

**NEW METHODS IN OPTIMIZATION ALGORITHMS AND
TECHNOLOGY APPLIED TO VMAT AND IMRT INVERSE TREATMENT
PLANNING**

by

Joshua Allen Mathews

May 2020

A dissertation submitted to the
Faculty of the Graduate School of
the University at Buffalo, The State University of New York
in partial fulfilment of the requirements for the
degree of

Doctor of Philosophy

Department of Radiology, Medical Physics Program

ProQuest Number:27997297

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent on the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



ProQuest 27997297

Published by ProQuest LLC (2020). Copyright of the Dissertation is held by the Author.

All Rights Reserved.

This work is protected against unauthorized copying under Title 17, United States Code
Microform Edition © ProQuest LLC.

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

Copyright by
Joshua Allen Mathews
2020

Dedication

To Scratch (AKA Captain Scratchopants AKA Scroucho AKA El Scratchador) who has been with me during my entire medical physics education.



At some point during my dissertation writing I decided the Scratcho needed eyebrows. I had a dry-erase marker nearby, so he got some eyebrows.

Acknowledgments

I have been very fortunate to have learned from some fantastic physicists over the course of my education: I would like to thank Dr. Larry Sweeney and Dr. Miron Zaini from the Northwest Medical Physics Center for introducing me to the field and teaching me the roles of a clinical medical physicist. Working for NMPC as an intern is what solidified my desire to pursue this career. Thank you to my medical physicist assistant (MPA) colleagues at Roswell Park: Stephen Bhagroo, Joshua Kilian-Meneghin, and Yanan Cao. Working with you all was an absolute privilege, and I can't thank you enough for covering the clinic while I was out interviewing for a residency position. I look forward to bumping into you at conferences and seminars.

Additionally, I was trained to be an MPA by Samuel French, Dennis Mohatt, Tianjun (TJ) Ma, and Naveed Islam. A huge thanks to them for their knowledge and patience as they trained me on different aspects of the clinic.

Thank you to Dr. Matthew Podgorsak, Dr. Lalith Kumaraswamy, and Dr. Anh Le for their mentorship during my time as an MPA. Anytime I had a question or concern regarding protocol, they were always able to point me in the right direction. As chief MPA, they provided me with leadership guidance, which helped me effectively lead the MPA team. I will certainly follow their example as I pursue a career in medical physics.

I would like to thank my research team members: Ankit Pant, Kun Lin, Dr. Shawn Matott, and Dr. Matthew Jones. Your contributions to the projects detailed in this dissertation are greatly appreciated. I can't wait to see what more will come from the projects for which the groundwork has been laid.

I thank the other students in the program: Amy Lau, Hope Genko, Tara Tyson, Jonathan Troville, Jordan Krebs, to Eftekhar Bolookat, Zeynab Rezaee, Allison Shields, Chao Guo, and others I may have missed for your support, and mostly for putting up with my inappropriate sense of humor and keeping me sane during my time here. From now on, any time I make a late night cafeteria run, I'll be reminded of you all.

Thank you to the rest of the medical physics faculty who had a role in my education at UB and Roswell Park: Dr. Iris Wang, Dr. Ciprian Ionita, Dr. Rutao Yao, Professor Steven de Boer, and Dr. Richard Harvey. Whether it be formal didactic work or informal question and answer sessions, I learned so much from them all that will be put to good use.

I would like to thank Dr. Stephen Rudin for allowing me to attend this program. Right away I was impressed by the level of organization this program has; it is clearly orchestrated by someone with a wealth of knowledge and experience in training the next generation of medical physicists both clinical and imaging.

Thank you to those with whom my research collaborated with from the University of Toronto and Fujitsu. The new technologies and their potential applications present some very exciting prospects, and I look forward to seeing the innovations that will be produced.

I thank my Ph.D. committee members: Dr. Harish Malhotra, Dr. Daniel Bednarek, and Dr. Kenneth Hoffmann. Their questions and comments on my

projects have pushed me to produce quality work as well as how to present my ideas and results clearly.

Last, but certainly not least, I would like to thank my Ph.D. advisor: Dr. Daryl Nazareth. I met him during my visit to UB and Roswell Park in March, 2016; a trip for me to gauge whether the program and I would be a good fit. Dr. Nazareth impressed me with his description of the program and its resources available to students, and I entered the program the following fall. Over the course of my time here he has put up with my numerous questions with patience and understanding. As a mentor, I couldn't have asked for anyone better; I truly am grateful for his guidance during my pursuit of a Ph.D.

Table of Contents

Dedication	iii
Acknowledgments	iv
List of Tables	xii
List of Figures	xiv
Nomenclature	xxviii
Abstract	xxxii
Chapter 1 Introduction	1
1.1 Cancer and Radiation Therapy	1
1.2 Monte Carlo Radiation Dose Calculation	2
1.3 Organization of this Dissertation	3
1.4 Publications and Conference Presentations	4
Chapter 2 Inverse Planning and Optimization of Dynamic Radiotherapy Plans	6
2.1 Using Multiple Beams and Arcs to Mitigate Dose to Healthy Tissues	6
2.2 Beam Collimation	8

2.2.1	The Jaws	8
2.2.2	The Multileaf Collimator	9
2.3	DICOM Files	10
2.4	Dose-Volume Objectives and Objective Function Optimization in Inverse Planning	13
2.5	IMRT Treatment Plan Optimization	16
2.6	VMAT Treatment Plan Optimization	17
2.6.1	Enhanced Optimization	18
Chapter 3	A Framework for Parallelized Monte Carlo Simulations	21
3.1	Motivation	21
3.1.1	Monte Carlo Simulations	21
3.1.2	Independent Calculation of Dose Volume Histograms for Optimization	23
3.2	Methods	23
3.2.1	Anonymized DICOM File Export	23
3.2.2	EGSnrc User Codes and Input Files	24
3.2.3	Parallelization of Monte Carlo Simulations	25
3.2.4	Generation of EGSnrc Input Files	26
3.2.4.1	DOSXYZnrc Input File	27
3.2.4.2	MLC Leaf Position File	28
3.2.4.3	Dynamic Jaws Position File	32
3.2.4.4	Virtual Phantom	35
3.2.5	Running EGSnrc Jobs on the Cluster	36
3.2.6	DVH Calculation Using Binary Masks	40
3.3	Results	44

3.3.1	Anonymized DICOM Files	44
3.3.2	EGSnrc Input Files and SLURM Job Arrays	44
3.3.3	Binary Masks	44
3.3.3.1	Comparison of calculated volumes	45
3.3.3.2	Visual Comparison of Masks and Contours	46
3.3.3.3	Comparison of DVH's	48
3.4	Discussion	52
3.4.1	EGSnrc Input Files and SLURM Job Arrays	52
3.4.2	Evaluating the Accuracy of Binary Mask Generation	53

Chapter 4 Enhanced Optimization of VMAT Plans Using Monte Carlo

	Generated Beamlets	55
4.1	Motivation	55
4.2	Materials and Methods	56
4.2.1	Exporting VMAT Plan Files	56
4.2.2	Beamlet Calculation Using Monte Carlo	57
4.2.2.1	Processing DOSXYZnrc Output - the 3ddose File	58
4.2.2.2	Exclusion of Unnecessary Beamlet Calculations .	61
4.2.2.3	Generating Input Files and Automation Scripts for Beamlet Calculations	64
4.2.2.4	Choosing the Number of Histories for the Full Arcs and Beamlets	65
4.2.2.5	Running Modified Control Point MC Simulations on the Cluster	71
4.2.2.6	Beamlet Post-processing	72
4.2.3	Enhanced Optimization Objective Function	74

4.2.4	Evaluations	76
4.3	Results	77
4.3.1	Enhanced Optimization with Original Objectives	78
4.3.2	Enhanced Optimization with Modified Objectives	82
4.4	Discussion	87
4.4.1	Clinical Implications	87
4.4.2	Differences in Eclipse and MC Calculations	90
Chapter 5 Application of the Hamiltonian Engine for Radiotherapy Optimization on Prostate IMRT Plans		93
5.1	Motivation	94
5.2	Materials and Methods	96
5.2.1	Using CERR to Obtain Pencil Beam Information for Optimization	96
5.2.2	Optimization of IMRT Plans with Eclipse	100
5.2.3	Optimization of IMRT Plans with the HERO	100
5.2.4	Evaluations	101
5.3	Results	102
5.4	Discussion	109
5.4.1	Normal and Strict Objectives	109
5.4.2	Limitations in the Study	109
5.4.3	Comparison to Quantum Annealer Approach	110
Chapter 6 Conclusions		112
6.1	A Framework for Parallelized Monte Carlo Simulations	112
6.2	Enhanced Optimization of VMAT Plans Using Monte Carlo Generated Beamlets	114

