcal{T}\). The third constraint ensures that if an isocentre is not selected, all sector durations for that isocentre must be zero. Conversely, if an isocentre is selected, all sector durations for that isocentre must be within a specified range of \([t_{\text{MIN}}, t_{\text{MAX}}]\).

Note that in the SDIO objective function, the first term is divided by constant values \(|V_s|\) and the second term is divided by constant value \(|\Theta|\). Although the effect of these values in the optimization could be absorbed by the single parameter \(\beta\), the \(|V_s|\) and \(|\Theta|\) values serve as scaling factors so that \(\beta\) does not need to be tuned for each individual patient case or sampling strategy. Regardless of whether the case has a large or small number of voxels, or whether there are many or few candidate isocentres, \(\beta\) will represent the same emphasis on reducing the number of selected isocentres. Thus, these scaling factors make the model easier to implement on a variety of patients.

3.3 Determination of isocentre bounds

The lower isocentre limit \(K_s\) for any target structure \(s \in \mathcal{T}\) can be provided by the user, though in this study, it is calculated with the following strategy. Define \(B_0\), \(B_1\), and \(B_2\) as the volume of spheres of radius 8mm, 4mm, and 2mm, respectively, to approximate spherical shots, as if all sectors were open to the same collimator size and delivered
radiation for the same duration. For each sphere size, we calculate $K^i_s$, the maximum number of whole spheres of size $i = 1, 2, 3$ whose volume can fit in the remaining target; that volume is then removed from the target. Let $V_i$ be the remaining volume of the target after considering all sphere sizes larger than $i$. $K^i_s$ and $V_i$ are calculated as

$$K^i_s = \lfloor V_{i-1}/B_{i-1} \rfloor \quad \forall i = 1, 2, 3$$

$$V_i = V_{i-1} - K^i_s B_{i-1}$$

(3.1)

Finally, $K_s = K^1_s + K^2_s + K^3_s$.

The upper bound for number of isocentres in target $s \in \mathcal{T}$, $\bar{K}_s$, is calculated similarly to the lower bound. Instead, we use only 74% of the remaining volume, recursively:

$$\bar{K}^i_s = \lfloor 0.74 V_{i-1}/B_{i-1} \rfloor \quad \forall i = 1, 2, 3$$

$$V_i = V_{i-1} - \bar{K}^i_s B_{i-1}$$

(3.2)

Then, the upper bound $\bar{K}_s = \bar{K}^1_s + \bar{K}^2_s + \bar{K}^3_s + 1$. The isocentre bound calculations use the idea of Kepler’s conjecture in the concept of sphere packing [44]. By this conjecture, one can fill at most approximately 74% of a volume by spheres of the same size.

### 3.4 Evaluation of the model

The model was implemented in Matlab 2008b (The Mathworks, Inc.) on a 4 Dual-Core AMD Opteron™ Processor 2.2.GHz in an CentOS 2.6 platform with 40GB RAM. Gurobi Optimizer 2.0 [42] is used to solve the SDIO model, and is called by gurobi-mex v1.6 MATLAB interface [82]. The Gurobi optimization routine was set to terminate when the MIP gap is less than 10% or the time limit of 10h is reached.

The model was tested on seven radiosurgery cases comprising 11 targets, as specified in Table 1.1. We use the same clinical objectives as presented in 2.4.2. Three variations of SDIO were considered. First, no limits were placed on the number of isocentres, allowing SDIO to pick as many as isocentres needed from the set of predefined candidates;
Table 3.1: Comparison of the three isocentre limit scenarios over the seven test cases. Average values are shown, with lower and upper values in brackets.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>time (min)</th>
<th>MIP gap %</th>
</tr>
</thead>
<tbody>
<tr>
<td>No-limit</td>
<td>455 [138,600]</td>
<td>18.88 [7.82,66.58]</td>
</tr>
<tr>
<td>Loose-limit</td>
<td>206 [11,600]</td>
<td>9.54 [3.78,22.08]</td>
</tr>
<tr>
<td>Tight-limit</td>
<td>384 [56,600]</td>
<td>10.17 [4.46,16.41]</td>
</tr>
</tbody>
</table>

we call this variation the “no-limit scenario”. Second, lower and upper bounds on the number of isocentres were calculated by equations 3.1 and 3.2, called the “loose-limit scenario”. Finally, tight bounds on the number of isocentres, specifically, the number of isocentres used in the clinical treatment, was enforced; we call this scenario the “tight-limit scenario”.

We anticipate that with increasing strictness on the number of isocentres that can be used, there will be computational gains from the smaller solution space. The smaller search space may allow for better solutions to be found by improved convergence of the optimization routines, or quality solutions may be excluded from the search space. We will compare performance of the three scenarios to determine whether restricting the solution is helpful or harmful overall.

3.5 Results

3.5.1 Numerical results

Table 3.1 presents the computation times and MIP gaps in final solution for each isocentre limit case. In the no-limit scenario (Table 3.3), Gurobi frequently terminated at 10h computation time before reaching the desired MIP gap, despite aggressive voxel sampling
Table 3.2: Plan quality comparison of the three isocentre limit scenarios over the seven test cases. Average values are shown, with lower and upper values in brackets.

<table>
<thead>
<tr>
<th>Percent improvement over forward plans</th>
<th>CI\textsubscript{Paddick}</th>
<th>CI\textsubscript{Classic}</th>
<th>Brainstem dose</th>
</tr>
</thead>
<tbody>
<tr>
<td>No-limit</td>
<td>8.15 [-2.2, 15.94]</td>
<td>8.05 [-3.74, 13.8]</td>
<td>9.14 [0, 36.67]</td>
</tr>
<tr>
<td>Tight-limit</td>
<td>6.44 [-1.1, 14.29]</td>
<td>5.07 [-1.87, 14.62]</td>
<td>-21.01 [-133.33, 11.64]</td>
</tr>
</tbody>
</table>

of as few as 5\% of all target voxels. Even though Gurobi could only reach on average a \sim 19\% MIP gap (Table 3.1) and only 5-50\% of target voxel doses were constrained, the inverse treatments still achieved improved dose conformity for all but one target. Tables 3.1 and 3.2 also show that the computation time can benefit from placing loose bounds on the number of isocentres with only slight compromise to the plan quality.

### 3.5.2 Clinical results

Table 3.2 provides a summary of the inverse plans quality improvements over the forward plans in terms of clinical metrics. Clinical treatment quality metrics for the seven test cases are presented for the no-limit (Table 3.3), loose-limit (Table 3.4), and tight-limit (Table 3.5) scenarios. The voxel sample sizes ranged from 5\% to 50\%, and were obtained through empirical testing. In all three isocentre bounding scenarios, the inverse plans for all cases are clinically satisfactory, and generally outperform the forward plans in clinical measures. However, the beam-on times of the inverse plans are higher than the forward plans. The brainstem was spared in all plans except in Case 7, which is a particularly challenging case due to irregular target shapes and adjacency to the brainstem. However, the inverse plan was able to reduce dose delivered to the brainstem compared to the
forward plan. In fact, brainstem dose was reduced in six of the cases, and stayed the same in one case.

Dose-volume histograms and isodose lines for several slices of a typical no-limit case (Case 1 in Table 3.3) are presented in Figure 3.1. The DVHs show that the target receives acceptable dose while all structures are spared. In the cross-sectional images, the dashed and solid lines represent the 50% and 100% isodose lines (relative to the prescription dose), respectively, which illustrate the conformity of the treatment. Because treatments for all inverse plans were very similar regardless of isocentre bounds, DVHs and slices for only this one representative case are presented, although DVHs and slices were examined and verified to be clinically acceptable for all plans.

In the loose-limit scenario (Table 3.4), approximately 54% computation time savings were realized by bounding the number of isocentres per target, and the MIP gaps averaged 9.54%, about 50% improvement over the no-limit scenario (Table 3.1). Both the Paddick and classic conformity indices improved for 10 of the 11 targets. Brainstem dose improved for four of the seven cases, with the worst change from the forward plans being 2.4Gy additional dose in Case 2; however, the total brainstem dose was still 5.4Gy, well within clinical guidelines. It is important to note that the total 5.4Gy brainstem dose in Case 2 represented a degradation of 80% from the forward plan, skewing the averages in Table 3.2. Eliminating Case 2 from the average brainstem dose improvement calculations yields an average 3.93% improvement over the forward plans.

In the tight-limit scenario (Table 3.5), the number of isocentres was fixed to the number of isocentres used clinically for each case, but the actual isocentre positions were selected by the SDIO model. With these tight bounds on the number of integer values equal to 1, computation times were 16% improved over the no-limit scenario, but 86% worse than the loose-limit scenario. The average MIP gaps similarly improved 46% over the no-limit scenario, but worsened by 7% compared to the loose-limit scenario. However, the worst MIP gap found in the tight-limit scenario (16.41%) is better than the
Figure 3.1: Dose-volume histogram and isodose lines for a sample case (Case 1, Table 3.3). 100% dose refers to 12Gy in this case. Top: Dose-volume histogram for the target (solid line) and OARs (dashed lines). The $V_{100}$ is also indicated with vertical and horizontal dashed lines. Bottom: Cross-sectional views of the target and brainstem showing the conformal prescription isodose line (100% Rx) as well as 50% of the prescription isodose.
Table 3.3: Plan quality summary in the no-limit scenario, where there are no bounds on the number of isocentres per target. Fwd: forward plans. Inv: inverse plans using SDIO model. BOT: beam-on time. ∗: (number of isocenters selected)/(candidate pool size).

<table>
<thead>
<tr>
<th>Case</th>
<th>Isocenter</th>
<th>CI&lt;sub&gt;Paddick&lt;/sub&gt;</th>
<th>CI&lt;sub&gt;Classic&lt;/sub&gt;</th>
<th>Brainstem dose (Gy)</th>
<th>BOT (min)</th>
<th>Sample size</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Fwd</td>
<td>Inv*</td>
<td>Fwd</td>
<td>Inv</td>
<td>Fwd</td>
<td>Inv</td>
</tr>
<tr>
<td>1</td>
<td>18</td>
<td>44/100</td>
<td>0.85</td>
<td>0.90</td>
<td>1.14</td>
<td>1.07</td>
</tr>
<tr>
<td>2a</td>
<td>20</td>
<td>37/100</td>
<td>0.84</td>
<td>0.88</td>
<td>1.17</td>
<td>1.11</td>
</tr>
<tr>
<td>2b</td>
<td>13</td>
<td>100/100</td>
<td>0.80</td>
<td>0.86</td>
<td>1.23</td>
<td>1.13</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>12/100</td>
<td>0.81</td>
<td>0.88</td>
<td>1.15</td>
<td>1.11</td>
</tr>
<tr>
<td>4a</td>
<td>3</td>
<td>25/100</td>
<td>0.77</td>
<td>0.86</td>
<td>1.30</td>
<td>1.13</td>
</tr>
<tr>
<td>4b</td>
<td>18</td>
<td>53/100</td>
<td>0.83</td>
<td>0.92</td>
<td>1.18</td>
<td>1.06</td>
</tr>
<tr>
<td>4c</td>
<td>12</td>
<td>100/100</td>
<td>0.82</td>
<td>0.90</td>
<td>1.21</td>
<td>1.08</td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>13/100</td>
<td>0.82</td>
<td>0.86</td>
<td>1.20</td>
<td>1.14</td>
</tr>
<tr>
<td>6</td>
<td>26</td>
<td>24/100</td>
<td>0.69</td>
<td>0.80</td>
<td>1.40</td>
<td>1.22</td>
</tr>
<tr>
<td>7a</td>
<td>6</td>
<td>3/26</td>
<td>0.67</td>
<td>0.75</td>
<td>1.38</td>
<td>1.30</td>
</tr>
<tr>
<td>7b</td>
<td>24</td>
<td>26/100</td>
<td>0.91</td>
<td>0.89</td>
<td>1.07</td>
<td>1.11</td>
</tr>
</tbody>
</table>
Table 3.4: Plan quality summary in the loose-limit scenario, where there are bounds on the number of isocentres per target. Fwd: forward plans. Inv: inverse plans using SDIO model. BOT: Beam-on time. \(^*\): \((\text{number of isocenters selected})/(\text{candidate pool size})\) [lower bound, upper bound].