6.3 A Hamiltonian Engine for Radiotherapy Optimization	115
Chapter A Code and Utilities	117
References	118
Vita	124
Colophon	125

List of Tables

2.1	Summary of contents of DICOM files. Although other types of DICOM files are used in radiation oncology, the types included in this table are the only ones needed for the work detailed in this dissertation.	12
4.1	A summary of the computing resources used and time required to run a single VMAT arc using DOSXYZnrc on the UB CCR cluster. Additionally, the mean uncertainty reported by each simulation is included along with the standard deviation.	66
4.2	Designations VMAT plans selected for this study and their associated number of arcs, objectives, and variables (i.e., the number of active leafs for which beamlets were calculated).	77
4.3	Objective modification types used for the pediatric brain case. The objective is the maximum dose to the left cochlea PRV.	77
4.4	Summary of the enhanced optimization results with original objectives. Penalties are calculated with MC data applied to Eq. 4.5, and dose values are extracted directly from Eclipse.	79

4.5	Objective function scores for pediatric brain plans. The EO plans correspond to different objectives on one OAR. "Unmodified" and "Type 1" resulted in the same plan after the greedy search was completed. The "Type 2" DVH was only different in high-dose region (see text). All EO plans had lower scores than the Eclipse plans.	84
4.6	Summary of the pediatric brain plan's enhanced optimization results with original, "Type 1" and "Type 2" objectives applied to the left cochlea PRV. Penalties are calculated with MC data applied to Eq. 4.5, and dose values are extracted directly from Eclipse.	84
5.1	A summary of the objectives used for optimizing the prostate IMRT plans using SA, Eclipse's DVO, and the HERO.	102
5.2	Dose-volume data for each objective averaged over all seven IMRT patients for each optimization approach. For each OAR objective, the average HERO result is lower than the corresponding Eclipse optimizer result, which indicates that, to varying amounts, the HERO produced more favorable OAR doses at the volume specified by the objective.	103

List of Figures

2.1	A prostate case with relative dose color washes for a five-beam plan (a) and an arc plan (b). For the five-beam plan, the gantry does not move during beam delivery. Depending on the type of treatment, the beam's shape and intensity may be modulated, but the beam's direction will not change. For an arc plan, the gantry rotates around a fixed point, known as the isocenter, and continuously delivers beam as the gantry moves.	8
2.2	A graphical representation radiation beam collimation. The jaws move along a half arc in and out of the beam's path to create rectangular-shaped beams. The leafs move in and out of the beam's path to create beams of irregular shapes. Image from (8) .	9
2.3	A graphical representation of MLC leafs. The leafs move in and out of the beam's path to create beams of irregular shapes during the treatment. Image from (9).	10

2.4 A textual representation of part of a DICOM file. Specifically, this shows the information about the linear accelerator, source to axis distance, and beam limiting devices (i.e., x and y jaws - the MLC information is not shown in this figure). On the left-most side are alphanumeric keys that can be called by the DICOM reader, and the indentations indicate nested data. For example, should the user want to access the number of jaw pairs for the Y jaws, they would need to use the alphanumeric key: (300a,00b0),(300a,00b6),(300a,00bc), which would fetch "1", indicating that there is 1 pair of Y jaws. Every key piece of information regarding a treatment plan can be obtained with this method from the DICOM files. 11

2.5 A visual representation of the area under the dose-volume histogram curve used for the penalty of dose-volume objectives. The penalty scales with the volume, ΔV , and dose, ΔD , that violates the objective. 15

2.6 An example of a DVH belonging to a head and neck VMAT case. Objectives are denoted by triangles where the right-angle point corresponds to the objective. 16

2.7 A graphical representation of Otto's VMAT optimization algorithm. The algorithm optimizes a few initial control points (the white triangles) (a), adds additional interpolated control points (the grey triangles) (b and c), does another round of optimization, and then adds even more control points (the black triangle) (d) until the desired sampling frequency (i.e., maximum number of control points) is reached. Image from (7) 17

2.8	A simplified 1-D representation of the objective space for a VMAT optimization problem. The orange curve is the score calculated by Eclipse during optimization, while the blue curve is the true score. After Eclipse locates an approximate local minimum, it is used as the starting point for the EO. The EO searches the neighborhood using a more accurate dose (and therefore score) calculator. (b) is an enlargement of the region of (a) indicated by the grey box.	19
3.1	A flowchart showing the DICOM files needed to generate each type of EGSnrc input file or binary mask.	27
3.2	A schematic of the MLC leaf geometry as modelled by the BEAM-nrc user code. The x ray source is indicated by the red dot, and the beam geometry is indicated by the red lines. Using the rule of similar triangles, physical horizontal distance between the central axis and the tip of the curve of the MLC leaf, x , can be represented in terms of the source to axis distance (SAD), the vertical distance between the target and the center of the leaf, z_{MLC} , and the coordinate of the leaf projected to a plane at isocenter x'	30

3.3	A schematic of the x jaws as modelled by the BEAMnrc user code. The x ray source is indicated by the red dot, and the beam geometry is indicated by the red lines. Using the rule of similar triangles, the values of XTN , XBN , XTP , and XBP , which are needed for the dynamic jaws input file, can be solved for in terms of $z_{x,T}$ and $z_{x,B}$, the vertical distances between the x ray source and top and bottom of the jaws, respectively; x_1 and x_2 , the coordinate of each jaw projected to a plane at isocenter; and SAD , the source to axis distance (100 cm).	33
3.4	An example of a job array read by the SLURM scheduler. The header contains information that sets the parameters for the job (e.g., job title, maximum time), and the body contains the commands that execute the job.	38
3.5	3-D structures are defined as a collection of 2-D contoured polygons. Pictured are several slices from a prostate VMAT case with the bladder (blue), rectum (green), and femoral heads (red and cyan) contours.	41

3.9	A CT slice of a brain VMAT patient containing a contour (blue line) of a PTV with the associated binary mask overlaid (red). The main difference between the contour and mask is apparent at the periphery. For this particular slice, the mask appears to be slightly smaller than the contour.	48
3.10	(a) A CT slice of a head and neck VMAT plan containing a two non-overlapping contours (yellow line) of a PTV with the associated binary mask overlaid (red). (b) A CT slice of a separate head and neck VMAT plan containing a hollow structure defined by two overlapping contours (pink line) with the binary mask overlaid (red).	49
3.11	Four adjacent CT slices of a brain VMAT plan containing the contours (yellow lines) of an optic chiasm, a very small structure ($0.4cm^3$) with the binary mask overlaid (red)	50
3.12	DVH with structures of a brain VMAT plan for the same Eclipse dose matrix. Particularly, the smallest structures defined for the plan (and the GTV) are included. The dashed curves are data directly from Eclipse while the solid curves were calculated using binary masks and the dose array extracted from the RD file.. . . .	51
3.13	DVH with structures of a prostate VMAT plan for the same Eclipse dose matrix. These structures are relatively large when compared to structures in brain and head and neck plans. The dashed curves are data directly from Eclipse while the solid curves were calculated using binary masks and the dose array extracted from the RD file.	51

4.1	A flow chart of the clinical treatment planning process. The conventional workflow is indicated in blue, while the proposed EO addition is indicated in red.	57
4.2	The beam's eye view of contoured patient anatomy for a prostate VMAT plan at a point along the arc (i.e., a control point). The red square represents the cross section of a beamlet generated by moving a single leaf indicated in the figure out by 0.5 cm.	58
4.3	An axial cross section of the same anatomy shown in Figure 4.2 with the beamlet dose displayed as a relative color wash.	59
4.4	A CT slice of a VMAT prostate case with a color wash (blue-low dose to pink-high dose) of the dose contribution of a partial arc.	59
4.5	Perturbations to the TPS plan are performed by adding/subtracting a beamlet dose matrix to/from the current dose matrix, depending on whether the MLC leaf position is moved out of or into the field.	60
4.6	Beam's eye view, including MLC leafs, of a PTV from a VMAT brain case. The leafs indicated overlap the PTV's projection in this beam's eye view. Other leafs would be excluded from beamlet calculations.	63
4.7	A log-linear plot of the number of histories in a single simulation related to the average relative uncertainties. The error bars represent the standard deviations of the average uncertainties.	67

4.8	The relative uncertainties corresponding to cumulative percentages of voxels for MC simulations run with 10^6 - 10^{10} histories. Note: EGSnrc has an uncertainties cutoff for those greater than 0.5, voxels associated with that high of an uncertainty are set to zero.	67
4.9	Relative dose color washes for a VMAT brain plan using (a) 10^6 , (b) 10^7 , (c) 10^8 , and (d) 10^9 histories. Noise is visible as "graininess" in each figure, where (a) has a considerable amount of graininess and (c) and (d) very grainy. Differences between these dose distributions and a reference dose distribution can be seen in Figure 4.10. The dose ranges from 0 to 37.6 Gy.	68
4.10	The noise contribution to MC dose calculations, which are shown as differences in dose, presented as color washes, for a VMAT brain plan using 10^{10} histories as a reference and (a) 10^6 , (b) 10^7 , (c) 10^8 , and (d) 10^9 histories. Ideally, there should be no color wash present, which indicates no difference to the reference. As the number of histories increases, the visible amount of noise decreases. The dose differences range from 0 to 4.7 Gy.	69

- 4.11 Relative dose color wash of a single beamlet in a prostate VMAT plan. (a) was obtained using 10^6 histories, and (b) was obtained using 10^7 histories. Noise is very apparent in (a) but is less apparent in (b). Increasing the number of histories ten fold decreased the variance in the data to the point where noise is not visible in the color wash. Note: noise visible outside the beamlet results from taking the difference between the unmodified and modified control point dose distributions, which is how beamlets are calculated. 71
- 4.12 (a) Sum of the beamlets calculated for a brain VMAT plan using 10^6 for each beamlet. (b) Difference in dose matrices between the unmodified VMAT plan and the same plan modified with the non-excluded MLC leaves moved out 0.5 cm. Both (a) and (b) use the same color wash scale. The PTV is contoured in teal, which is the general area where the beamlets converge. 72
- 4.13 (a) Sum of the beamlets calculated for a brain VMAT plan using 10^7 for each beamlet. (b) Difference in dose matrices between the unmodified VMAT plan and the same plan modified with the non-excluded MLC leaves moved out 0.5 cm. Both (a) and (b) use the same color wash scale. The PTV is contoured in teal, which is the general area where the beamlets converge. 73

4.14	The DVH curves for a VMAT brain plan PTV resulting from all non-excluded leafs being moved out by 0.5 cm. The solid curve resulted from a single simulation - modifying the original VMAT plan parameters. The dashed and dotted curves resulted from adding the calculated beamlets to the original plan's dose matrix. In this example, the 10^6 history beamlets overestimate the dose to the PTV by about 2.7 Gy while the 10^7 history beamlets underestimate it by about 0.4 Gy.	74
4.15	DVH curves for the first brain VMAT plan before (original) and after EO. The pituitary PRV improved, with a small increase to the 59.4 PTV. OAR's with objectives already met before and after EO are not shown.	78
4.16	DVH curves for the second brain VMAT plan. The cochlea, lacrimal, lacrimal PRV, and pituitary were improved by the EO. Note that the lacrimal and lacrimal PRV dose objectives are identical, and are very close to the right cochlea PRV objective.	80
4.17	DVH curves for the pediatric brain VMAT plan. There was a slight improvement to the left cochlea PRV.	81
4.18	DVH curves for the first head and neck VMAT plan. There was an improvement to the right cochlea and a small increases in both PTV doses.	81
4.19	DVH curves for the second head and neck VMAT plan. Improvements to the NTO planning structures (Rings) produced improvements to the left cochlea.	82

4.20 DVH curves for the first prostate VMAT plan, showing very little change with the EO. Note that the max dose objective for the bladder, rectum, and penile bulb are identical. The left and right femoral head dose objectives are also identical.	83
4.21 DVH curves for the second prostate VMAT plan, showing no change with the EO. Note that the femoral head objectives are identical.	84
4.22 DVH curves for the pediatric brain case after the EO was applied. "No Mod" indicates the original objectives and subsequent EO plan, while "Type 1" and "Type 2" indicates modifications to the left cochlea objective to make it progressively stricter. Type 1 plan was the same as the No Mod plan, as explained in Discussion. . .	85
4.23 Isodose curves for the pediatric brain case, showing the left cochlea PRV, which is shaded. The no-modification and Type 1 plans (a) are identical. The Type 2 plan (b) shows sparing of the left cochlea PRV; there is reduction in area covered by the 27 and 29 Gy isodose lines in favor of the 23 and 25 Gy isodose lines. Note that the isodose curves indicate a lower dose toward the center of the structure.	85
4.24 An enlargement of the rectum and PTV DVH curves for the prostate case after various modifications were made to the objectives. No Mod indicates the original objectives, while Type 1 and Type 2 objectives are progressively stricter. The Type 3 dose objectives were the same as Type 2, but with reduced priority weights. In every plan, reduction of the rectum dose resulted in compromised PTV dose homogeneity.	86

4.25	The MC-calculated DVH curves of the pediatric brain case's left cochlea PRV resulting from the original EO, type 1, and type 2 plans. The DVH's for the original EO and type 1 plans are identical, and the type 2 plan's DVH coincides with the other two at doses greater than about 33.1 Gy. Thus, the region under all three curves evaluated using the original and type 1 objective functions are identical and produce the same penalties.	90
4.26	DVH curves for the same dose matrices calculated using the same plan parameters before and after EO by (a) MC and (b) Eclipse. Differences are especially apparent for the rectum and penile bulb. Note that the max dose objective for the bladder, rectum, and penile bulb are identical. The left and right femoral head objectives are also identical.	91
5.1	Dose color wash for all five beams of an IMRT prostate patient with default weights. The pink contour is the PTV, the yellow is the bladder, the blue is the rectum, and the green are the femoral heads.	97
5.2	Beam's eye view of a single fluence (gantry at 288°) of an IMRT prostate patient with default weights. The fluence is represented by a grid pattern where each element is corresponds to a single PB. The whited-out elements do not contain at least part of the two-dimensional projection of the PTV, so there are no PB's associated with those elements. The pink structure is the PTV, the yellow is the bladder, the blue is the rectum, and the green are the femoral heads.	98

5.3	Beam's eye view of post-optimization fluence color wash of a prostate IMRT case at gantry angles (a) 0°, (b) 72°, (c) 144°, (d) 216°, and (e) 288°. Only the PTV structure is shown to illustrate that the fluence contours to the shape of the PTV projected to the corresponding beam's eye view.	99
5.4	DVH's resulting from optimizing with Eclipse's DVO (dashed curves) and the HERO (solid curves) for the first prostate case. The results using the regular objectives (Obj1) are in (a) while the results using the strict objectives are in (b).	104
5.5	DVH's resulting from optimizing with Eclipse's DVO (dashed curves) and the HERO (solid curves) for the second prostate case. The results using the regular objectives (Obj1) are in (a) while the results using the strict objectives are in (b).	104
5.6	DVH's resulting from optimizing with Eclipse's DVO (dashed curves) and the HERO (solid curves) for the third prostate case. The results using the regular objectives (Obj1) are in (a) while the results using the strict objectives are in (b).	105
5.7	DVH's resulting from optimizing with Eclipse's DVO (dashed curves) and the HERO (solid curves) for the fourth prostate case. The results using the regular objectives (Obj1) are in (a) while the results using the strict objectives are in (b).	105
5.8	DVH's resulting from optimizing with Eclipse's DVO (dashed curves) and the HERO (solid curves) for the fifth prostate case. The results using the regular objectives (Obj1) are in (a) while the results using the strict objectives are in (b).	106

5.9	DVH's resulting from optimizing with Eclipse's DVO (dashed curves) and the HERO (solid curves) for the sixth prostate case. The results using the regular objectives (Obj1) are in (a) while the results using the strict objectives are in (b).	106
5.10	DVH's resulting from optimizing with Eclipse's DVO (dashed curves) and the HERO (solid curves) for the seventh prostate case. The results using the regular objectives (Obj1) are in (a) while the results using the strict objectives are in (b).	107
5.11	Plots comparing the percent overdoses relative to the corresponding objectives averaged over the seven prostate IMRT plans. . . .	107
5.12	Log-log plots showing time to best solution versus objective function score for the HERO and Eclipse optimizations. (a) corresponds to the regular objectives while (b) corresponds to the strict objectives.	108
6.1	A flowchart summarizing the process the user follows, from exporting anonymized DICOM files to job array submission.	113

Nomenclature

3DCRT	Three-dimensional conformal radiotherapy
AAA	Analytical Anisotropic Algorithm
ART	Adaptive Radiotherapy
BEAMnrc	EGSnrc module for modelling linacs
CERR	Computational Environment for Radiotherapy Research
DICOM	Digital Imaging and Communications in Medicine
.djaws	Dynamic jaws BEAMnrc input file
DOSXYZnrc	EGSnrc module for phantom calculations
DVH	Dose-Volume Histogram
DVO	Dose Volume Optimizer
eComp	Electronic Compensation
.egsinp	Input file for DOSXYZnrc
EGSnrc	Electron Gamma Shower - National Research Council of Canada
.egsphant	Phantom file for DOSXYZnrc
EO	Enhanced optimization
EOF	Enhanced optimization function
Gray, Gy	SI unit of radiation dose - Joule/kg
GTV	Gross Tumor Volume