<table>
<thead>
<tr>
<th>Case</th>
<th>Fwd</th>
<th>Inv*</th>
<th>CI\text{Paddick}</th>
<th>CI\text{Classic}</th>
<th>Brainstem dose (Gy)</th>
<th>BOT (min)</th>
<th>Sample size</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Fwd</td>
<td>Inv</td>
<td>Fwd</td>
<td>Inv</td>
<td>Fwd</td>
<td>Inv</td>
<td>Fwd</td>
</tr>
<tr>
<td>1</td>
<td>18</td>
<td>17/100 [17,42]</td>
<td>0.85</td>
<td>0.89</td>
<td>1.14</td>
<td>1.10</td>
<td>14.4</td>
</tr>
<tr>
<td>2a</td>
<td>20</td>
<td>45/100 [10,49]</td>
<td>0.84</td>
<td>0.88</td>
<td>1.17</td>
<td>1.10</td>
<td>3.0</td>
</tr>
<tr>
<td>2b</td>
<td>13</td>
<td>33/100 [13,33]</td>
<td>0.80</td>
<td>0.81</td>
<td>1.23</td>
<td>1.19</td>
<td>105</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>10/100 [10,13]</td>
<td>0.81</td>
<td>0.86</td>
<td>1.15</td>
<td>1.14</td>
<td>14.6</td>
</tr>
<tr>
<td>4a</td>
<td>3</td>
<td>4/20 [4,9]</td>
<td>0.77</td>
<td>0.86</td>
<td>1.30</td>
<td>1.14</td>
<td>11</td>
</tr>
<tr>
<td>4b</td>
<td>18</td>
<td>46/100 [14,85]</td>
<td>0.83</td>
<td>0.92</td>
<td>1.18</td>
<td>1.05</td>
<td>1.8</td>
</tr>
<tr>
<td>4c</td>
<td>12</td>
<td>32/100 [7,34]</td>
<td>0.82</td>
<td>0.91</td>
<td>1.21</td>
<td>1.07</td>
<td>89</td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>11/100 [11,26]</td>
<td>0.82</td>
<td>0.83</td>
<td>1.20</td>
<td>1.18</td>
<td>14.2</td>
</tr>
<tr>
<td>6</td>
<td>26</td>
<td>39/100 [11,43]</td>
<td>0.69</td>
<td>0.83</td>
<td>1.40</td>
<td>1.17</td>
<td>14.9</td>
</tr>
<tr>
<td>7a</td>
<td>6</td>
<td>5/40 [5,5]</td>
<td>0.67</td>
<td>0.79</td>
<td>1.38</td>
<td>1.22</td>
<td>16.9</td>
</tr>
<tr>
<td>7b</td>
<td>24</td>
<td>25/100 [3,27]</td>
<td>0.91</td>
<td>0.85</td>
<td>1.07</td>
<td>1.17</td>
<td>113</td>
</tr>
</tbody>
</table>
worst MIP gap found in the loose-limit scenario (22.08%). In terms of clinical metrics, the tight-limit scenario improved on both the Paddick and classic conformity indices compared to the forward plans in 10 out of 11 targets, but only strictly improved on the brainstem dose in one case (though all cases achieved clinically satisfactory brainstem dose). The brainstem dose increase from 1.8Gy in the Case 4 forward plan to 4.2Gy in the tight-limit inverse plan constitutes a 133% change, even though 4.2Gy is still a very low dose. Ignoring this case, the average brainstem dose improvement over the forward plans increases from -21.01% to -2.29%.

The effects of voxel sample size on treatment plan quality and on computation time were examined. Figure 3.2 (left) shows that as the sample size decreases, target coverage and conformity indices worsen, although they remain good even with small sampling. Figure 3.2 (right) shows that beam-on time also worsens as the sample size decreases, though computation time drastically improves, as expected. The last column of Tables 3.3, 3.4, and 3.5 represents the percentage of voxels used as candidate isocenters. Considering the fact that every spherical shape shot of size 4mm (in diameter) may cover approximately 27 voxels of size $1\text{mm} \times 1\text{mm} \times 1\text{mm}$, few voxels should need to be constrained to receive prescribed dose and expect a reasonable target coverage for the rest. However, proper selection of the sample voxels is needed based on the geometry of the target.

### 3.6 Conclusions

The results indicate that our MIP approach to complete inverse planning for radiosurgery on Perfexion\textsuperscript{TM} yields conformal treatment plans that satisfy clinical objectives. These automatically generated plans are capable of being delivered on the treatment unit, although the beam-on times are longer than the manually-created clinical plans. Nevertheless, we present a mathematical framework for solving the simultaneous isocentre and
Table 3.5: Plan quality summary with tight bounds on the number of isocentres, where the number of isocentres per target must be the same number used in the forward (clinical) plan. Fwd: forward plans. Inv: inverse plans using SDIO model. BOT: Beam-on time. ∗: (required number of isocenters)/(candidate pool size).

<table>
<thead>
<tr>
<th>Case</th>
<th>Isocenter</th>
<th>CI&lt;sub&gt;Paddick&lt;/sub&gt;</th>
<th>CI&lt;sub&gt;Classic&lt;/sub&gt;</th>
<th>Brainstem dose (Gy)</th>
<th>BOT (min)</th>
<th>Sample size</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Fwd</td>
<td>Inv</td>
<td>Fwd</td>
<td>Inv</td>
<td>Fwd</td>
</tr>
<tr>
<td>1</td>
<td>18/100</td>
<td>0.85</td>
<td>0.90</td>
<td>1.14</td>
<td>1.10</td>
<td>14.4</td>
</tr>
<tr>
<td>2a</td>
<td>20/100</td>
<td>0.84</td>
<td>0.88</td>
<td>1.17</td>
<td>1.11</td>
<td>3.0</td>
</tr>
<tr>
<td>2b</td>
<td>13/100</td>
<td>0.80</td>
<td>0.86</td>
<td>1.23</td>
<td>1.13</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>8/100</td>
<td>0.81</td>
<td>0.87</td>
<td>1.15</td>
<td>1.14</td>
<td>14.6</td>
</tr>
<tr>
<td>4a</td>
<td>3/20</td>
<td>0.77</td>
<td>0.88</td>
<td>1.30</td>
<td>1.11</td>
<td></td>
</tr>
<tr>
<td>4b</td>
<td>18/100</td>
<td>0.83</td>
<td>0.85</td>
<td>1.18</td>
<td>1.15</td>
<td>1.8</td>
</tr>
<tr>
<td>4c</td>
<td>12/100</td>
<td>0.82</td>
<td>0.90</td>
<td>1.21</td>
<td>1.08</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>13/100</td>
<td>0.82</td>
<td>0.86</td>
<td>1.20</td>
<td>1.15</td>
<td>14.2</td>
</tr>
<tr>
<td>6</td>
<td>26/100</td>
<td>0.69</td>
<td>0.74</td>
<td>1.40</td>
<td>1.33</td>
<td>14.9</td>
</tr>
<tr>
<td>7a</td>
<td>6/40</td>
<td>0.67</td>
<td>0.73</td>
<td>1.38</td>
<td>1.34</td>
<td>16.9</td>
</tr>
<tr>
<td>7b</td>
<td>24/100</td>
<td>0.91</td>
<td>0.90</td>
<td>1.07</td>
<td>1.09</td>
<td></td>
</tr>
</tbody>
</table>
sector duration optimization problem that can be used to guide treatment planners in exploring the trade-offs between delivery efficiency and dose conformity.

Since the number of isocentres in the final plan is sometimes close to the lower and upper limits (specifically, in the loose-limit scenario), it is likely that we can improve the quality of the plans by providing better bounds on the number of isocentres. The results presented in Tables 3.3 and 3.5 show that at most 50% of the target voxels need to be explicitly constrained dose in order to make quality plans, and in some targets, as few as 5% of target voxels need to be constrained. One approach is to investigate bounds on the number of isocentres based on the shape of the target, surrounding organs, and treatment quality rather than just the volume of the target. In terms of efficiency, voxel sampling techniques make the SDIO model tractable, however, the size of the sample set is important. The sample set size is highly dependent on the size of each voxel (CT resolution) and the shot sizes. For instance, a ”spherical” shot of size 4mm in diameter may approximately cover three voxels of size 1mm × 1mm × 1mm from each direction away from the isocentre. So, approximately 27 voxels around the isocentre are covered by the shot. Therefore, if properly chosen, approximately 4% of the target voxels seem enough to produce a quality plan. Surrounding OARs can also affect the sample size.
We observed that more isocentres close to the boundaries of the adjacent OARs results in better conformity. More investigation is needed to find better sampling techniques to minimize the sample size.
Chapter 4

Methods to control beam-on time

4.1 Introduction

Although the use of mathematical optimization techniques can greatly improve the quality of treatment plans in various radiation therapy treatment settings, one complication is the potentially clinically unrealistic nature of optimized treatments. The difficulty arises from two factors: (1) machine limitations that govern the minimum amount of radiation delivery time, and (2) long treatment times due to the complexity of optimized treatments. In the first scenario, if a particular configuration of the radiation delivery device is used, then it typically must deliver radiation for a minimum length of time. Incorporation of such requirements in a mathematical model generally requires additional constraints and binary variables, increasing the difficulty of the optimization. In the second scenario, mathematically optimized treatments commonly assign (small amounts of) radiation to be delivered from many configurations, drastically increasing the time needed to deliver the treatment (beam-on time). We examine these two issues within the penalty-based SDO in Chapter 2 and the MIP SDO model in Chapter 3 to reduce beam-on time and to ensure that machine limitations regarding delivery times are met.
4.2 BOT calculation and analytic approximations

Assume the following uni-isocentre treatment plan is the final plan generated by the inverse planning system explained in the previous chapters:

\[
\begin{align*}
\text{isocenter} & \quad \begin{array}{c} \text{sector durations} \\
(250, 230, 90) & \begin{array}{cccc}
\frac{4\text{mm}}{} & \frac{8\text{mm}}{} & \frac{16\text{mm}}{} \\
\frac{4\text{mm}}{} & \frac{8\text{mm}}{} & \frac{16\text{mm}}{}
\end{array}
\end{array} \\
1\text{st sector} & \begin{array}{c} 0, 1.2, 0.5 \end{array} \\
2\text{nd sector} & \begin{array}{c} 0, 1.2, 0 \end{array} \\
8\text{th sector} & \begin{array}{c} 0, 1.2, 0 \end{array}
\end{align*}
\]

where sectors 2 through 8 all deliver the same radiation from each collimator. Since in PFX each sector in a shot can be opened with only one size (4mm, 8mm, or 16mm) at a time, this plan cannot be delivered as one shot. One shot keeps all sectors open with size 8mm for 1.2min at isocentre (250, 230, 90) and the other shot keeps sector 1 open for 0.5min with size 16mm at the same isocentre. The radiation beam-on time of this delivery plan is 1.2 + 0.5 = 1.7min. On the other hand, one may make nine shots by opening only one sector with only one size at a time. Then the beam-on time is 8 × 1.2 + 0.5 = 10.10min, which is, in fact, the upper bound of the beam-on time of all shot combinations for this treatment scenario.

Define the beam-on time, \( \tau_{\text{rad}} \), to be the shortest radiation time amongst all the possible delivery strategies. To calculate the beam-on time, assume the treatment plan has only one isocentre. The duration time is represented as follows:

\[
t = \begin{bmatrix}
t_{11} & t_{12} & \cdots & t_{18} \\
t_{21} & t_{22} & \cdots & t_{28} \\
t_{31} & t_{32} & \cdots & t_{38}
\end{bmatrix}
\]

(4.2)

where each column represents durations of each sector and each row represents a collimator size. A sector is closed if every element of its column is zero. Since each sector cannot deliver two different sizes at the same time,

\[
\tau_{\text{rad}} \geq \sum_{e \in C} t_{bc} \quad \forall b \in B
\]

(4.3)
and hence
\[ \tau_{\text{rad}} \geq \max_{b \in B} \left\{ \sum_{c \in C} t_{bc} \right\} \] (4.4)
Since \( \tau_{\text{rad}} \) is the shortest delivery time, we have
\[ \tau_{\text{rad}} = \max_{b \in B} \left\{ \sum_{c \in C} t_{bc} \right\} = \|t\|_1 \] (4.5)
where \( \| \cdot \|_1 \) is the matrix 1-norm.

As each isocentre in a treatment plan is delivered separately, in the general case, the beam-on time is calculated as
\[ \tau_{\text{rad}} = \sum_{I \in \Theta} \left( \max_{b \in B} \left\{ \sum_{c \in C} t_{Ibc} \right\} \right) = \sum_{I \in \Theta} \|t_I\|_1 \] (4.6)
where \( t_I \) is the duration time matrix, similar to the one in 4.2 at isocentre \( I \). This notation definition of BOT is important because it defines the BOT as a convex function of duration time.

### 4.3 Incorporation of BOT in SDO

As discussed in Chapter 2, treatment plans generated by inverse planning may suffer from high beam-on time. Consider the fact that the SDO is a highly degenerate problem where the optimal solution forms an infinite set. Therefore, different algorithms might generate different solutions with different BOT. In our experience, GP algorithms generate plans with lower BOT than plans generated by IPCG or MOSEK when initialized with an all-zero vector. In order to reduce BOT in the final plan, independent of the solver employed, a penalty term on the beam duration \( H(t) \) is added to the objective of the SDO model. The sector duration optimization with beam-on time consideration is then
\[
\begin{align*}
\text{minimize} & \quad \sum_{s \in S \cup T} \sum_{j \in V_s} F_s(z_j) + \omega H(t) \\
\text{subject to} & \quad z_j = \sum_{I \in \Theta} \sum_{b \in B} \sum_{c \in C} D_{Ibcj} t_{Ibc} \\
& \quad 0 \leq t_{Ibc} \leq t_{\text{MAX}}
\end{align*}
\] (SDO-BOT)

\[ I \in \Theta, \ b \in B, \ c \in C \]
where $F_s$ has the same definition as $\text{Equation 2.2}$, $t$ is the vector of all $t_{Ibc}$’s and $H(t)$ is any of the following penalty functions:

\[
H_1(t) = \sum_{I \in \Theta} \sum_{b \in B} \sum_{c \in C} t_{Ibc} \quad \text{\textit{(BOT-Lin)}}
\]

\[
H_2(t) = \sum_{I \in \Theta} \sum_{b \in B} \sum_{c \in C} t_{Ibc}^2 = \sum_{I \in \Theta} \|t_I\|_F^2 \quad \text{\textit{(BOT-Quad)}}
\]

\[
H_3(t) = H_1(t) + H_2(t) \quad \text{\textit{(BOT-LinQuad)}}
\]

\[
H_4(t) = \sum_{I \in \Theta} \left( \max_{b \in B} \left\{ \sum_{c \in C} t_{Ibc} \right\} \right) = \tau_{rad} \quad \text{\textit{(BOT-MaxSector)}}
\]

where $\|\cdot\|_F$ is the Frobenius norm. Since all of the above penalty functions are convex, the SDO-BOT problem is a convex optimization problem. Any of $H_1$, $H_2$, and $H_3$ functions dominates the beam-on time $\tau_{rad}$ and may be used to force down BOT, while $H_4$ is exactly $\tau_{rad}$. Figure 4.1 visualizes each penalty function on a two-dimensional domain. In the SDO-BOT objective, $H(t)$ is penalized by $\omega$ to balance BOT with treatment quality.