GUI	Graphical User Interface
HIPAA	Health Insurance Portability and Accountability Act
HU	Hounsfield unit
IMRT	Intensity Modulated Radiation Therapy
.jBLinfo	Beamlet information file
linac	Linear accelerator
.mlc	MLC BEAMnrc input file
MLC	Multileaf Collimator
MC	Monte Carlo
MMCTP	McGill Monte Carlo Treatment Planning (System)
MU	Monitor Unit
OAR	Organ at risk
PB	Pencil beam
PEGS	Preprocessor for EGS - files containing material data
PIPA	Point-in-polygon algorithm
POpR	Post-optimization Refinement (of MLC leaf positions)
PRO	Progressive Resolution Optimization
PRV	Planning organ at Risk Volume
PTV	Planning Target Volume
QA	Quantum Annealing (not quality assurance)
QUBO	Quadratic Unconstrained Binary Optimization
RCT	DICOM CT file
RD	DICOM dose file
RP	DICOM plan file
RS	DICOM structures file
RTOG	Radiation Therapy Oncology Group

SAD Source to axis distance
SSD Source to surface distance
SLURM Simple Linux Utility for Resource Management
TPS Treatment Planning System
VMAT Volumetric Modulated Arc Therapy
voxel Volumetric pixel element
UB CCR The University at Buffalo Center for Computational Research

Abstract

A TPS produces IMRT and VMAT plans by applying an optimization process to an objective function, followed by an accurate calculation of the final, deliverable dose. We developed two novel methods to optimize these plans. The first method involves using Monte Carlo (MC) routines. MC is considered to be the gold standard of dose calculation due to its high accuracy; however, it is currently too slow for practical comprehensive VMAT optimization. Instead, we developed an approach called *enhanced optimization* (EO), which employs the TPS VMAT plan as a starting point, and applies small perturbations, called beamlets, which are calculated using MC, to nudge the solution closer to a true objective minimum. A significant framework was developed in order to efficiently run parallelized MC simulations. The second method involves optimization of IMRT plans using new computer hardware: the Hamiltonian Engine for Radiotherapy Optimization (HERO). For both proposed novel methods, DICOM files for clinical VMAT plans files are exported from the TPS and used to generate input files for the EGSnrc MC toolkit or the HERO. For EO, a simple greedy search algorithm is applied to minimize the objective function, and the resulting modified control point parameters are imported into the TPS to calculate the final, deliverable dose, and to compare the EO plan with the original. EO produced improved objective scores (by 6% to 60%) and DVH's for

the brain plans and the head and neck plans. Although EO also reduced the objective scores for the prostate plans (by 46% and 79%), their absolute score and DVH improvements were not substantial. Further development will reduce the EO beamlet computation time and result in more sophisticated EO treatment planning methods. The HERO was able to quickly (under 45 seconds each) find solutions to seven prostate IMRT plans and produced objective scores 93.8% to 99.3% lower than the TPS objective scores. In the future, these novel methods will be useful clinically in finding more optimal solutions to IMRT and VMAT optimization problems.

Chapter 1

Introduction

It's showtime.

Ben Richards - The Running Man
(1987)

1.1 Cancer and Radiation Therapy

Cancer is one of the leading causes of death globally. Because of this, research into the treatment of cancer is one of the leading fields in medical research. Treatment modalities include surgical resection, chemotherapy, and radiation therapy.

Cancer is the uncontrolled replication of cells, which is caused by mutations in the cells' DNA. These cells cluster into tumors, which consist of highly-metabolic replicating cells. When tumors get to be large in size, the innermost cells are no longer exposed to oxygen and become hypoxic. This leads to permanent cell death and necrosis of healthy tissues, which can result in organ failure and death.

Radiation therapy utilizes the DNA-damaging properties of ionizing radiation to treat cancerous tumors. Damaging a cell's DNA prevents it from being able to replicate itself, so the cell dies. Since tumors tend to divide much more rapidly than the surrounding healthy tissue, they are more sensitive to radiation. Although some damage is done to nearby healthy tissue, radiation therapy preferentially targets tumor cells due to their increased radiosensitivity. Furthermore, a fractionated treatment scheme allows for healthy tissue cells to repair themselves. Additionally, since the outer oxygenated tumor cells are more radiosensitive than the hypoxic inner cells, they are preferentially killed by radiation. Once the outer cells are killed off, time is allowed for the inner tumor cells to reoxygenate, which drastically increases their radiosensitivity. The next treatment of radiation targets those cells, kills them, and exposes an additional layer to oxygen, and the cycle repeats.

1.2 Monte Carlo Radiation Dose Calculation

Monte Carlo (MC) dose calculation is a stochastic computational method that determines the behavior of a macroscopic system by averaging microscopic events, or histories (1). For radiation dose calculations, these histories include particle interactions and their associated probabilities. The AAPM TG-105 Report (2) indicates that MC methods are more accurate than the conventional dose calculation algorithms employed by typical treatment planning systems. This is particularly true in heterogeneous media, where the accurate modeling of electron transport is especially challenging. In one study (3), MC calculations were used, together with a direct-aperture optimization algorithm, to create IMRT plans in heterogeneous low-density media (e.g., lung tissue).

Typically, secondary monitor unit (MU) verification is performed with dose algorithms which are less accurate than those of the TPS. However, due to advances in computing techniques, TG-114 (4) recommends more sophisticated algorithms for this purpose. A study (5) compared MC VMAT calculations, employing vendor-provided phase space data, to TPS calculations, and found agreement to within 2% in high-dose regions. Another study involved comparison of an MC method to Eclipse's AAA for VMAT calculations, and found absolute dose value agreement to within 3% (6).

1.3 Organization of this Dissertation

This dissertation is composed of three parts: A Framework for Parallelized Monte Carlo Simulations, Enhanced Optimization of VMAT Plans Using Monte Carlo Generated Beamlets, and Application of the Hamiltonian Engine for Radiotherapy Optimization on Prostate IMRT Plans.

Each part is formatted into the following sections: motivation, methods and materials, results, and discussion. The motivation section provides background information specific to the project as well as motivations for doing the project. The Methods and Materials section describes, in detail, how the project was carried out. The results section provides a summary of the outcomes, and the discussion section analyzes the outcomes and relates them to the motivations.

Chapter 2 provides background on inverse planning and optimization of dynamic radiotherapy plans. Chapters 3 through 5 detail the projects done for this dissertation. Chapter 6 provides conclusions for each of the projects.

1.4 Publications and Conference Presentations

1. J. Mathews, S. French, S. Bhagroo, A. Pant, and D. Nazareth; Enhanced Optimization Of Volumetric Modulated Arc Therapy Plans Using Monte Carlo Generated Beamlets (Submitted to Medical Physics March 2020, currently revising).
2. M. Malekmohammadi, K. Dabiri, J. Mathews, D. P. Nazareth, H. Tamura, and A. Sheikholeslami; A Hamiltonian Engine for Radiotherapy Optimization (accepted for presentation - 2020 42nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) in conjunction with the 43rd Annual Conference of the Canadian Medical and Biological Engineering Society (EMBC 2020) at the Montréal, Canada, on July 20-24, 2020.)
3. S. Bhagroo, S.B. French, J.A. Mathews, and D.P. Nazareth; Secondary monitor unit calculations for VMAT using parallelized Monte Carlo simulations, J. Appl. Clin. Med. Phys. 20(6), 60-69 (2019)
4. J. Mathews, S. French, S. Bhagroo, and D. Nazareth; Secondary optimization of VMAT plans using Monte Carlo beamlets. Med. Phys. 46(6), e473 (2019).
5. J. Mathews, S. French, S. Bhagroo, and D. Nazareth; Enhanced Optimization of VMAT Plans Using Monte Carlo-Based Modifications to Control Points, semi-annual meeting of the Upstate NY Chapter of the AAPM, Buffalo, New York, April 12, 2019.
6. J. Mathews, S. French, S. Bhagroo, and D. Nazareth; Secondary Optimization of VMAT Plans Using Monte Carlo Beamlets, American Association of Physicists in Medicine (AAPM), 61st annual meeting, San Antonio, Texas

(July 17, 2019).

7. J. Mathews, S. French, and D. Nazareth; Enhanced Optimization of VMAT Plans Using Monte Carlo – Based Modifications to Control Points, American Association of Physicists in Medicine (AAPM), 61st annual meeting, San Antonio, Texas (July 17, 2019).
8. J. Mathews, S. French, and D. Nazareth, “Secondary Optimization of VMAT Plans Using Monte Carlo Beamlets.” Fall Upstate NY Chapter Meeting 2017, 17 Nov. 2017, Rochester, NY, Rochester Regional Health.

Chapter 2

Inverse Planning and Optimization of Dynamic Radiotherapy Plans

Get to the chopper!

Major Alan "Dutch" Schaefer -

Predator (1987)

2.1 Using Multiple Beams and Arcs to Mitigate Dose to Healthy Tissues

The primary goal of radiation therapy is to deliver a prescription radiation dose to a target volume in the patient, which is typically a cancerous tumor. Ideally, only the target volume would receive radiation dose while the surrounding healthy tissues would receive none. This is, of course, not possible, as radiation delivers dose along its path. Instead, the goal of treatment planning is to generate a plan that delivers the prescription dose to the target while sparing the

healthy tissues as much as possible. The process of finding the best possible distribution of dose that satisfies this goal is known as *optimization*.

To mitigate the dose to healthy tissue, multiple beams that converge on the target volume are used. The dose propagation in the volume where the beams converge is cumulative, so it follows that more beams leads to more mitigation of dose to healthy tissue as the contribution of more beams spreads the dose out more evenly. Treatment modalities like three-dimensional conformal radiotherapy (3DCRT) and intensity modulated radiotherapy (IMRT) utilize this multiple beam concept.

There is an inherent trade off between the number of beams used in a plan and the plan's efficiency, though. The addition of multiple beams increases the treatment delivery time, which includes patient setup, moving the gantry and collimator, verifying the setup parameters, and delivering the beam. Delivering more and more beams may spread the healthy tissue doses, but there are diminishing returns to this benefit as well. Arc therapy was the answer to this issue (7); the beam remains on while the gantry rotates around the patient. This allows for maximum spread of the healthy tissue doses while maintaining treatment efficiency; only one setup is required for each arc. Examples of dose distributions for a multiple beam plan and an arc plan can be seen in Figure 2.1.

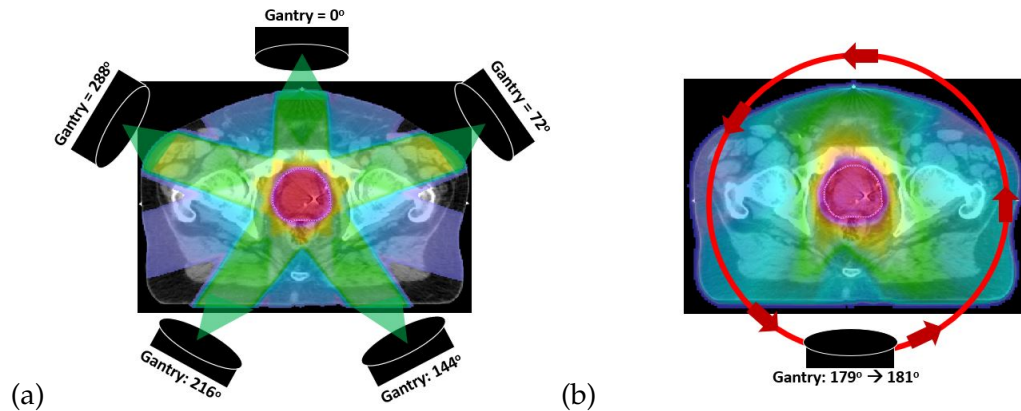


Figure 2.1: A prostate case with relative dose color washes for a five-beam plan (a) and an arc plan (b). For the five-beam plan, the gantry does not move during beam delivery. Depending on the type of treatment, the beam's shape and intensity may be modulated, but the beam's direction will not change. For an arc plan, the gantry rotates around a fixed point, known as the isocenter, and continuously delivers beam as the gantry moves.

2.2 Beam Collimation

2.2.1 The Jaws

All radiotherapy plans involve beam collimation by the jaws. The jaws are a set of thick rectangular metal slabs located between the target and the isocenter in the path of the beam. Refer to Figure 2.2 for a graphical representation of the jaws (and MLC). Linacs typically have two sets of jaws with one set moving in and out in the x direction and the other in the y direction. The y jaws are typically located above the x jaws. The jaws can create rectangular shaped fields only. Rather than moving along a straight line, the jaws move along a semi-arc path such that the edge of the jaws aligns with the edge of the beam. This is to prevent beam penumbra caused by the edges of the beam passing through different thicknesses of barrier material (i.e., the inner part of the beam edge passing through less material than the outer part).

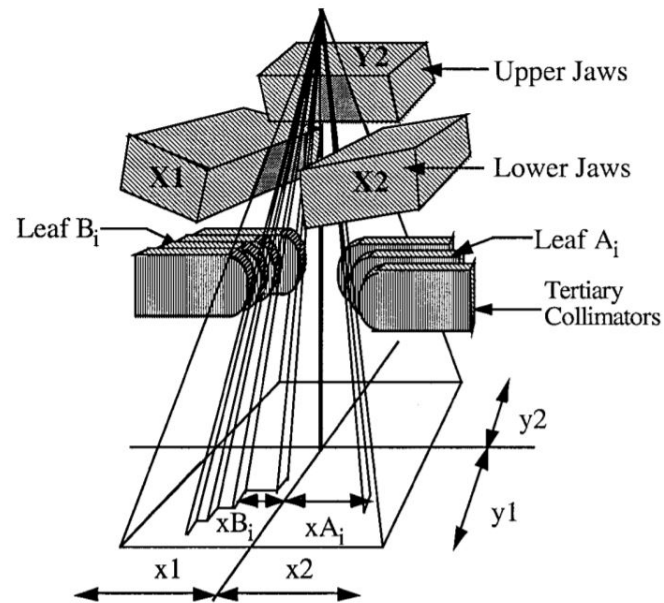


Figure 2.2: A graphical representation radiation beam collimation. The jaws move along a half arc in and out of the beam's path to create rectangular-shaped beams. The leaves move in and out of the beam's path to create beams of irregular shapes. Image from (8)

2.2.2 The Multileaf Collimator

An essential device that makes inverse planning possible is the multileaf collimator (MLC). The MLC is composed of thin leaves that can be moved in and out of the beam's path to change its shape. For dynamic plans, these leaves are moved while the beam is on to continuously change the shape of the beam and obtain the optimal dose distribution found during treatment planning. The beam is not necessarily completely open at certain points while the beam is on, so OAR's in the path of the beam can be blocked by the MLC leaves while at least part of the target volume receives radiation dose. An schematic of MLC leaves can be seen in Figure 2.3. MLC leaf positions at different points in the treatment are determined during the optimization step of treatment planning. Once the objectives are defined and initial treatment parameters are set, the optimization algorithm

attempts to find the best solution that satisfies a function like Equation (2.1) by making changes to parameters such as the MLC leaf positions and testing the outcomes.

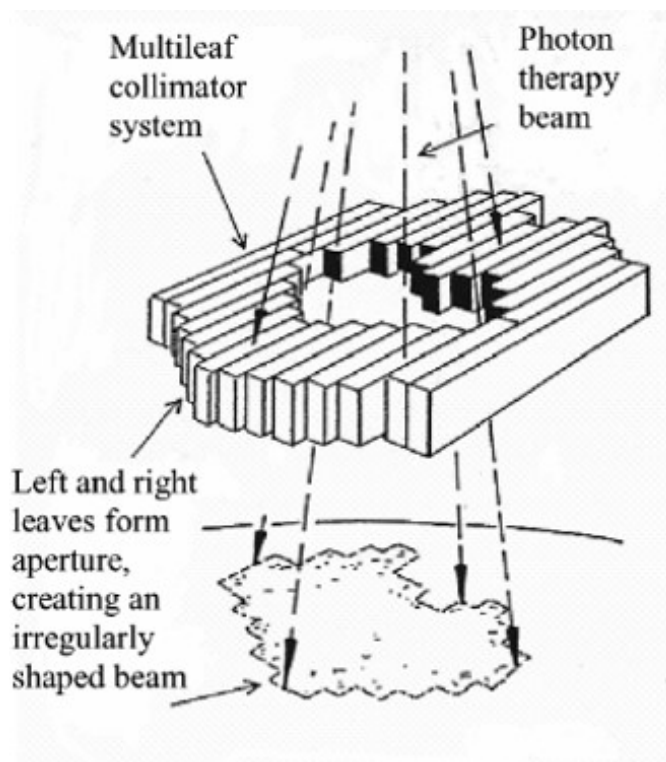


Figure 2.3: A graphical representation of MLC leaves. The leaves move in and out of the beam's path to create beams of irregular shapes during the treatment. Image from (9).