\textbf{BOT-Lin} is the possible radiation time one can introduce by letting each shot consist of opening one sector with one size opened at a time. Hence $\tau_{rad} \leq H_1(t)$ can be derived from the plan beam-on time. We use penalty function $H_1$ when linearity of the objective function is important. Using the fundamental properties of matrix norms, for any given duration time matrix, $t_I$ we have

\[
\tau_{rad} = \sum_{I \in \Theta} \|t_I\|_1 \leq 3 \sum_{I \in \Theta} \|t_I\|_F = 3H_2(t)
\]

that is, we can also reduce the beam-on time by penalizing $H_2(t)$. Compared to $\text{BOT-Lin}$, $\text{BOT-Quad}$ has a bigger reduction on duration times exceeding 1 minute. The sum of $H_1$ and $H_2$, $H_3$ is also used to put more pressure on reducing BOT when needed. The last penalty function, $H_4(t)$, is to penalize the actual plan BOT. Non-differentiability is the only drawback of $H_4(t)$ when using a gradient-based algorithm to solve SDO-BOT. However, it is not a problem when using IPCG, which does not require differentiability of the objective.
4.4 Incorporation of BOT in **SDIO**

Due to the complexity of the **SDIO** model, it is very computationally expensive to employ similar penalty functions as we used in **SDO-BOT**. However, by reducing $t_{\text{MAX}}$ in the $t_{Ibc}$ constraints, the BOT can be reduced significantly. Note that reducing $t_{\text{MAX}}$ is equivalent to reducing the matrix max-norm $\|t_I\|_{\text{max}}$ for the given duration matrix $t_I$ at isocentre $I$. Using the matrix-norm inequality

$$\frac{1}{\sqrt{24}}\|t\|_{\text{max}} \leq \tau_{\text{rad}} \leq 6\sqrt{2}\|t\|_{\text{max}}$$

we can see that reducing $t_{\text{MAX}}$ results in reducing the beam-on time.

\footnote{The general case of the inequality is $\frac{1}{\sqrt{mn}}\|A\|_{\text{max}} \leq \frac{1}{\sqrt{m}}\|A\|_1 \leq \sqrt{mn}\|A\|_{\text{max}}$, where $A$ is an $m \times n$ matrix.}
4.5 Results

The SDO-BOT model is evaluated for a representative clinical case, Case 1, for all four BOT penalty functions presented in previous section. Case 1 is also used to demonstrate the effect of $t_{\text{MAX}}$ and the number of isocentres in SDIO on BOT.

4.5.1 Numerical results

The IPCG algorithm is used to solve SDO-BOT for a pre-selected set of 25 isocentres for Case 1. The computation time for IPCG to solve SDO-BOT for each of the four penalty functions is 10 to 12 minutes. The algorithm stops when the duality gap of 5% is achieved. Gurobi 2.0 is used to solve the SDIO model with a fixed value of 10sec for $t_{\text{MIN}}$ and different values for $t_{\text{MAX}}$. The solution time varies from 8min to 600min with positive correlation to the changes on $t_{\text{MAX}}$ (see Table 4.2). The best solution time of 8min was achieved with $t_{\text{MIN}} = 10$sec and $t_{\text{MAX}} = 20$sec when there is no penalty on the number of isocentres, i.e., $\beta = 0$. The Gurobi solver is set to stop when the MIPgap is smaller than 10% or computation time exceeds 600min.

4.5.2 Clinical results

Figure 4.2 shows the effects of different penalty functions on the quality of the plans. Paddick and classic conformity indices, maximum brainstem dose, and BOT are compared for each scenario by using different weights on the BOT penalty functions. The value for each metric in the clinical treatment plan for the case is also presented in each figure by a dashed line. Figure 4.2a shows that the BOT can be reduced to under an hour (from 3 hours) but it never reaches the clinical BOT ($\sim32$min). The conformity indices worsened as BOT improved. Not surprisingly, the BOT reduces faster when we use quadratic penalty term rather than the linear term (Figure 4.2b). But, the conformity indices go beyond the allowable clinical limits. The maximum brainstem dose also
reaches the maximum dose limit of 15Gy. Figure 4.2c also shows that the \text{BOT-LinQuad} penalty function decreases the radiation time much faster than the linear and quadratic penalties, but has the worst degradation in the quality of the plans. The best balance between BOT and quality measures is made by the \text{BOT-MaxSector} function. Figure 4.2d shows that the beam-on time can be reduced to almost the clinical treatment while the conformity indices and the maximum brainstem dose stay close to the inverse plan compared to the other penalty functions.

Table 4.1 and 4.2 present the results of solving SDIO by changing $t_{\text{MIN}}$ and $t_{\text{MAX}}$. The first column contains the values of $t_{\text{MIN}}$ and $t_{\text{MAX}}$ within the brackets. The quality of the plans is compared to the clinical plans, and better values are in bold. The second column, $\beta$, is the penalty weight on the number of isocentres in the SDIO objective. Number of isocentres selected in the final plan, Paddick and Classic conformity indices, maximum dose to 1mm$^3$ of brainstem, BOT, solution time, and MIP duality gap are compared.

Table 4.1 shows that the plans generated by the SDIO model are of better quality compared to the clinical plans. For the plans presented in this table, $t_{\text{MAX}}$ is fixed and we are only concern about the effect of $t_{\text{MIN}}$ on the quality of the plans. It is clear from the table that increasing the minimum duration for each beamlet causes less isocentres to be selected by the solver to prevent overdosing the target and OARs. Increasing $t_{\text{MIN}}$ also helps to reduce the beam-on time as more dose per minute is delivered to the target by forcing all sectors to be open at the same time. To see the effect of having non-zero values for $t_{\text{MIN}}$ on the number of isocentres, the penalty term on the number of isocentres, in SDIO is removed by setting $\beta = 0$.

Table 4.2 presents the effect of reducing $t_{\text{MAX}}$ on the quality of the plans and BOT while $t_{\text{MIN}}$ is fixed to 10sec. The results show that the beam-on time can be reduced dramatically by reducing $t_{\text{MAX}}$ while the conformity indices and the max brainstem dose stay above clinical plans. $t_{\text{MIN}}$ also plays a role in reducing BOT. By setting $t_{\text{MIN}} = 10$, we forced the model to generate plans for which all open sectors must be open for a
Chapter 4. Methods to control beam-on time

Figure 4.2: Comparison of plan quality as function of BOT penalty term $\omega$ in SDO-BOT for representative Case 1. Dashed lines represent clinical plan values.
Table 4.1: The effect of $t_{\text{MIN}}$ on treatment plan quality and BOT in SDIO for representative Case 1. Bold values indicates improvement over the clinical plan.

| $[t_{\text{MIN}}, t_{\text{MAX}}]$ | $\beta$ | Iso | $\beta_{\text{CI Paddick}}$ | $\beta_{\text{CI Classic}}$ | BST max (Gy) | BOT (min) | Soln time (min) | MIP gap (%) |
|---|---|---|---|---|---|---|---|---|---|
| [0.0, 60] | 0 | 100 | 0.94 | 1.04 | 13.30 | 432 | 4 | 0.00 |
| [1.2, 60] | 0 | 57 | 0.93 | 1.06 | 13.10 | 107 | 8 | 1.50 |
| [1.8, 60] | 0 | 38 | 0.92 | 1.07 | 13.30 | 79 | 9 | 3.45 |
| [3.0, 60] | 0 | 30 | 0.92 | 1.08 | 13.10 | 65 | 10 | 5.10 |
| [4.2, 60] | 0 | 43 | 0.92 | 1.07 | 13.40 | 86 | 10 | 0.76 |
| [4.8, 60] | 0 | 32 | 0.91 | 1.09 | 13.30 | 70 | 11 | 2.60 |
| [6.0, 60] | 0 | 32 | 0.91 | 1.08 | 13.10 | 71 | 10 | 2.65 |
| [7.2, 60] | 0 | 27 | 0.91 | 1.09 | 13.20 | 62 | 11 | 2.36 |
| [7.8, 60] | 0 | 24 | 0.90 | 1.09 | 13.10 | 55 | 8 | 4.01 |
| [9.0, 60] | 0 | 25 | 0.91 | 1.09 | 13.00 | 56 | 9 | 3.42 |
| [10.2, 60] | 0 | 22 | 0.90 | 1.10 | 12.90 | 53 | 8 | 5.31 |
significant amount of time, which results in exposing more radiation per unit of time. Conversely, more isocentres are needed to overcome the underdose introduced by reducing the maximum sector durations. Another observation from Table 4.2 is that larger $t_{\text{MAX}}$ values need more computation time.

### 4.6 Conclusions

The beam-on time can be controlled in both SDO and SDIO models. Amongst four presented penalty functions on duration time, the actual beam-on time function introduces better balance between quality of the plans and the BOT. In the SDIO model, since the isocentres are not fixed, tightening duration limits from both below and above can reduce the beam-on time while maintaining the quality of the plans. The combination of reducing the number of isocentres and penalizing the exact beam-time (rather than upper bounds) shows that we can generate treatment plans that have high quality and low BOT.
Table 4.2: The effect of $t_{\text{MAX}}$ and $\beta$ on treatment plan quality and BOT in SDIO. Bold values indicates improvement over the clinical plan.

<table>
<thead>
<tr>
<th>$[t_{\text{MIN}}, t_{\text{MAX}}]$ (s)</th>
<th>$\beta$</th>
<th>#</th>
<th>$\text{CI}_{\text{Paddick}}$</th>
<th>$\text{CI}_{\text{Classic}}$</th>
<th>BST max (Gy)</th>
<th>BOT (min)</th>
<th>Soln time (min)</th>
<th>MIP gap (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$[10, 20]$</td>
<td>0</td>
<td>41</td>
<td>0.87</td>
<td>1.12</td>
<td>12.7</td>
<td>33</td>
<td>59</td>
<td>0.48</td>
</tr>
<tr>
<td>$[10, 30]$</td>
<td>0</td>
<td>30</td>
<td>0.88</td>
<td>1.11</td>
<td>12.9</td>
<td>40</td>
<td>8</td>
<td>2.93</td>
</tr>
<tr>
<td>$[10, 60]$</td>
<td>0</td>
<td>22</td>
<td>0.90</td>
<td>1.10</td>
<td>12.9</td>
<td>53</td>
<td>8</td>
<td>5.31</td>
</tr>
<tr>
<td>$[10, 120]$</td>
<td>0</td>
<td>13</td>
<td>0.87</td>
<td>1.12</td>
<td>13.1</td>
<td>54</td>
<td>13</td>
<td>9.85</td>
</tr>
<tr>
<td>$[10, 240]$</td>
<td>0</td>
<td>15</td>
<td>0.88</td>
<td>1.12</td>
<td>13.1</td>
<td>86</td>
<td>26</td>
<td>9.68</td>
</tr>
<tr>
<td>$[10, 360]$</td>
<td>0</td>
<td>15</td>
<td>0.89</td>
<td>1.11</td>
<td>13.4</td>
<td>108</td>
<td>69</td>
<td>9.18</td>
</tr>
<tr>
<td>$[10, 600]$</td>
<td>0</td>
<td>20</td>
<td>0.90</td>
<td>1.10</td>
<td>13.4</td>
<td>125</td>
<td>92</td>
<td>9.44</td>
</tr>
<tr>
<td>$[10, 20]$</td>
<td>5</td>
<td>36</td>
<td>0.86</td>
<td>1.13</td>
<td>12.7</td>
<td>31</td>
<td>56</td>
<td>1.23</td>
</tr>
<tr>
<td>$[10, 30]$</td>
<td>5</td>
<td>34</td>
<td>0.88</td>
<td>1.11</td>
<td>12.7</td>
<td>34</td>
<td>65</td>
<td>1.70</td>
</tr>
<tr>
<td>$[10, 60]$</td>
<td>5</td>
<td>20</td>
<td>0.89</td>
<td>1.10</td>
<td>13.2</td>
<td>49</td>
<td>23</td>
<td>3.26</td>
</tr>
<tr>
<td>$[10, 120]$</td>
<td>5</td>
<td>17</td>
<td>0.91</td>
<td>1.08</td>
<td>13.4</td>
<td>73</td>
<td>86</td>
<td>3.16</td>
</tr>
<tr>
<td>$[10, 240]$</td>
<td>5</td>
<td>19</td>
<td>0.89</td>
<td>1.10</td>
<td>13.2</td>
<td>94</td>
<td>212</td>
<td>8.03</td>
</tr>
<tr>
<td>$[10, 360]$</td>
<td>5</td>
<td>21</td>
<td>0.90</td>
<td>1.10</td>
<td>13.7</td>
<td>142</td>
<td>219</td>
<td>9.70</td>
</tr>
<tr>
<td>$[10, 600]$</td>
<td>5</td>
<td>16</td>
<td>0.90</td>
<td>1.10</td>
<td>13.6</td>
<td>103</td>
<td>600</td>
<td>11.01</td>
</tr>
<tr>
<td>$[10, 20]$</td>
<td>10</td>
<td>33</td>
<td>0.85</td>
<td>1.15</td>
<td>12.8</td>
<td>29</td>
<td>56</td>
<td>1.65</td>
</tr>
<tr>
<td>$[10, 30]$</td>
<td>10</td>
<td>23</td>
<td>0.87</td>
<td>1.12</td>
<td>12.6</td>
<td>33</td>
<td>58</td>
<td>2.73</td>
</tr>
<tr>
<td>$[10, 60]$</td>
<td>10</td>
<td>14</td>
<td>0.88</td>
<td>1.11</td>
<td>12.8</td>
<td>41</td>
<td>71</td>
<td>3.04</td>
</tr>
<tr>
<td>$[10, 120]$</td>
<td>10</td>
<td>11</td>
<td>0.88</td>
<td>1.12</td>
<td>13.4</td>
<td>56</td>
<td>92</td>
<td>6.19</td>
</tr>
<tr>
<td>$[10, 240]$</td>
<td>10</td>
<td>20</td>
<td>0.91</td>
<td>1.08</td>
<td>13.2</td>
<td>106</td>
<td>154</td>
<td>9.72</td>
</tr>
<tr>
<td>$[10, 360]$</td>
<td>10</td>
<td>18</td>
<td>0.91</td>
<td>1.08</td>
<td>13.3</td>
<td>122</td>
<td>600</td>
<td>10.07</td>
</tr>
<tr>
<td>$[10, 600]$</td>
<td>10</td>
<td>10</td>
<td>0.82</td>
<td>1.19</td>
<td>13.0</td>
<td>81</td>
<td>600</td>
<td>16.43</td>
</tr>
</tbody>
</table>
Chapter 5

Interior-point constraint generation method

In order to be consistent with the published paper [62] that originated from materials presented in this chapter, we keep the notation similar to that appearing in [62], and independent of the notation used in the previous chapters.