2.3 DICOM Files

Digital Imaging and Communications in Medicine (DICOM) files are the standard for communicating medical information across oncology, dentistry, ophthalmology, and other medical practice types (10). There are multiple types of DICOM files, and each type of file contains a very specific set of information related to a patient and their treatment (11). Each DICOM file contains meta-

data, which includes the type of DICOM file and the patient's information (e.g., name, date of birth, identification number). This prevents the DICOM file from accidentally being associated with the incorrect patient or treatment plan. The "image" of the DICOM file is the set of data the file contains, which could be pixel-by-pixel (or voxel-by-voxel) information or some other type of necessary attribute being conveyed. An example of part of a DICOM file can be seen in Figure 2.4.

```
(300a, 00b0) Beam Sequence 2 item(s) ----
(0008, 0070) Manufacturer LO: 'Varian Medical Systems'
(0008, 1090) Manufacturer's Model Name LO: '2300IX'
(0018, 1000) Device Serial Number LO: '1270'
(3002, 0050) Primary Fluence Mode Sequence 1 item(s) ----
(3002, 0051) Fluence Mode CS: 'STANDARD'
-----
(300a, 00b2) Treatment Machine Name SH: 'Linac3-1270'
(300a, 00b3) Primary Dosimeter Unit CS: 'MU'
(300a, 00b4) Source-Axis Distance DS: '1000'
(300a, 00b6) Beam Limiting Device Sequence 3 item(s) ----
(300a, 00b8) RT Beam Limiting Device Type CS: 'ASYMX'
(300a, 00bc) Number of Leaf/Jaw Pairs IS: '1'
-----
(300a, 00b8) RT Beam Limiting Device Type CS: 'ASYMY'
(300a, 00bc) Number of Leaf/Jaw Pairs IS: '1'
```

Figure 2.4: A textual representation of part of a DICOM file. Specifically, this shows the information about the linear accelerator, source to axis distance, and beam limiting devices (i.e., x and y jaws - the MLC information is not shown in this figure). On the left-most side are alphanumeric keys that can be called by the DICOM reader, and the indentations indicate nested data. For example, should the user want to access the number of jaw pairs for the Y jaws, they would need to use the alphanumeric key: (300a,00b0),(300a,00b6),(300a,00bc), which would fetch "1", indicating that there is 1 pair of Y jaws. Every key piece of information regarding a treatment plan can be obtained with this method from the DICOM files.

DICOM files are formatted as binary, not text, so they cannot be simply opened in a text editor. An image interpreter is required to read the information. Programming languages like Python (Pydicom (12)) and Matlab (Image Processing Toolbox (13)) can extract information from DICOM files. DICOM files utilize a nested data structure. That is, information is organized in layers,

Table 2.1: Summary of contents of DICOM files. Although other types of DICOM files are used in radiation oncology, the types included in this table are the only ones needed for the work detailed in this dissertation.

File Type	Abbreviation	Contents
Dose	RD	Voxel doses, DVH's
Structure	RS	Coordinates of structure contour vertices
Plan	RP	Plan parameters at each control point
Image	RCT	Voxel CT data

where objects similar in nature are grouped together. This makes calling data relatively simple as each item has a specific and unique address within the DICOM file.

For the work detailed in this dissertation, four types of DICOM files are needed to generate the necessary files for running the MC simulation as well as evaluating the OF. How each of these files is specifically utilized will be explored in detail in the following sections of this chapter. Dose DICOM (RD) files contain voxel-by-voxel dose information. This includes the dose matrix dimensions and the coordinate bounds for the voxels on each axis (x , y , and z) with respect to the isocenter, which is considered to be the origin (i.e., x , y , and $z = 0$). Structure DICOM (RS) files contain slice-by-slice vertices coordinates for each structure defined with contours during treatment planning. Plan DICOM (RP) files contain plan parameters as defined at each control point (e.g., gantry and MLC angles, MLC leaf positions, dose rate). Patient image DICOM (RCT) files contain voxel-by-voxel CT image information. Radiation therapy plans can be represented by one RD, one RS, and one RP file. RCT files are formatted such that each only contains information about a single slice; thus, the number of RCT files per plan is determined by the number of slices obtained during imaging. Refer to Table 2.1 for DICOM file types, abbreviations, and summaries of what each contains.

2.4 Dose-Volume Objectives and Objective Function Optimization in Inverse Planning

For simplicity in planning, only anatomical organs near the target volume are considered during treatment planning optimization. These are referred to as organs at risk (OAR's). OAR's have varying sensitivities to radiation depending on their composition and functionality. The Radiation Therapy Oncology Group (RTOG) has compiled data from many studies to form a set of guidelines to follow when evaluating risk to various OAR's. These guidelines include dose-volume objectives: thresholds indicating how much radiation dose is acceptable to be received by a relative percentage of an OAR.

There are two types of planning - forward planning and inverse planning. For forward planning, the treatment planner defines the initial beam parameters (e.g., shape, intensity), calculates the resulting dose, and evaluates the results to some clinical objectives. Should the plan need adjustment, the planner can make modifications to the plan parameters, calculate, and reevaluate the results. Static field treatments are forward planned.

Inverse planning requires much more computation. A set of user-defined objectives (i.e., dose-volume objectives) as well as preliminary beam parameters (e.g., gantry and collimator angle) are supplied to an optimization algorithm, which then attempts to locate the best solution that fits the objectives as closely as possible. During the optimization process, the algorithm selects and tests a wide array of plan parameters. Many adjustments are made while searching for the solution. Dose distributions are computed quickly as well as the objective function. Optimization typically terminates after a certain number of search attempts (i.e., iterations) or a certain amount of time has passed.

A widely-used objective function in radiotherapy optimization research (14) is:

$$f(w_D) = \sum_j W_j \sum_i (d_{j,min} - d_i(w_D))^2 * H(d_{j,min} - d_i(w_D)) + (d_i(w_D) - d_{j,max})^2 * H(d_i(w_D) - d_{j,max}) \quad (2.1)$$

$$H(d) = \begin{cases} 1 & d \geq 0 \\ 0 & d < 0 \end{cases} \quad (2.2)$$

where w_D is the set of parameters (e.g., MLC leaf positions) for the plan. This function quadratically adds the dose objective violations, $d_{j,min} - d_i(w_D)$ for minimum dose objectives and $d_i(w_D) - d_{j,max}$ for maximum dose objectives, over individual voxels, i , which violate dose objectives, $d_{j,max}$ or $d_{j,min}$, and then sums all of the penalties into a single score. The Heaviside function (Eq. (2.2)) limits the penalty to the range of doses which violate the objective (refer to Figure 2.5 for a visual representation). W_j is a priority weighting given to individual objectives. If a calculated dose-volume meets an objective, no penalty is incurred.

The treatment planning system (TPS) typically applies an optimization algorithm to minimize the function $f(w_D)$. During this process, the dose distribution to the patient must be calculated every time $f(w_D)$ is evaluated. Since this calculation occurs a large number of times, it must be performed very rapidly, typically within the time frame of milliseconds (7). Eclipse Version 15.6 (Varian Medical Systems, Palo Alto, CA) employs a pencil beam dose calculation algorithm, which is relatively inaccurate, to determine dose while optimizing (15).

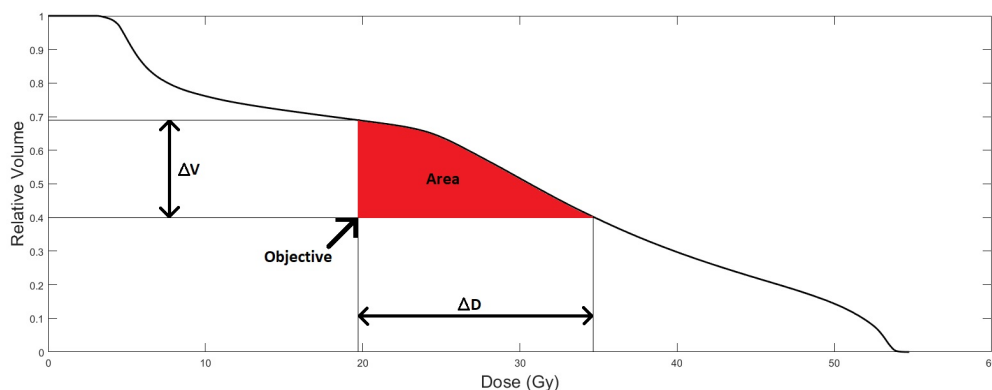


Figure 2.5: A visual representation of the area under the dose-volume histogram curve used for the penalty of dose-volume objectives. The penalty scales with the volume, ΔV , and dose, ΔD , that violates the objective.

Once the optimization process is complete, an accurate dose calculation (using Acuros or the Anisotropic Analytical Algorithm) is performed to provide the final, deliverable dose distribution (15).

Once the optimization step has been completed, the TPS provides the planner with a dose-volume histogram (DVH). A DVH is a convenient way of including the dose-volume data and objectives on a single graph (16). An example of a DVH with objectives is shown in 2.6. This DVH contains the dose volume statistics of a head and neck VMAT plan that was accepted and used clinically. Note that not all of the objectives were met; particularly the parotid (right) objective, in yellow, is an example where an objective was not met during optimization. The plan associated with this DVH was used clinically. Optimization will not always find a solution that satisfies every objective; a solution doing so may not even exist. It is common for plans to be accepted for clinical use with one or more objectives being unmet; it is left to the judgement of the clinician (i.e., the radiation oncologist).

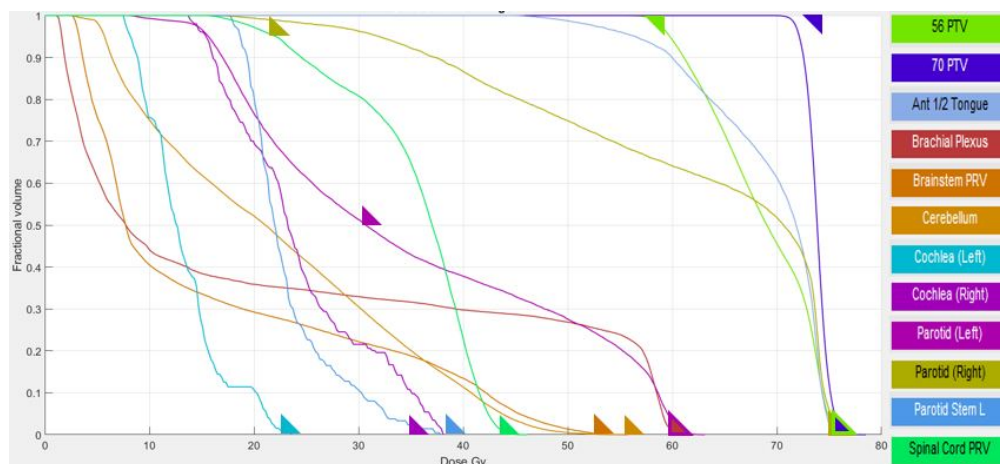


Figure 2.6: An example of a DVH belonging to a head and neck VMAT case. Objectives are denoted by triangles where the right-angle point corresponds to the objective.

2.5 IMRT Treatment Plan Optimization

IMRT optimization is performed in two steps. The first step is to optimize the fluences of each beam. Beams are discretized into pencil beams (PB). The fluence of a single beam is the collection of PB weights. During optimization, PB's are typically calculated with a fast algorithm, and each PB has its own weight. A PB's weight corresponds to the intensity of that PB relative to other PB's. Each PB and its weight is independent of the other PB's.

The second step in the optimization process is to sequence the MLC leaves and dose rate. As the leaves have finite size and speed, producing the PB's as indicated by the ideal fluence map may not be possible. The sequencing step itself is a separate optimization process as it attempts to find MLC leaf movements that produce a fluence map as close to the ideal fluence map as possible. Once leaves are sequenced, they will typically move from one side of the field to the other at varying speeds while the beam is on, and they will only move in one direction. This is known as "sliding window".

2.6 VMAT Treatment Plan Optimization

Optimization of VMAT treatment plans is significantly more complicated than it is for IMRT plans. The complication comes from the rotation of the gantry during treatment; since the gantry rotation is continuous, parameters (e.g., MLC leaf positions) at each point along the arc are dependent on previous and future points. For example, the MLC leaves have a finite speed, so their positions are physically constrained from point-to-point based on their previous positions.

Karl Otto published a paper describing his newly developed optimization algorithm for VMAT (7). Otto's VMAT algorithm consists of breaking down an arc into individual samples known as "control points". At first, the algorithm defines only a few control points; however, as it iterates, it moves from coarse to fine sampling by increasing the number of control points until the desired sampling frequency is reached. Refer to Figure 2.7 for a schematic of this process. In addition to Otto's method, a variety of optimization algorithms have been adapted for VMAT such as greedy, simulated annealing, and genetic.

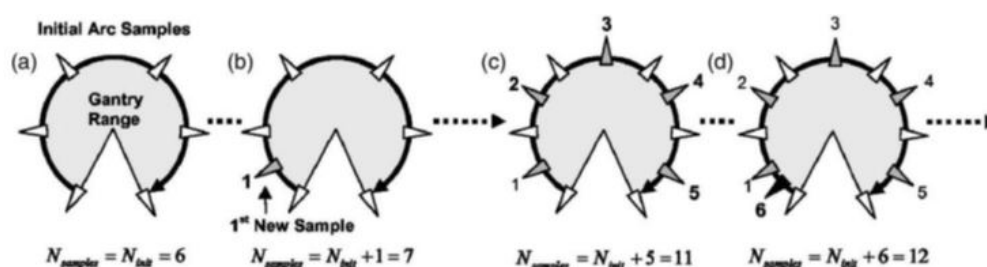


Figure 2.7: A graphical representation of Otto's VMAT optimization algorithm. The algorithm optimizes a few initial control points (the white triangles) (a), adds additional interpolated control points (the grey triangles) (b and c), does another round of optimization, and then adds even more control points (the black triangle) (d) until the desired sampling frequency (i.e., maximum number of control points) is reached. Image from (7)

Clinically, Eclipse uses a Progressive Resolution Optimization (PRO) algo-

rithm. The PRO is based on Otto's technique, with some refinements over the course of several releases. It begins with coarse sampling (few control points) and progressively adds more as the optimization continues. The PRO utilizes a pencil beam dose calculation method during optimization. Periodically, the PRO performs an accurate dose calculation during optimization, but the majority of the dose calculations are performed using the faster pencil beam technique (15).

2.6.1 Enhanced Optimization

Treatment planning systems tend to use pencil-beam dose calculation algorithms while performing plan optimization. Since these dose algorithms are relatively inaccurate, using them does not necessarily model the objective function landscape correctly. Even in the case of Eclipse's PRO algorithm, which periodically performs an accurate dose calculation during optimization, the majority of the dose calculation is performed using the faster pencil beam technique (15). Therefore, the function which is minimized may be substantially different from the true objective function, which would be obtained with a more accurate dose algorithm. This results in the function minima, whether local and global, being inaccurately identified even with an effective optimization algorithm and an accurate final, deliverable dose calculation. This situation is represented schematically in Figure 2.8. This opens the possibility of improving the optimization process by beginning with the TPS-optimized plan, which lies in an approximate local minimum, and searching this neighborhood for a plan whose score would be lower when calculated with an accurate dose algorithm. Much work has been done in evaluating (17; 18; 19; 20) and improving (21; 22; 23; 24) VMAT

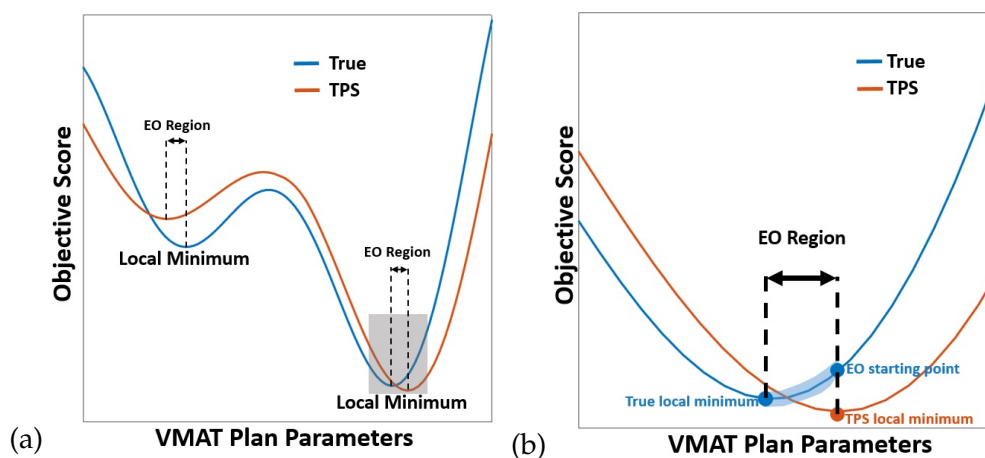


Figure 2.8: A simplified 1-D representation of the objective space for a VMAT optimization problem. The orange curve is the score calculated by Eclipse during optimization, while the blue curve is the true score. After Eclipse locates an approximate local minimum, it is used as the starting point for the EO. The EO searches the neighborhood using a more accurate dose (and therefore score) calculator. (b) is an enlargement of the region of (a) indicated by the grey box.

optimization algorithms. To the best of our knowledge, however, there has been no investigation into this type of post-optimization VMAT improvement.