5.1 Introduction

Consider the following SILO problem:

$$\max \{ b^\top y : a_\omega^\top y \leq c_\omega, \ \omega \in \Omega \} ,$$

(5.1)

where $\Omega$ is a compact set, $b \in \mathbb{R}^m$, $a_\omega \in \mathbb{R}^m$, and $c_\omega \in \mathbb{R}^1$, for $\omega \in \Omega$. This problem has been well-studied in the literature and has numerous applications in engineering, healthcare and management. See [36] for a theoretical survey and [51] for a more recent survey on this topic.

We propose a constraint generation build-up technique to solve problem (5.1), and show that a finite number of iterations is required to obtain an $\varepsilon$-optimal solution. The idea of this principle is that if a model is too large, or a function is too complex, it is better
not to know it entirely, but to discover it as needed by intelligent queries (questions). Let
\[ F_\Omega = \{ y \in \mathbb{R}^m : a_\omega^\top y \leq c_\omega, \ \omega \in \Omega \}, \]
be the feasible region of problem (5.1), assumed to be compact. Consider a discretization of problem (5.1), where the feasible region is an outer approximation of \( F_\Omega \), and is defined by a finite number of constraints:
\[
\max \{ b^\top y : A^\top y \leq c \}, \quad (5.2)
\]
where \( A \in \mathbb{R}^{m \times n} \) is a full row rank matrix composed of column vectors \( a_i \)'s and \( c \in \mathbb{R}^n \) is composed of scalars \( c_i \)'s. Problem (5.2) is a relaxation of the original problem. Let us call it the dual problem. The corresponding primal problem reads
\[
\min \{ c^\top x : Ax = b, x \geq 0 \}. \quad (5.3)
\]
The main idea of our algorithm is as follows: We start from problem (5.2) with only an artificial box constraint whose bounds (RHS) are dynamically updated. Using a point in the vicinity of the central path of problem (5.2), multiple violated (deep) constraints from \( F_\Omega \) are identified. The feasible region of the dual problem is updated by adding the violated constraints and the barrier function is simultaneously updated by reducing the barrier parameter. This is equivalent to adding columns to primal problem (5.3). Then the strict feasibility for the new feasible region is recovered and the central path is updated. This process continues until the barrier parameter is small enough, i.e., the duality gap approaches to zero.

We derive two theoretical complexity results. After adding \( p \) violated constraints and simultaneously updating the centring parameter \( \mu \) by \( \mu^+ = (1-\eta)\mu \) where \( \eta = \frac{1}{9\sqrt{2m}} \), we show that only \( O(p \log(p+1)) \) Newton steps are required to obtain a point in the vicinity of the new \( \mu^+ \)-centre. We also show that our IPCG algorithm stops with an \( \epsilon \)-solution to the SILO problem after adding at most
\[
O \left( \frac{m^2 p^2}{\delta^2} e^{3\sqrt{m}/\epsilon} \right)
\]
constraints, where \( \delta \) is the radius of the largest full dimensional ball contained in \( \mathcal{F}_\Omega \), and \( \hat{p} \) is the maximum number of constraints added simultaneously.

Our build-up approach is in the spirit of the analysis centre cutting plane methods (ACCPM), introduced by [68], and further developed by [39], [40], [54], and [81]. The theoretical results of the above-mentioned works are all in the context of convex feasibility problem with finite number of constraints.

We implement our IPCG algorithm to solve SDO-SILO. Since the IPCG algorithm requires a bounded feasible region, the algorithm is initiated with the nonnegativity constraints and artificial but dynamic bound constraints on the decision variables, the number of which depends on the number of isocentres. When a query point approaches one of the artificial bounds, the bound is moved away to increase flexibility. At each iteration, an oracle is called to return an outer approximation of the piecewise quadratic constraint. This is obtained by computing the gradient of the constraint function (the “if” function) at the current \( \mu \)-centre. This procedure is continued until the duality gap falls within the desired accuracy as detailed in the earlier sections.

It is worth mentioning that the violated constraint in SDO-SOCO does not violate the feasibility of the original problem. The feasibility of the original problem SDO is established by nonnegativity of the time of radiation delivery, which is satisfied in all iterations of SDO-SOCO. In other words, generating constraints for this model does not improve feasibility of the original problem, but improves its optimality.

Using real patient data we show that our IPCG algorithm can efficiently handle real-life large-scale healthcare applications, see 5.1. Furthermore, we develop an equivalent SOCO model of the sector duration optimization problem by adding more variables and linear constraints. Comparing our results with that of MOSEK when applied to the SOCO model, we show that IPCG algorithm outperforms commercial software packages that use the classical primal-dual interior point methods both in accuracy and time. We also compare our results with that of a gradient projection method and show that in
both cases IPCG algorithm obtains a more accurate solution substantially faster.

The rest of the chapter is organized as follows: Section 5.2 presents preliminaries and some technical lemmas that are needed throughout this chapter. In Section 5.3, we describe our IPCG algorithm in detail. Complexity of recovering the \( \mu \)-centre and complexity and convergence of the algorithm are given in Sections 5.4 and 5.5, respectively. Section 5.6 presents our computational experience with the SDO model and compare our results with that of the classical primal-dual interior point methods using MOSEK, and a gradient projection method.

5.2 Preliminaries

We denote the primal, dual and primal-dual feasible regions of the discretization problem by \( F_p \), \( F_d \), and \( F \), respectively:

\[
F_p = \{ x \in \mathbb{R}^n : Ax = b, x \geq 0 \},
\]

\[
F_d = \{ s \in \mathbb{R}^n : A^\top y + s = c, s \geq 0 \},
\]

\[
F = F_p \times F_d.
\]

Let \( \mu > 0 \) be the barrier parameter. The corresponding barrier functions read as

\[
\varphi_p(x, \mu) := \frac{\mathbf{c}^\top x}{\mu} - \sum_{i=1}^{n} \log x_i,
\]

\[
\varphi_d(s, \mu) := -\frac{\mathbf{b}^\top y}{\mu} + \sum_{i=1}^{n} \log s_i,
\]

\[
\varphi(x, s, \mu) := \frac{x^\top s}{\mu} - \sum_{i=1}^{n} \log x_i s_i.
\]

Due to the one-to-one correspondence between \( y \) and \( s \) in \( F_d \), we drop the argument \( y \) from the barrier function. The unique minimizer of \( \varphi(x, s, \mu) \) over \( F \), denoted by \( (x(\mu), s(\mu)) \), is a point on the central path, and it satisfies the primal-dual feasibility and the centring condition \( x s = \mu e \), where \( x s \) is the Hadamard product of \( x \) and \( s \), i.e., it is
an $n$-vector composed of $x_is_i$, and $e \in \mathbb{R}^n$ is the vector with all its components equal to 1. We also call this point the $\mu$-centre. For the $\mu$-centre $(x(\mu), s(\mu))$, one has

$$\varphi(x(\mu), s(\mu), \mu) = n - n \log \mu.$$  

A $\theta$-approximate $\mu$-centre $(\bar{x}, \bar{s})$ is a point in the vicinity of the central path that satisfies

$$A^\top \bar{y} + \bar{s} = c, \quad A\bar{x} = b, \quad \left\| \frac{\bar{x} \bar{s}}{\mu} - e \right\| \leq \theta < 1. \quad (5.4)$$

We now state some technical lemmas that are needed later in this chapter. The proofs of those lemmas that are not given here can be found in interior point methods books, such as [23], [65], and [80].

**Lemma 5.2.1.** Let $z \in \mathbb{R}^n$, and $\|z\| < 1$. Then

$$\phi(\|z\|) \leq \psi(z) \leq \phi(-\|z\|),$$

where $\psi(z) = e^\top z - \sum_{j=1}^n \log(1 + z_j)$, and $\phi(\alpha) = \alpha - \log(1 + \alpha)$.

**Lemma 5.2.2.** If $z \in \mathbb{R}^n$ and $\|z\|_\infty < 1$, then

$$e^\top z - \frac{\|z\|^2}{2(1 - \|z\|_\infty)} \leq \sum_{j=1}^n \log(1 + z_j) \leq e^\top z.$$ 

**Lemma 5.2.3.** Let $(\bar{x}, \bar{s})$ be a $\theta$-approximate $\mu$-centre. Then

$$n - \theta \sqrt{n} \leq \frac{\bar{x}^\top \bar{s}}{\mu} \leq n + \theta \sqrt{n}.$$ 

Moreover if $\mu^+ = (1 - \eta)\mu$ with $0 < \eta < 1$, then

$$\left\| \frac{\bar{x} \bar{s}}{\mu^+} - e \right\| \leq \frac{1}{1 - \eta} (\theta + \eta \sqrt{n}).$$

**Corollary 5.2.4.** For $n \geq 2$, $\eta = \frac{1}{9\sqrt{n}}$, and arbitrary $\theta \leq 1/4$, one has

$$\left\| \frac{\bar{x} \bar{s}}{\mu^+} - e \right\| \leq 0.40.$$
5.3 Interior point constraint generation algorithm

In this section we present our IPCG algorithm for solving problem \((5.1)\). We make the following assumptions:

**Assumption 1.** The set \(\Omega\) is compact, and the mappings \(t \rightarrow a_t\) and \(t \rightarrow c_t\) are continuous in \(t\).

**Assumption 2.** The feasible region \(\mathcal{F}_\Omega\) contains a \(\delta\)-radius full dimensional ball.

**Assumption 3.** \(\mathcal{F}_\Omega\) is contained in the unit cube \([0,1]^m\), and all \(m\)-vectors \(b\) and \(a_t\) are normalized.

Assumption 1 is made to ensure that the optimal solution of the constraint generation algorithm coincides with that of problem \((5.1)\) (see Lemma 5.3.2). Assumption 2 is needed to establish a bound on the number of constraints, and Assumption 3 is a scaling assumption that will help to keep the complexity bound simple.

We now describe the algorithm. Let \(\bar{y}\) be a point in the vicinity of the central path of \(\mathcal{F}_d\) (see \((5.4)\)) and \(\bar{a}_j^\top y \leq \bar{c}_j\), for \(j = 1,\ldots,p\) be \(p\) constraints in \(\mathcal{F}_\Omega\) such that \(\bar{c}_j < \bar{a}_j^\top \bar{y}\). The feasible region of the updated discretization therefore reads as

\[
\mathcal{F}_d^+ = \{ s \in \mathbb{R}_+^n, r \in \mathbb{R}_+^p : A^\top y + s = c, A^\top y + r = \tilde{c} \},
\]

where \(\tilde{A} \in \mathbb{R}^{m \times p}\) is composed of the \(p\) column vectors \(\bar{a}_i\)’s and \(\tilde{c} = (\bar{c}_1; \ldots; \bar{c}_p)\). Let \(\mu^+ = (1 - \eta)\mu \) be the updated barrier parameter for a later-specified value \(0 < \eta < 1\). The task is now to find a point in the vicinity of the central path of the updated discretization, close to the \(\mu^+\)-centre of \(\mathcal{F}_d^+\). However, since \(\bar{c} < \tilde{A}^\top \bar{y}\), then \(\tilde{A}^\top y \leq \bar{c}\) are deep constraints for \(\mathcal{F}_d\), the current point \(\bar{y}\) is not a feasible point of \(\mathcal{F}_d^+\). Therefore we first need to derive a strictly feasible point for \(\mathcal{F}_d^+\). Let

\[
\bar{t} = \arg \min \left\{ \frac{p}{2} t^\top V t - \sum_{i=1}^p \log t_i \right\},
\]

(5.5)
where \( V = \bar{A}^\top (A\bar{X}^2A^\top)^{-1}\bar{A} \), where \( X \) is a diagonal \( n \times n \) matrix with the components of vector \( x \) as its diagonal elements. Also define

\[
\bar{d} = p(\bar{A}^\top \bar{y} - \bar{c})\bar{t}.
\]  

(5.6)

Notice that since \( \bar{A}^\top \bar{y} - \bar{c} > 0 \), and \( \bar{t} > 0 \), then \( \bar{d} > 0 \). Let \( \alpha < 1 - \theta \) be fixed. We consider two cases:

1. **Moderately deep constraints**: \( \bar{d} < \alpha e \). In this case we show that all violated constraints cross the Dikin ellipsoid around \( \bar{y} \), and the dual feasibility can be recovered using the current point \( \bar{y} \).

2. **Very deep constraints**: There exists a constraint for which \( \bar{d}_i \geq \alpha \). In this case, dual feasibility cannot be recovered. We show that one can recover feasibility in the primal space

\[
\mathcal{F}_p^+ = \{ x \in \mathbb{R}^n_+, t \in \mathbb{R}^p_+ : Ax + \bar{A}t = b \},
\]

and obtain the new \( \mu^+ \)-centre using the primal barrier function.

The concept of shallow and deep cuts was first introduced in the context of analytic centre cutting plane method by \[37\].

**Lemma 5.3.1.** Let \( \mathcal{F}_p \) and \( \mathcal{F}_d \) be the primal and dual feasible regions of the discretization problem, respectively. Let \( \mu \) be the barrier parameter, and \( (\bar{x}, \bar{s}) \) be a point in the vicinity of the central path that satisfies (5.4). Let \( p \) violated constraints \( \bar{A}^\top y \leq \bar{c} \) be added to \( \mathcal{F}_d \), \( \Delta x = -\bar{X}^2 A^\top (A\bar{X}^2 A^\top)^{-1}\bar{A}t \), and \( \bar{d}_i < \alpha < 1 - \theta \), for \( i = 1, \ldots, p \). Then \( x^+ = (\bar{x} + \alpha \Delta x; \alpha \bar{t}) \) is strictly feasible for \( \mathcal{F}_p^+ \). Furthermore, define \( \Delta s = A^\top (A\bar{X}^2 A^\top)^{-1}\bar{A}t \), and

\[
\bar{r} = \frac{1}{p}(\alpha e - \bar{d})\bar{t}^{-1},
\]  

(5.7)

where the \( p \)-vector \( \bar{t}^{-1} \) is the component-wise inverse of vector \( \bar{t} \). Then \( s^+ = (\bar{s} + \alpha \Delta s; \bar{r}) \) is strictly feasible for \( \mathcal{F}_d^+ \).
**Proof:** A similar lemma is presented in [38] for multiple cutting plane algorithm where \( \mu = 1 \) and \( \bar{d} = 0 \). The directions \( \Delta x \) and \( \Delta s \) defined in this lemma are similar to those of [38]. Therefore, to some extent, their proof remains valid here. In particular, \( A(\Delta x) + \bar{A} \bar{t} = 0 \) is obtained by construction. Also, the strict feasibility of the updating directions \( \bar{x} + \alpha \Delta x > 0 \) and \( \bar{s} + \alpha \Delta s > 0 \) are obtained by Lemma 5.4.1 below, and by the fact that \( \alpha < 1 - \theta \).

We prove that \( \bar{A}^\top y^+ + \bar{r} = \bar{c} \) and \( \bar{r} > 0 \). Notice that \( A^\top (\bar{y} + \Delta y) + \bar{s} + \Delta s = c \), implies \( A^\top \Delta y = -\Delta s \) and \( \Delta y = -(A \bar{X}^2 A^\top)^{-1} \bar{A} \bar{t} \). Therefore

\[
\bar{A}^\top y^+ + \bar{r} = \bar{A}^\top \bar{y} + \alpha \bar{A}^\top \Delta y + \bar{r} = \bar{A}^\top \bar{y} - \alpha V \bar{t} + \bar{r},
\]

and from the KKT optimality conditions of problem (5.5), we have

\[
\bar{A}^\top y^+ + \bar{r} = \bar{A}^\top \bar{y} - \frac{\alpha}{p} \bar{t}^{-1} + \frac{1}{p} (\alpha e - \bar{d}) \bar{t}^{-1}
\]
\[
= \bar{A}^\top \bar{y} - \frac{1}{p} \bar{d} \bar{t}^{-1}
\]
\[
= \bar{c}.
\]

Now since \( \bar{d} < \alpha e \), we have \( \bar{r} > 0 \). \( \square \)

Lemma 5.3.1 shows that if the violated constraints are moderately deep, then Newton’s method can be initiated from \( x^+ \) and \( s^+ \) to obtain a point in the vicinity of the new central path. In the next section we derive a bound on the number of Newton steps required to update the \( \mu^+ \)-centre.

When there is at least one very deep inequality, dual feasibility cannot be recovered because it is not clear how far the constraint is away from the Dikin ellipsoid. In this situation one can still recover primal feasibility by using \( x^+ \) and Newton’s method can be applied in the primal space to update the \( \mu^+ \)-centre. This procedure is repeated until the barrier parameter \( \mu \) falls within the desired accuracy.

The next lemma, due to [55], shows that the constraint generation algorithm delivers an \( \varepsilon \)-optimal solution for problem (5.1).
Lemma 5.3.2. Let $\varepsilon > 0$ be given. Under Assumption 1, if $\bar{y} \in F_{\Omega}$ is in the vicinity of $\mu < \frac{\varepsilon}{n + \sqrt{n}}$, then $\bar{y}$ is an $\varepsilon$-maximizer of problem (5.1).

We now formally present our algorithm in Figure 2. A visualization of IPCG steps is presented in Figure 5.1. In the illustrated example, in Figure 5.1a, IPCG starts with a box as an approximation of the original feasible set (ellipse) with an initial solution within the box. In Figure 5.1b, a violating cut is found by an oracle if the current solution is not feasible for the original problem, then the current feasible set is updated (oracle step). In Figure 5.1c, the new feasible solution is found (recovery step). The $\mu$-centre is found in Figure 5.1d, and $\mu$ is updated and the new $\mu$-centre found until infeasibility occurs (Figure 5.1e). Finally, a new violating cut is found with the oracle (Figure 5.1f) and the process repeats.

5.4 Complexity of recovering the $\mu$-centre

In this section we derive a bound on the number of Newton steps that is required to obtain a point in the vicinity of the $\mu^+$-centre when all violating constraints are moderately deep.

First, we recall a lemma from [38]:

Lemma 5.4.1. For directions $\Delta x$ and $\Delta s$ in Lemma 5.3.1, we have

$$\|X^{-1}\Delta x\| \leq \frac{1}{1 - \theta} \quad \text{and} \quad \|S^{-1}\Delta s\| \leq \frac{1}{1 - \theta}.$$ 

We also need the following technical lemma:

Lemma 5.4.2. Let $(\bar{x}, \bar{s})$ be a $\theta$-approximate $\mu$-centre. Then

1. $\varphi(\bar{x}, \bar{s}, \mu) = \varphi(x(\mu), s(\mu), \mu) + \frac{\theta^2}{2(1 - \theta)}$

2. $\varphi(\bar{x}, \bar{s}, \mu^+) + n \log(1 - \eta) \leq \varphi(x(\mu), s(\mu), \mu) + \nu(n, \eta, \theta)$

3. $\varphi(\bar{x}, \bar{s}, \mu^+) \leq \varphi(x(\mu^+), s(\mu^+), \mu^+) + \nu(n, \eta, \theta)$
**Algorithm 2** Interior-point constraint generation

1: $\mathcal{F}_0^\circ = [0, 1]^m$, $\mu_0 = 1$, $y^0 = \frac{1}{2} e$, $s^0 = \frac{1}{2} e$, $n_0 = 2m$, $\eta_0 = \frac{1}{9\sqrt{2}m}$, $\theta = \frac{1}{4}$ and $k = 1$

2: while $(n_k + \sqrt{n_k})\mu_k \geq \varepsilon$ do

3: identify $p_k$ violated constraints $(\bar{A}^k)^\top y \leq \bar{c}^k$ in $\mathcal{F}_\Omega$ such that $\bar{c}^k > (\bar{A}^k)^\top y^k$

4: update $n_k = n_{k-1} + p_k$, $\eta_k = \frac{1}{9\sqrt{n_k}}$, $\mu_k = (1 - \eta_k)\mu_{k-1}$, $A^k = [A^{k-1} \bar{A}^k]$, and $c^k = (c^{k-1}; \bar{c}^k)$

5: compute $\bar{t}$ from (5.5) and $\bar{d}$ from (5.6)

6: if $\bar{d} < \alpha e$ then

7: use $s^+$ to start a dual Newton procedure to obtain $s^k$

8: define $x^k := x(s^k)$ in the vicinity of the $\mu_k$-centre of $\mathcal{F}_d^k$

9: else

10: use $x^+$ to start a primal Newton procedure to obtain $x^k$

11: define $s^k := s(x^k)$ in the vicinity of the $\mu_k$-centre of $\mathcal{F}_p^k$

12: end if

13: $k = k + 1$.

14: end while

15: return $x^k$ and $s^k$

where

$$\nu(n, \eta, \theta) = n \log(1 - \eta) + \frac{n(\eta + \theta\sqrt{n})}{1 - \eta} + \frac{\theta^2}{2(1 - \theta)}.$$  

**Proof:** In view of Lemma 5.2.2, the first inequality is straightforward. Let us prove the second inequality. Since $(\bar{x}, \bar{s})$ is in the vicinity of the $\mu$-centre, from Lemma 5.2.2 we have

$$\varphi(\bar{x}, \bar{s}, \mu^+) = \frac{\bar{x}^\top \bar{s}}{\mu^+} - n \log \mu - \sum_{i=1}^{n} \log \frac{x_i \bar{s}_i}{\mu}$$

$$\leq \frac{\bar{x}^\top \bar{s}}{\mu^+} - \frac{\bar{x}^\top \bar{s}}{\mu} + n - n \log \mu + \frac{\theta^2}{2(1 - \theta)}.$$
Figure 5.1: The IPCG steps: initialization, oracle to find violated constraints, recovery of feasibility, centring the current point, updating the solution along the central path, then repeat.
The second inequality follows from Corollary 5.2.4 and
\[
\bar{x}^\top \bar{s} - \bar{x}^\top \bar{s} = \frac{\eta x^\top s}{1 - \eta} \leq \frac{\eta(n + \theta \sqrt{n})}{1 - \eta}.
\]

The third inequality implies from the second one. \( \square \)

Notice that the bounds on the primal-dual barrier function in Lemma 5.4.2 are also valid for the primal and the dual barrier functions. The following corollary simplifies these bounds for some given values.

**Corollary 5.4.3.** For \( n \geq 2 \), \( \eta = \frac{1}{9\sqrt{n}} \), and arbitrary \( \theta \leq 1/4 \) we have

1. \( \varphi(\bar{x}, \bar{s}, \mu) \leq \varphi(x(\mu), s(\mu), \mu) + 0.05. \)

2. \( \varphi(\bar{x}, \bar{s}, \mu^+) + n \log(1 - \eta) \leq \varphi(x(\mu), s(\mu), \mu) + 0.10, \)

3. \( \varphi(\bar{x}, \bar{s}, \mu^+) \leq \varphi(x(\mu^+), s(\mu^+), \mu^+) + 0.10. \)

**Proof:** Since \( 0 < \eta < 1 \), we have
\[
\nu(n, \eta, \theta) \leq -n\eta + \frac{\eta(n + \theta \sqrt{n})}{1 - \eta} + \frac{\theta^2}{2(1 - \theta)} = \frac{\theta \eta \sqrt{n} + n\eta^2}{1 - \eta} + \frac{\theta^2}{2(1 - \theta)}.
\]

For \( \eta = \frac{1}{9\sqrt{n}} \) and \( n \geq 2 \), the bound simplifies as follows
\[
\nu(n, \eta, \theta) \leq \frac{12}{99}(\theta + 1/9) + \frac{\theta^2}{2(1 - \theta)}.
\]

The proof follows by the assumption \( \theta \leq 0.25. \) \( \square \)

Now we establish an upper bound on the primal barrier function at \( (x^+, \mu^+) \).

**Lemma 5.4.4.** Let \( (\bar{x}, \bar{s}) \) be a \( \theta \)-approximate \( \mu \)-centre, \( \mu^+ = \left(1 - \frac{1}{9\sqrt{n}}\right) \mu \), and all the violated constraints are moderately deep, i.e., \( \bar{d} < \alpha e \). Then for \( \alpha < 1 - \theta \) and \( 0 < \theta \leq 1/4 \), one has
\[
\varphi^+_p(x^+, \mu^+) \leq \varphi_p(\bar{x}, \mu^+) - \alpha - \log \left(1 - \frac{\alpha}{1 - \theta}\right) - e^\top \bar{d} - \sum_{j=1}^{p} \log \alpha \bar{t}_j + 0.40.
\]
**Proof:** The primal barrier function at \((x^+, \mu^+)\) reads

\[
\varphi_p(x^+, \mu^+) = \left(\frac{(c^+)^T x^+}{\mu^+}\right) - \sum_{j=1}^n \log \bar{x}_j (1 + \alpha \Delta x_j / \bar{x}_j) - \sum_{j=1}^p \log \alpha \bar{t}_j.
\]

Since \(\|\bar{X}^{-1} \Delta x\| \leq \frac{1}{1-\theta}\) and \(\alpha < 1 - \theta\), from Lemma 5.2.1

\[
\varphi_p(x^+, \mu^+) \leq \left(\frac{(c^+)^T x^+}{\mu^+}\right) - \sum_{j=1}^n \log \bar{x}_j - \frac{\alpha}{1-\theta} - \log \left(1 - \frac{\alpha}{1-\theta}\right) - \sum_{j=1}^p \log \alpha \bar{t}_j.
\] (5.8)

On the other hand,

\[
\frac{(c^+)^T x^+}{\mu^+} - \alpha e^T \bar{X}^{-1} \Delta x = \frac{c^T \bar{x}}{\mu^+} + \alpha \left[\frac{c^T \Delta x}{\mu^+} + \bar{c}^T \bar{t} - e^T \bar{X}^{-1} \Delta x\right].
\] (5.9)

In view of (5.6) one has

\[
\bar{c}^T \bar{t} = \bar{y}^T \bar{A} \bar{t} - \frac{\bar{t}^T \bar{D} \bar{t}^{-1}}{p},
\]

where \(\bar{D}\) is a diagonal matrix composed by vector \(\bar{d}\). Thus, the term in brackets in (5.9) reads as

\[
\frac{\bar{s}^T \Delta x}{\mu^+} - e^T \bar{X}^{-1} \Delta x = \frac{e^T \bar{d}}{\mu^+},
\]

or

\[
\left(\frac{s \bar{x}}{\mu^+} - e\right)^T (\bar{X}^{-1} \Delta x) = \frac{e^T \bar{d}}{\mu^+},
\] (5.10)

and using the Cauchy-Schwartz inequality we have

\[
\left(\frac{s \bar{x}}{\mu^+} - e\right)^T (\bar{X}^{-1} \Delta x) \leq \left\|\frac{s \bar{x}}{\mu^+} - e\right\| \left\|\bar{X}^{-1} \Delta x\right\|.
\]

Now, from (5.10), Corollary 5.2.4, Lemma 5.4.1 and the assumption that \(\mu \leq 1\), we have

\[
\frac{(c^+)^T x^+}{\mu^+} - \alpha e^T \bar{X}^{-1} \Delta x \leq \frac{c^T \bar{x}}{\mu^+} - e^T \bar{d} + 0.40.
\]

The proof follows from (5.8). \(\square\)

Notice that since \(\bar{d} > 0\), the term \(e^T \bar{d}\) can be eliminated from the bound in Lemma 5.4.4. We now bound the dual barrier function.
Lemma 5.4.5. Let the assumptions of Lemma 5.4.4 be satisfied. Then

\[ \varphi_d^+(s^+, \mu^+) \leq \varphi_d(\bar{s}, \mu^+) - \alpha - \log \left( 1 - \frac{\alpha}{1 - \theta} \right) - \sum_{j=1}^{p} \log \bar{r}_j + 0.40. \]

Proof: Observe that

\[ \varphi_d^+(s^+, \mu^+) = -b^\top y^+ - \sum_{j=1}^{n+p} \log s_j^+ \]
\[ = -b^\top \bar{y} - \frac{\alpha b^\top \Delta y}{\mu^+} - \sum_{j=1}^{n} \log \bar{s}_j (1 + \alpha \Delta s_j / \bar{s}_j) - \sum_{j=1}^{p} \log \bar{r}_j. \]

Now since \( \alpha < 1 - \theta \), in view of Lemma 5.2.1 and Lemma 5.4.1 we have

\[ \varphi_d^+(s^+, \mu^+) \leq \varphi_d(\bar{s}, \mu^+) + \frac{\alpha \bar{x}^\top \Delta s}{\mu^+} - \alpha e^\top \bar{S}^{-1} \Delta s - \frac{\alpha}{1 - \theta} - \log \left( 1 - \frac{\alpha}{1 - \theta} \right) - \sum_{j=1}^{p} \log \bar{r}_j. \]

On the other hand,

\[ \frac{\bar{x}^\top \Delta s}{\mu^+} - e^\top \bar{S}^{-1} \Delta s \leq \left| \left( \frac{\bar{x} \bar{s}}{\mu^+} - e \right)^\top \bar{S}^{-1} \Delta s \right| \]
\[ \leq \left| \frac{\bar{x} \bar{s}}{\mu^+} - e \right| \left\| \bar{S}^{-1} \Delta s \right\| \]
\[ \leq \frac{0.40}{1 - \theta}, \]

where the last inequality is due to Corollary 5.2.4 and Lemma 5.4.1. □

We now present the main result of this section.

Theorem 5.4.6. Let \((\bar{x}, \bar{s})\) be a \( \theta \)-approximate \( \mu \)-centre, \( \mu^+ = \left( 1 - \frac{1}{9\sqrt{n}} \right) \mu \), and all the violated constraints are moderately deep. Moreover, let \( \bar{d} < (\alpha/2)e \). Then for \( \alpha < 1 - \theta \) and \( 0 < \theta \leq 1/4 \) we have

\[ \varphi^+(x^+, s^+, \mu^+) - \varphi^+(x(\mu^+), s(\mu^+), \mu^+) \leq p \log p + \xi(p, \theta, \alpha), \]

where

\[ \xi(p, \theta, \alpha) = 1.0 - 2\alpha - 2 \log \left( 1 - \frac{\alpha}{1 - \theta} \right) - p \left( 1 + \log \frac{\alpha^2}{2} \right). \]
**Proof:** Adding inequalities in Lemma 5.4.4 and Lemma 5.4.5 gives

\[ \varphi^+(x^+, s^+, \mu^+) \leq \varphi(\bar{x}, \bar{s}, \mu^+) + 0.80 - 2\alpha - 2\log \left( 1 - \frac{\alpha}{1 - \theta} \right) - \sum_{j=1}^{p} \log \alpha t_j r_j, \]

From (5.7) we have

\[ \sum_{j=1}^{p} \log \alpha t_j r_j = \sum_{j=1}^{p} \log \frac{\alpha}{p} (\alpha - \bar{d}_j) \]
\[ = p\log \frac{\alpha}{p} + \sum \log (\alpha - \bar{d}_j) \]
\[ \geq p\log \frac{\alpha^2}{2p}, \]

where the inequality is valid because \( \bar{d}_j < \alpha/2 \), for \( j = 1, \ldots, p \). Thus,

\[ \varphi^+(x^+, s^+, \mu^+) \leq \varphi(\bar{x}, \bar{s}, \mu^+) + 0.80 - 2\alpha - 2\log \left( 1 - \frac{\alpha}{1 - \theta} \right) + p\log \frac{2p}{\alpha^2}, \]

and from Lemma 5.4.2 and Corollary 5.4.3 we have

\[ \varphi^+(x^+, s^+, \mu^+) \leq \varphi(x(\mu^+), s(\mu^+), \mu^+) + 1.0 - 2\alpha - 2\log \left( 1 - \frac{\alpha}{1 - \theta} \right) + p\log \frac{2p}{\alpha^2}. \] (5.11)

On the other hand,

\[ \varphi(x(\mu^+), s(\mu^+), \mu^+) = n - n \log \mu^+ \]
\[ = n + p - (n + p) \log \mu^+ - (p - p \log \mu^+) \]
\[ = \varphi^+(x(\mu^+), s(\mu^+), \mu^+) - p + p \log \mu^+ \]
\[ \leq \varphi^+(x(\mu^+), s(\mu^+), \mu^+) - p. \]

The proof follows from (5.11). \( \square \)

Note that at each iteration of the Newton method the barrier function is reduced by a constant amount. Therefore, Theorem 5.4.6 shows that after adding \( p \) moderately deep constraints and simultaneously updating \( \mu \), only \( O(p \log (p + 1)) \) Newton steps are required to obtain a point in the vicinity of the new \( \mu^+ \)-centre. We remark that the assumption \( \mu \leq 1 \) has been made only to simplify this bound. If \( \mu > 1 \), the complexity changes to \( O(p \log (\mu p + 1)) \).
5.5 Complexity analysis and convergence

The complexity analysis and convergence of Algorithm 2 is presented for the general case. Let $\bar{A}^\top y \leq \bar{c}$ be the $p$ violated constraints such that $\bar{c} < \bar{A}^\top \bar{y}$. In this section, we do not differentiate between moderate and very deep constraints. for simplicity we treat all constraints as deep. This approach covers the worst case behaviour of our algorithm.

**Lemma 5.5.1.** For $n \geq 2$, $\eta = \frac{1}{\sqrt{n}}$, $\theta = 0.25$ and $\alpha = 0.50$, we have

$$\phi_d^+(s(\mu^+, \mu^+)) \geq \phi_d(s(\mu), \mu) - p \log p - \sum_{i=1}^{p} \log v_i^{1/2},$$

where $v \in \mathbb{R}^p$ is composed of the diagonal elements of matrix $V$ as defined in Section 5.3.

**Proof:** First observe that

$$\phi_d^+(s(\mu^+, \mu^+)) = n + p - (n + p) \log \mu^+ - \phi_p^+(x(\mu^+), \mu^+)$$

$$\geq n + p - (n + p) \log \mu^+ - \phi_p^+(x^+, \mu^+),$$

and from Lemma 5.4.4 we have

$$\phi_d^+(s(\mu^+, \mu^+)) \geq n - n \log \mu - n \log(1 - \eta) - \phi_p(\bar{x}, \mu^+)$$

$$+ p - p \log \mu + \alpha + \log \left(1 - \frac{\alpha}{1 - \theta}\right) + e^\top \bar{d} + \sum_{j=1}^{p} \log \alpha \bar{t}_j - 0.40.$$  

Now, from Corollary 5.4.3 we have

$$\phi_d^+(s(\mu^+, \mu^+)) \geq n - n \log \mu - \phi_p(x(\mu), \mu)$$

$$+ p - p \log \mu + \alpha + \log \left(1 - \frac{\alpha}{1 - \theta}\right) + e^\top \bar{d} + \sum_{j=1}^{p} \log \alpha \bar{t}_j - 0.50.$$  

Thus

$$\phi_d^+(s(\mu^+, \mu^+)) \geq \phi_d(s(\mu), \mu) + \sum_{j=1}^{p} \log \bar{t}_j + p + \alpha + \log \left(1 - \frac{\alpha}{1 - \theta}\right) + e^\top \bar{d} + p \log \alpha - 0.50.$$  

On the other hand, [38] prove that

$$\sum_{j=1}^{p} \log \bar{t}_j \geq -p \log p - \sum_{j=1}^{p} \log v_j^{1/2}.$$
Therefore,
\[ \varphi_d^+ (s(\mu^+), \mu^+) \geq \varphi_d(s(\mu), \mu) - p \log p - \sum_{j=1}^{p} \log v_j^{1/2} + p + \alpha + \log \left( 1 - \frac{\alpha}{1 - \theta} \right) + e^\top \bar{d} + p \log \alpha - 0.50. \]

The proof follows by substituting \( \theta = 0.25 \) and \( \alpha = 0.50 \).

Lemma 5.5.1 establishes a bound on the optimal value of the updated dual barrier function after adding \( p \) deep constraints and updating \( \mu \). Notice that inequality (5.12) derived in this lemma is the same inequality that was derived for central cuts by [81], and [38]. Here, we simply ignore \( e^\top \bar{d} > 0 \) from the bound because we do not have any information on the depth of the cut. However, in practice, having deep constraints are beneficial in the sense that a feasible solution to the original problem is reached faster when constraints are added with no changes to their right hand side.

**Lemma 5.5.2.** At the \( k \)th iteration of the algorithm, let \( \mu_k \leq \mu_0 := 1 \), \( n_k := n_0 + n_p := 2m + \sum_{i=1}^{k} p_i \), and

\[ p = \max_{i=1,...,k} \{ p_i \} \]

where \( p_i \)'s are the number of deep constraints added at iteration \( i \). Then

\[ -\frac{\sqrt{m}}{\mu_k} + n_k \log \delta \leq -\varphi_d^k(s(\mu_k), \mu_k) \leq \frac{\sqrt{m}}{2} + 2m \log \frac{1}{2} + n_p \log(p + 1) + \sum_{i=1}^{n_p} \log v_i^{1/2}, \]

where \( \delta \) is the radius of the full dimensional ball contained in \( F_\Omega \).

**Proof:** The right hand side inequality follows from Lemma 5.5.1

\[ \varphi_d^k(s(\mu_k), \mu_k) \geq \varphi_d^0(s(\mu_0), \mu_0) - \sum_{i=1}^{n_p} p_i \log(p + 1) - \sum_{i=1}^{n_p} \log v_i^{1/2} \]

and the fact that

\[ \varphi_d^0(s(\mu_0), \mu_0) = \frac{-b^\top y}{\mu_0} - 2m \log \frac{1}{2} \]

\[ = -\frac{-b^\top e}{2} - 2m \log \frac{1}{2} \]

\[ \geq -\frac{\sqrt{m}}{2} - 2m \log \frac{1}{2} \]
where the inequality is due to \( \|b\| \leq 1 \), as presented in Assumption 3.

To prove the left hand side inequality, let \((y^c, s^c)\) be the centre of the \(\delta\)-ball. Then 
\[
s_i^c = c_i - a_i^\top y^c \geq \delta, \quad \text{for all } i = 1, \ldots, 2m + n_p.
\]
Also from Assumption 2 since \(F_{\Omega}\) is contained in the unit cube, we have \(\|y\|_\infty \leq 1\). Therefore, at the \(k\)th iteration
\[
\varphi_d^k(s^c, \mu_k) = -\frac{b^\top y^c}{\mu_k} - \sum_{i=1}^n \log s_i^c \leq \frac{\sqrt{m}}{\mu_k} - (n_p + 2m) \log \delta.
\]
The lemma now follows from \(\varphi_d^k(s(\mu_k), \mu_k) \leq \varphi_d^k(s^c, \mu_k)\).

The following lemma is due to \[81\].

**Lemma 5.5.3.** Let \(p \leq m\), then
\[
\sum_{i=1}^{n_p} \log v_i \leq 2m^2 \log \left(1 + \frac{n_p}{8m^2}\right).
\]

We now present the main theoretical result:

**Theorem 5.5.4.** For all \(i\), let \(1 \leq p_i \leq p \leq m\). Then after adding at most
\[
O\left(\frac{m^2 p^2}{\delta^2} e^{3\sqrt{m}/\varepsilon}\right)
\]
constraints, the IPCG algorithm stops with an \(\varepsilon\)-solution to the SILO problem.

**Proof:** From Lemma 5.5.2 we have
\[
-\frac{\sqrt{m}}{2} - \frac{\sqrt{m}}{\mu_k} + n_k \log \delta - n_p \log(p + 1) \leq 2m \log \frac{1}{2} + \sum_{i=1}^{n_p} \log v_i^{1/2}.
\]
Since \(p \geq 1\) and \(n_k \geq n_p\) one has
\[
-\frac{3\sqrt{m}}{2n_k \mu_k} + \log \left(\frac{\delta}{p + 1}\right) \leq \frac{1}{2n_k} (2m \log \frac{1}{4} + \sum_{i=1}^{n_p} \log v_i)
\]
\[
\leq \frac{1}{2} \log \frac{\frac{m}{2} + \sum_{i=1}^{n_p} v_i}{n_k}.
\]
\[
\leq \frac{1}{2} \log \frac{\frac{m}{2} + 2m^2 \log(1 + \frac{n_p}{8m^2})}{n_k},
\]
(5.13)
(5.14)
where (5.13) is due to the Geometric Mean Inequality, and (5.14) is due to Lemma 5.5.3. Notice that inequality (5.14) is valid at each iteration of the IPCG algorithm. Therefore, a feasible solution in the $\delta$-ball is obtained when this inequality is violated. That is,

$$\log \frac{m}{2} + 2m^2 \log(1 + \frac{m^2}{5m^2}) \leq - \frac{3\sqrt{m}}{n_k \mu_k} + \log \left( \frac{\delta}{p+1} \right)^2.$$ 

On the other hand, from Lemma 5.3.2, an $\varepsilon$-solution of the original problem is reached when $\mu_k \leq \frac{\varepsilon}{n_k + \sqrt{n_k}}$. Therefore, an $\varepsilon$-solution is achieved when

$$\log \frac{m}{2} + 2m^2 \log(1 + \frac{m^2}{5m^2}) \leq - \frac{3\sqrt{m}}{\varepsilon} + \log \left( \frac{\delta}{p+1} \right)^2,$$

or when

$$\frac{m}{2} + 2m^2 \log(1 + \frac{m^2}{5m^2}) \leq \frac{e^{-3\sqrt{m}/\varepsilon} \delta^2}{(p+1)^2}$$

holds. The proof now follows from this inequality. \hfill \Box

### 5.6 Numerical results

To compare the efficiency of the IPCG on an arbitrary case, the isocentres are randomly generated inside the GTV and PTV. In all of our test problems, we consider case 1, which has the following 10 structures: GTV, PTV, left eye, right eye, left lens, right lens, left optic, right optic, brainstem, and chiasm. Each structure contributes to the piecewise quadratic functions in SDO-SILO and the second-order cone constraint in SDO-SOCA. The number of voxels required to cover each structure depends on the volume of the structure and the resolution of the MRI picture. In our test problems we used 1 mm resolution in each dimension which resulted in 7278, 8357, 6083, 5740, 159, 162, 1336, 1589, 18260, and 774 voxels in the above 10 structures respectively. Therefore the dimension of the second-order cone in SDO-SOCA is 99,377. Note that since a structure is a three-dimensional object, improving the resolution by a factor of $p$ increases the dimension of the cone by a factor of $p^3$. For example, if we use 0.5 mm resolution and
improve the MRI picture from 30 × 30 × 30 to 60 × 60 × 60, the number of voxels and therefore the dimension of the cone is multiplied by 8.

Next we compare the three algorithms using problems with different number of isocentres. Table 5.1 illustrates our computational results. The first column shows the number of isocentres generated for the given GTV and PTV. The rest of this table compares the problem dimensions, the objective values, and the CPU times of the GP algorithm applied to SDO, the IPCG algorithm applied to SDO-SILO, and MOSEK applied to SDO-SOCO respectively. The problem dimension reported in this table is the maximum of the number of variables and the number of constraints. The number of variables in the original model SDO is 24 (8 sectors of size 3) times the number of isocentres. For example, in the case of 10 isocentres, this problem has 240 variables with only nonnegativity constraints. The IPCG algorithm on the SILO formulation SDO-SILO starts with the artificial dynamic bound constraints, the number of which is twice the number of variables. In case of 10 isocentres this box has 480 upper and lower bounds. While the number of variables remains the same, the number of constraints increases as the IPCG algorithm advances. In this example, the algorithm stops after generating 141 linear constraints. These new constraints are the tangent lines to the the “if” functions in SDO-SILO. Observe that at each iteration of the IPCG algorithm we have a feasible solution for the original problem SDO. The new constraints are generated to improve the objective value. The dimension of model SDO-SOCO is much larger. Besides the number of isocentres, the dimension of this problem also depends on the number of voxels in each structure, which is very large. The number of constraints in this problem is $2 \sum_{s \in S} v_s + 1$. The number of variables is $240 \times n_{\text{Iso}}$ (the number of isocentres) more than the number of constraints. This number is reported as the dimension for MOSEK in Table 5.1.

The computational results reported in Table 5.1 clearly show that the IPCG algorithm outperforms classical interior point methods (MOSEK) and the GP algorithm on both the objective value and the CPU time. For instance, in case of 35 isocentres, the GP
Table 5.1: Computational results of applying the GP algorithm to SDO, IPCG to SDO-SILO, and MOSEK (MSK) to SDO-SOCO to solve SDO. IPCG converges faster than GP and MSK. It also can handle more isocentres than MSK.

<table>
<thead>
<tr>
<th>nIso</th>
<th>GP</th>
<th>IPCG</th>
<th>MSK</th>
<th>GP</th>
<th>IPCG</th>
<th>MSK</th>
<th>GP</th>
<th>IPCG</th>
<th>MSK</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>240</td>
<td>480+141</td>
<td>101,777</td>
<td>57.06</td>
<td>56.84</td>
<td>56.88</td>
<td>20.36</td>
<td>1.19</td>
<td>2.30</td>
</tr>
<tr>
<td>15</td>
<td>360</td>
<td>720+160</td>
<td>102,977</td>
<td>58.08</td>
<td>57.92</td>
<td>57.96</td>
<td>29.28</td>
<td>2.02</td>
<td>4.13</td>
</tr>
<tr>
<td>20</td>
<td>480</td>
<td>960+189</td>
<td>104,177</td>
<td>53.77</td>
<td>53.70</td>
<td>53.74</td>
<td>58.41</td>
<td>4.22</td>
<td>9.18</td>
</tr>
<tr>
<td>25</td>
<td>600</td>
<td>1,200+181</td>
<td>105,377</td>
<td>43.57</td>
<td>43.45</td>
<td>43.52</td>
<td>109.13</td>
<td>5.22</td>
<td>10.01</td>
</tr>
<tr>
<td>30</td>
<td>720</td>
<td>1,440+188</td>
<td>106,577</td>
<td>41.89</td>
<td>41.44</td>
<td>41.54</td>
<td>166.71</td>
<td>7.56</td>
<td>12.36</td>
</tr>
<tr>
<td>35</td>
<td>840</td>
<td>1,680+187</td>
<td>107,777</td>
<td>40.48</td>
<td>39.35</td>
<td>39.52</td>
<td>277.01</td>
<td>11.14</td>
<td>22.57</td>
</tr>
<tr>
<td>40</td>
<td>960</td>
<td>1,920+195</td>
<td>108,977</td>
<td>43.63</td>
<td>42.63</td>
<td>42.78</td>
<td>487.59</td>
<td>15.39</td>
<td>26.70</td>
</tr>
<tr>
<td>45</td>
<td>1,080</td>
<td>2,160+199</td>
<td>110,177</td>
<td>40.71</td>
<td>39.78</td>
<td>39.96</td>
<td>520.25</td>
<td>21.15</td>
<td>29.47</td>
</tr>
<tr>
<td>50</td>
<td>1,200</td>
<td>2,400+195</td>
<td>111,377</td>
<td>38.75</td>
<td>38.31</td>
<td>38.50</td>
<td>820.61</td>
<td>24.10</td>
<td>37.81</td>
</tr>
<tr>
<td>55</td>
<td>1,320</td>
<td>2,640+200</td>
<td>112,577</td>
<td>35.60</td>
<td>35.22</td>
<td>35.53</td>
<td>1053.94</td>
<td>31.22</td>
<td>50.89</td>
</tr>
<tr>
<td>60</td>
<td>1,440</td>
<td>2,880+203</td>
<td>113,777</td>
<td>38.57</td>
<td>37.78</td>
<td>38.05</td>
<td>1286.70</td>
<td>32.93</td>
<td>51.23</td>
</tr>
<tr>
<td>65</td>
<td>1,560</td>
<td>3,120+206</td>
<td>114,977</td>
<td>37.78</td>
<td>36.22</td>
<td>–</td>
<td>1851.90</td>
<td>46.37</td>
<td>–</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>105</td>
<td>2,520</td>
<td>5,040+209</td>
<td>124,577</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>9999.99</td>
<td>129.22</td>
<td>–</td>
</tr>
</tbody>
</table>
algorithm applied to the original problem reaches the objective value of 40.48 in 277 minutes. MOSEK applied to SDO-SOCO does slightly better with the objective value of 39.52 and the CPU time of 22 minutes. However, for the same problem, the IPCG algorithm starts with 1680 box constraints, and after generating 187 constraints the algorithm stops with the objective value of 39.35, which only takes 11.14 minutes. This gives a CPU time improvement of 96% over the GP algorithm, and 50% over the classical interior point methods with MOSEK.

This behaviour is consistent throughout the table. However, the CPU time of the GP algorithm increases drastically as the problem size increases. This is largely due to the known zig-zag behaviour of the GP algorithms near the optimal solution. Moreover, there seems to be a memory jump in MOSEK. For problems with larger than 60 isocentres, MATLAB runs out of memory and quits the algorithm. Therefore this software is unable to solve very large-scale sector duration optimization problems.

We also tried to compare our computational results with SeDuMi and SDTP3 that are known to be efficient for SOCO. For a case of 10 isocentres, we used these software packages to solved the SOCO formulation SDO-SOCO. SDTP3 failed in uploading the data. SeDuMi successfully uploaded the data, but the coefficient matrix was very large. After 30 iterations and 13 minutes, SeDuMi ran into numerical problems and stopped with this error message: “No sensible solution found”. However, SeDuMi gives a solution which is not too far from the IPCG solution, but it cannot compute the optimal solution with high precision. We tested many other instances with different number of isocenter, but the result was the same. This might be due to the fact that the sector duration optimization problems are numerically hard in nature (personal communication with [64]). Nevertheless, even if the precision problem is fixed, the CPU time is not comparable with IPCG algorithm.

The convergence behaviour and iteration time of IPCG for a sample case is presented in Figure 5.2. Top figure represents the duality gap for the current solution in the current
relaxed linear optimization model. Unlike other constraint generation approaches, IPCG can solve to the optimal solution with high accuracy. Figure 5.2 bottom, shows that most of the iteration time consumed by centring procedure. It also shows that the algorithm become slower as it gets close to the optimal solution which is resulted from increases in dimension of the problem in later iterations. A clever constraint removal strategy will significantly decrease the solution time.

Figure 5.3 presents the GP convergence behaviour on solving SDO for Case 1. Objective value, the absolute value of the objective gradient, and the relative improvement of the objective in each iteration are presented in this figure. Although objective values do not change significantly at the later iterations, the changes on the gradients and relative improvements are noticeable. This confirms that the GP algorithm does not guarantee to provide a good approximation of the optimal solution in reasonable time.

5.7 Conclusions

IPCG is an exponential algorithm that can be used to solve semi-infinite linear optimization problems. In special cases, it can also be used to solve convex optimization problems. IPCG is not an exact algorithm meaning that the final solution may not be feasible although it might be very close to the optimal solution. Not surprisingly, this is not the case when we apply IPCG to the SDO problem in PFX treatment planning. With the SILO reformation of the SDO, IPCG always stays in positive orthant which is the only restriction in the SDO model. Another concern is that due to the infinite nature of the semi-infinite problems, it is not surprising that IPCG is not a polynomial algorithm. But similar to the linear optimization case, we can use heuristics to speed up the convergence. One such approach is to use Mehrotra’s predictor corrector step \[56\] in centring procedure of the algorithm. Similar to other interior-point based algorithms in the LP case, in case the optimal solution set is not singleton, IPCG will converge to
Figure 5.2: Convergence behaviour of IPCG: Top: The duality gap of the relaxed problem at each iteration. This gap represents a bound on optimal solution. Bottom: The proportion of iterate time consumed by centring, oracle, and recovering procedures. As new constraints are added to the LP approximation, the centring process time per iteration increases.
Figure 5.3: Convergence behaviour of the gradient projection algorithm in solving SDO is presented in this figure. Although objective values are not changing significantly as the algorithm goes on, the relative improvement and the magnitude of the objective gradient changes dramatically.
the analytic centre of the optimal face [15]. As this might not be considered as a disadvantage in many cases, in solving SDO, it causes a significantly large beam-on time for the plans generated by IPCG (or any interior-point based algorithm). Specially, when a large number of isocentres is considered in SDO compare to other type of algorithm that converges to a vertex rather than analytic centre. One can employ a purification scheme to avoid such a problem [41].

The algorithm that we described here has the potential to be combined with branch-and-cut algorithms and to be implemented to solve mixed integer conic programming problems. An efficient technique for problems of this kind is the use of outer approximation of the second-order cone constraints. See, for instance, [14] and [2]. The main reason to use polyhedral approximation is the opportunity to have a warm start in the branch-and-bound algorithm after adding an integer cut. [12] develop a polyhedral approximation for second-order cone optimization that is used by [74] in their approach to mixed integer conic programming.

The advantage of this approximation is that it is computed once and used at every relaxation node. However, this approximation, although tight, yields an LP with a large number of constraints and variables. For example the polyhedral approximation of a single second-order cone of dimension 4, creates an LP with over 10,000 variables and 22,000 constraints (with high precision). This could be costly for MOSEK, especially when the number of cones and their dimensions are large. We see a potential advantage of our algorithm in solving mixed conic integer programming problems. Recently, there has been a growing interest in problems with conic constraints and integer variables. See, for instance [19] and [9]. Many such problems arise in finance and engineering applications. Solving the conic relaxation of these problems by the classical interior point methods might be disadvantageous because there is no clear strategy for a warm start after an integer cut is introduced in the branch-and-cut algorithm. The use of outer approximation of conic constraints, on the other hand, has been proven to be a successful
approach to this class of problems. For instance [74] use the polyhedral approximation of [12] to take advantage of the warm start strategy in their branch-and-bound algorithm, [46] use a polyhedral cut and price approach for the maxcut problem, and [14] and [2] use the idea of polyhedral approximation in their integer programming solvers.

Constraint and column generation methods have been combined with branch-and-cut and branch-and-price algorithms (as for instance in [25, 26], [43]) and proved to be efficient methods for integer programming. These methods have also been efficiently combined with conic optimization (as for instance in [71], [59, 60], [22, 61], and [17]). Therefore, solving convex conic optimization problems by outer approximation constraint generation methods is not only beneficial in a class of large-scale problems as we explored in this chapter, it could also be combined with branch-and-cut, and branch-and-price methods to develop efficient algorithms for mixed integer conic programming. We intend to explore these avenues in our future research.
Chapter 6

Discussion and conclusion

We have shown that mathematical frameworks successful in IMRT optimization can be applied to Leksell Gamma Knife® Perfexion™ inverse planning. These models are flexible enough that despite the large number of voxels in PFX treatments, solutions can be obtained in a clinically viable amount of time using a standard projected gradient algorithm. Our treatment plans demonstrate high conformity and satisfactory clinical objectives. The results indicate that our approach to the inverse problem yield conformal treatment plans that satisfy the clinical objectives. These automatically generated plans are capable of being delivered on the treatment unit. The SDO model tends to find better conformal plans rather than a plan with lower BOT. However, the beam-on time can be reduced to a clinically acceptable range with the cost of reducing the conformity (still in the acceptable range). Nevertheless, the framework for posing the problem is created and can be used to guide treatment planners to explore the tradeoffs between delivery efficiency and dose conformity. A classical way to explore the tradeoff between treatment plan quality measures (conformity indices, beam-on time, brainstem max dose, ...) is to formulate a multi-objective model. The Pareto frontier of such a model indicates the possible tradeoff between different quality metrics. The IPCG algorithm developed in Chapter 5 could solve the sector duration optimization...
model of the Gamma Knife\textsuperscript{®} Perfexion\textsuperscript{TM} treatment planning problem and showed that our algorithm outperforms the primal-dual interior point methods as well as the gradient projected algorithm. The hybrid grassfire and sphere-packing isocentre selection algorithm is flexible and computationally inexpensive but does not necessarily result in the optimal positioning of isocentres. The SDIO model then is developed and used to solve combined sector durations and isocentre optimization. The results show that restricting the number of isocentres improves the computation time. The results also show that reducing the maximum duration time, $t_{\text{MAX}}$, reduces the beam-on time significantly with the cost of slightly reducing the conformity of the plan. With the SDIO model we can also incorporate machine minimum duration limit. In terms of computational performance, limiting the number of isocentres decreases the computation time of solving SDIO model significantly, though tight bounds result in more computational effort than loose bounds. On the other hand, if the bounds are poorly calculated, SDIO can easily become infeasible. Therefore, a clever and efficient way of bounding the number of isocentres is vital to the performance of SDIO model. Our isocentre bounding method performed well in the cases tested, though a larger population of test cases is required to verify that our bounding approach is always feasible. In terms of the quality of the generated treatments, not surprisingly, the tighter limit on the number of isocentres, the less improvement over the forward plans. After finding the sector durations with our inverse planning models, the sectors are combined together into deliverable shots supported by the treatment unit. In this process any shot of duration 10 seconds or less is removed to reflect the treatment unit limitations and minimize the effect of shutter dose. The obtained beam-on times for our plans are acceptable, but the rather large number of shots can be inconvenient if the shots are manually entered into PFX. Also, the number of shots for each plan depends on the optimization solution from SDO and therefore is not predictable.

To conclude, through computerization, our inverse planning approach fully utilizes Perfexion\textsuperscript{TM} features, perceivably improves plan quality, and reduces beam-on time. In
fact, computerization helped to find an optimal treatment plan and to achieve higher conformity than clinical plans while sparing healthy tissues. Our automatic approach to find an optimal treatment plan sets a high quality standard to the treatment planning process, resulting in a better and standardized brain cancer management solution. The proposed models and solution techniques may improve, prolong, and save lives by providing clinical solutions for treating cancers and brain disorders ensuring all treatment guidelines are satisfied. As a future extension to this study, we may consider incorporating non-linear objectives or constraints into our SDIO model which results in solving a non-linear mixed integer optimization (NMIO). One option is to consider a similar approach taken by Ferris et al. [30] but with an actual dose calculation function instead of approximation. We would also consider using binary variables instead of a continuous approximation that results non-convexity of each subproblems. Then, we would need to extend the IPCG algorithm to solve NMIO, which most likely will perform faster than other existing algorithms.
Appendix A

A.1 Semi-Infinite Linear Optimization

A.1.1 General Form

Semi-infinite linear optimization (or programming) (SILO) deals with an optimization problem with linear objective and linear constraints in which either the number of constraints or the dimension of the variables space, but not both, is allowed to be infinite. The primary purpose of the thesis is to develop and study an algorithm to solve SILO, i.e., programs that can be formulated as

$$\max \left\{ b^\top y : a_t^\top y \leq c_t, \ t \in \mathcal{T} \right\}, \quad (A.1)$$

where $b \in \mathbb{R}^m$, $\mathcal{T}$ is an arbitrary (possibly infinite) index set,

$$a_t = a(t) = (a_1(t), \ldots, a_m(t)) \quad (A.2)$$

maps $\mathcal{T}$ into $\mathbb{R}^m$, and $c_t = c(t)$ is a scalar function on $\mathcal{T}$. Problem (A.1) is said to be the Dual SILO.

Let us observe that the feasible set of (A.1);

$$\mathcal{F} = \left\{ y : a_t^\top y \leq c_t, \ t \in \mathcal{T} \right\} \quad (A.3)$$

\footnote{Different sources define semi-infinite linear optimization in different manner but equivalent to each other. The above definition is borrowed from Goberna and López, [34]. The reader may find more details and results in [7,8,31,35].}
Appendix A. SILO optimality

is a closed convex set in \( \mathbb{R}^m \), since it is the intersection of a family of closed half spaces. The *semi-infinite linear system* \( \{ a^T_i y \leq c_t, \ t \in \mathcal{T} \} \) provides an external representation of the feasible set \( \mathcal{F} \). Therefore, to some extent, (A.1) is a convex linear optimization problem. If (A.1) is consistent, i.e., if it has at least one feasible solution, then the *optimal value* \( \nu \) of problem (A.1) can be either a real number or \( +\infty \).

There are two major classes of SILOs. Those where the infinite index set \( \mathcal{T} \) is countable and those where the index set is the continuous subset of the Euclidean space (such as a line segment or rectangle). We shall call them the *countable* and *continuous* semi-infinite linear programs, respectively. Therefore, we can rewrite the countable case of problem (A.1) as follows:

\[
\max \left\{ b^T y : a^T_i y \leq c_i, \ i = 1, 2, 3, \ldots \right\}.
\] (A.4)

The primal semi-infinite linear optimization problem associated with problem (A.4) is defined as follows:

\[
\min \left\{ \sum_{i=1}^{\infty} c_i x_i : \sum_{i=1}^{\infty} x_i a_i = b, \ x \geq 0 \right\},
\] (A.5)

where \( x_i \) is zero for all \( i = 1, 2, 3, \ldots \), except for a finite number of indices for which \( x_i \geq 0 \) (see \([7]\)).

**Example 1** (Example of Karney). Consider the following primal semi-infinite linear optimization problem:

\[
\begin{align*}
\min \quad & x_1 \\
\text{s.t.} \quad & -x_1 - x_3 - x_4 - \cdots = -1, \\
& x_2 + \frac{1}{3} x_3 + \frac{1}{4} x_4 + \cdots = 0, \\
& x \quad \geq \quad 0
\end{align*}
\] (A.6)

Problem (A.6) meets its optimal value 1 at point \( x_1^* = 1, \ x_i^* = 0, \ i = 2, 3, \ldots \).
The dual problem associated with problem (A.6) is given as follows:

\[
\begin{align*}
\max & \quad -y_1 \\
\text{s.t.} & \quad -y_1 \leq 1, \\
& \quad y_2 \leq 0, \\
& \quad -y_1 + \frac{y_2}{i} \leq 0, \quad i = 3, 4, \ldots.
\end{align*}
\] (A.7)

The optimal solution is \(y = (0, 0)\) with the optimal value 0.

In Example of Karney, although both primal and dual are feasible, however, the duality gap is not zero.

### A.1.2 Convex Optimization and SILO

Consider the following convex linear optimization problem:

\[
\max_{y \in \mathcal{D}} \left\{ b^\top y : g(y) \leq 0 \right\}
\] (A.8)

where \(\mathcal{D}\) is a closed and bounded subset of \(\mathbb{R}^m\), and \(g : \mathbb{R}^m \to \mathbb{R}\) is a convex function. Problem (A.8) is equivalent to the following SILO

\[
\max_{y \in \mathbb{R}^m} \left\{ b^\top y : g(\bar{y}) + v(\bar{y})^\top (y - \bar{y}) \leq 0, \quad \bar{y} \in \mathcal{D} \right\},
\] (A.9)

where the vector \(v(\bar{y}) \in \mathbb{R}^m\) is a sub-gradient of \(g\) at \(\bar{y}\), i.e., it satisfies the following inequality:

\[
g(y) - g(\bar{y}) \geq v(\bar{y})^\top (y - \bar{y}).
\] (A.10)

If \(g\) is differentiable at \(\bar{y}\), then the only sub-gradient of \(g\) at \(\bar{y}\) is the gradient \(\nabla g(\bar{y})\) (See Figure A.1 for an illustration of sub-gradient cuts).

### A.2 Optimality theory

Consider the following semi-infinite linear optimization problem:

\[
\inf \left\{ \sum_{i=1}^{\infty} c_i x_i : \sum_{i=1}^{\infty} x_i a_i = b, \quad x_i \geq 0 \right\}, \quad (P_\infty)
\]
Figure A.1: The dashed lines indicate two different sub-gradient cuts of the function $g$ at $y_1$, while the only sub-gradient cut at $y_2$ is the tangent line.

where $c_i \in \mathbb{R}$, $a_i \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$, and $x_i > 0$ for only a finite number of indices $i$, i.e., all the sums have only a finite number of non zeros components.

The dual form of the problem $(P_\infty)$ is

$$\sup \left\{ b^\top y : a_i^\top y \leq c_i, \; i = 1, 2, 3, \ldots \right\} \tag{D_\infty}$$

Recall that a problem is called consistent if it has a feasible solution. In general, and without any regularization techniques, there is no reason that all such problems as $(P_\infty)$ can be discretized in order to be solved iteratively, or the optimal value of $(P_\infty)$ and $(D_\infty)$ coincide.

**Example 2.** Consider the SILO problem

$$\inf \left\{ y_1 : ty_1 + (1-t)y_2 \geq t - t^2, \; t \in (0, 1) \right\} \tag{A.11}$$

For any fixed $t \in (0, 1)$ the optimal value of the SILO problem given by (A.11) is $-\infty$, but the optimal value of problem (A.11) is 0, (see [34], Example 1.1). Thus, some conditions
Appendix A. SILO optimality

are needed to ensure that problem \((P_\infty)\) can be solved by discretization, or by constraint generation technics.

**Theorem A.2.1** (Weak Duality). If \((P_\infty)\) and \((D_\infty)\) are both consistent, then the optimum value of \((P_\infty)\) is greater than the optimum value of \((D_\infty)\), and both values are finite.

**Theorem A.2.2** (Complementary Slackness). If \(x = (x_i)_{i=1}^\infty\) is feasible for \((P_\infty)\), \(y \in \mathbb{R}^m\) is feasible for \((D_\infty)\) and
\[
\sum_{i=1}^\infty x_i(c_i - a_i^\top y) = 0, \tag{A.12}
\]
then \(x\) is optimal for \((P_\infty)\) and \(y\) is optimal for \((D_\infty)\).

Having formulated the primal and dual problems, we now state conditions which will ensure that their optimal values coincide. First we need to define two sets:

\[
M_A = \left\{ \sum_{i=1}^\infty x_i a_i : x \geq 0 \right\},
\]

\[
M_B = \left\{ \left( \sum_{i=1}^\infty x_i a_i, \sum_{i=1}^\infty x_i c_i \right) : x_i \geq 0, i = 1, 2, 3, \ldots \right\}.
\]

Note that \(M_A\) and \(M_B\) are cones in \(\mathbb{R}^m\) and \(\mathbb{R}^{m+1}\), respectively.

**Theorem A.2.3.** If the optimal value of \((D_\infty)\) is finite and \(M_B\) is closed, then the optimal values of \((D_\infty)\) and \((P_\infty)\) coincide.

The proofs of the theorems can be found in [7].

**Example 3.** Considering the Example of Karney (see Example 1), \(M_A\) is the cone generated by
\[
\{(-1, 0), (0, 1), (-1, 1/3), (-1, 1/4), \ldots\},
\]
and \(M_B\) is the cone generated by
\[
\{(-1, 0, 1), (0, 1, 0), (-1, 1/3, 0), (-1, 1/4, 0), \ldots\}.
\]
Note that the vector $b = (-1, 0)^T$ in this example is on the boundary, not in the interior of $M_A$, and $M_B$ is not a closed cone. Hence, we cannot expect to have zero duality gap in this example.
Bibliography


[64] I. Polik. Personal communication. *COR@L Computational Research At Lehigh.,* 2009.