For the work described in Chapter 4, *Enhanced Optimization* (EO) is defined as the process of making further refinements to a VMAT plan which has been optimized by a TPS. A concept along these lines was investigated for IMRT plan optimization by Niu *et al* (25). Niu used simple beamlet calculations to estimate perturbations to an IMRT plan's dose by moving MLC leaves in or out by 0.5 cm, a process referred to as post-optimization refinement (POpR). It was found that, by employing a greedy search algorithm, an IMRT plan could be improved quickly. However, an analogous process for VMAT involves a much larger number of control points and beamlets. In addition, a more accurate method for computing beamlet dose matrices would improve the quality of the resulting plan. These are the benefits of EO.

Figure 2.8 shows a simplified representation of both the true solution space

and the space calculated by the TPS during VMAT plan optimization. The objective function value is sketched against the VMAT plan parameters. Since the TPS employs an inaccurate pencil beam algorithm during this process, its score (orange curve) is only an approximation of the true score (blue curve). Note that the blue curve is an idealized function, because it is not possible in practice to calculate the true objective function exactly. However, MC simulations are the gold standard in dose calculation accuracy (26; 27; 28), and therefore will closely match the true values. The TPS creates the VMAT plan by locating a minimum (local or possibly global) of the orange curve. The Enhanced Optimization process uses those VMAT parameters as a starting point, performs a search in its neighborhood using MC calculations, and attempts to locate a nearby minimum of the blue curve. This neighborhood is indicated in the figure as *EO Region*.

Chapter 3

A Framework for Parallelized Monte Carlo Simulations

You should not drink and bake.

Mark Kaminski - Raw Deal (1986)

3.1 Motivation

3.1.1 Monte Carlo Simulations

The data from thousands of segments of VMAT Monte Carlo (MC) simulations were needed for the methods presented in Chapter 4. Prior to this work, there was no efficient and scalable way of running VMAT MC simulations available. Because of this, it was essential to develop an adaptable and easy-to-use method. This includes, among other things, development of automated input file generation, automated commands, and simple tools for MC data analysis.

In addition to the work presented in this dissertation, having access to simple-

to-run MC simulations is clinically useful. For example, in the work done by Bhagroo (6), MC can be used as a secondary dose calculation check that is independent of the treatment planning system (TPS). Secondary dose calculations are necessary for quality assurance of radiation therapy plans as they act as a verification of the TPS dose calculation. In instances where current secondary check software may be inaccurate, an MC secondary check has been shown to be a viable alternative in a proof-of-concept study. Developing tools that simplify and automate these MC simulations was the logical next step toward making an MC secondary check application usable in the clinic.

In a broader application, developing a way for inexperienced users to easily and conveniently run MC simulations could provide them with an extremely useful tool for data verification in their research. Running Monte Carlo simulations is often not an intuitive process. There are many pitfalls the user may encounter, and the learning curve is very steep, which may lead to those desiring MC data to bolster their work to simply forgo obtaining it as the effort of learning how to accurately and effectively run MC simulations may not be worth the payoff. To that end, one of the motivations of the work detailed in this chapter was to adapt the protocols developed for Enhanced Optimization and secondary MU checks to form a relatively easy-to-use method for running electronic compensation (eComp), IMRT, and VMAT Monte Carlo simulations with minimal user intervention.

3.1.2 Independent Calculation of Dose Volume Histograms for Optimization

Another task necessary for the completion of work detailed in Chapter 4 was to develop a way of independently calculating dose-volume data for use in rapid calculations of an objective function. To that end, developing a method of calculating the dose to specific anatomic structures is necessary. The method developed and used for the work detailed in this dissertation involves the application of binary masks, the concepts of which will be explained in the next section.

3.2 Methods

3.2.1 Anonymized DICOM File Export

The Health Insurance Portability and Accountability Act (HIPAA) (29) mandates the protection of patients' personal information. This project utilizes computing resources that are *not* secured under HIPAA guidelines. Because of this, it is necessary to anonymize patient data prior to processing DICOM files using these resources.

Eclipse, the TPS used for this project, contains a built-in module that allows for streamlined scrubbing of patient identifying information during DICOM file export. Patient information to be scrubbed is selected, and the option to replace it with a pseudo identifier is available. To track the patient using the anonymized DICOM files, a cipher is kept on a HIPAA compliant secured network. This cipher links the real patient identities to their associated pseudo identifiers used off network.

3.2.2 EGSnrc User Codes and Input Files

The Electron Gamma Shower by the National Research Council of Canada (EGSnrc) script package was chosen for MC simulations for the work detailed in this dissertation. EGSnrc has several useful user modules that are capable of different types of MC simulations. For the work detailed in this dissertation, two modules were used: BEAMnrc and DOSXYZnrc. These modules require precise parameter inputs in order to run correctly, which are contained in a set of input files. Plans with multiple arcs (or fields) have one set of input files for each arc.

A majority of particles will be absorbed in the linear accelerator itself, so running simulations of the entire path of a particle is inefficient. Instead, it is more efficient to simulate the accelerator separately. Particles that make it out of the accelerator and can travel to the system containing the patient are saved. Particles that do not make it are discarded. This is done with the BEAMnrc user code (30). This includes the interactions of particles as they travel from the target through flattening filter, primary collimation (i.e., the jaws), and secondary collimation (i.e., the MLC).

Building an accelerator model requires obtaining and inputting the parameters of the accelerator; however, this was beyond the scope of this work as there was already a functional and tested accelerator model developed previously (31). Instead, this work focuses on the automated and user friendly generation of input files.

The BEAMnrc code calls two input files generated from DICOM file extracted information: the dynamic jaws position file (.djaws) and the MLC leaf position file (.mlc). Both of these types of files contain control point position parameters of each jaw and MLC leaf. These parameters are contained within

RP files.

The DOSXYZnrc user code (32) simulates the rest of the particles' paths as they travel beyond the collimation and through media. The interface where the simulation of the particle transport ends for BEAMnrc and begins for DOSXYZnrc is known as the scoring plane. These interactions include scatter, absorption, and secondary particle generation. DOSXYZnrc calls two files: the DOSXYZnrc input file (.egsinp) and the virtual phantom (.egsphant).

The DOSXYZnrc input file contains control point gantry angle, collimator angle, and dose rate information along with the coordinates of the isocenter's offset between the DICOM coordinate system and the Eclipse (user defined) coordinates. It also contains parameters that direct the DOSXYZnrc code such as the name of the BEAMnrc code to be used (i.e., the accelerator model), the number of histories, and MC transport parameters.

The virtual phantom contains the voxel-by-voxel composition and density of the physical media belonging to the plan. These data are applied to pre-defined interaction probability cross sections when simulating the interactions a single particle undergoes (e.g., scatter, energy absorption) during the MC simulation.

3.2.3 Parallelization of Monte Carlo Simulations

MC simulations by their nature are slow to run and computationally expensive; however, histories (i.e., the collection of interactions a single particle undergoes) are independent of each other. Because of this, MC simulations are highly amenable to being run in parallel. Not only can separate simulations (e.g., arcs belonging to the same VMAT plan) be run simultaneously, but simulations can be broken down into parts which themselves can be run in parallel.

As long as there is bookkeeping on which parts of which simulations are assigned to each piece of computer hardware, those parts can be recombined once the simulation is completed to provide the final result. This allows for MC simulations to be done much faster than if they were run simply on a single piece of hardware in series (i.e., one history at a time). EGSnrc has scripts built in that allow for parallelization.

The University at Buffalo Center for Computational Research (UB CCR) (33) houses a supercomputer cluster accessible to researchers at the university. The UB CCR's cluster uses Simple Linux Utility for Resource Management (SLURM) (34) to process submissions, known as "jobs". The most convenient method of organizing and scheduling a job is with a "job array", which contains instructions for SLURM to follow to complete a set of tasks.

3.2.4 Generation of EGSnrc Input Files

Preliminary methods utilized the McGill Monte Carlo Treatment Planning (MMCTP) system to generate input files (35; 36; 37; 38). Once the scope of the project detailed in Chapter 4 reached the point where many simulations-worth of input files needed to be generated, using MMCTP to generate input files was found to be too slow and cumbersome. Although MMCTP is an effective and comprehensive way of treatment planning research with Monte Carlo, a bulk of the program is not needed, and input file generation is only a small part of the large application.

Python scripts were written to generate DOSXYZnrc input files, MLC leaf position files, and dynamic jaws files. These scripts extract the necessary information from the exported DICOM files and generate text files that are formatted

such that the EGSnrc user codes can read them. Refer to Figure 3.1 for a summary of which DICOM files are needed for each input file.

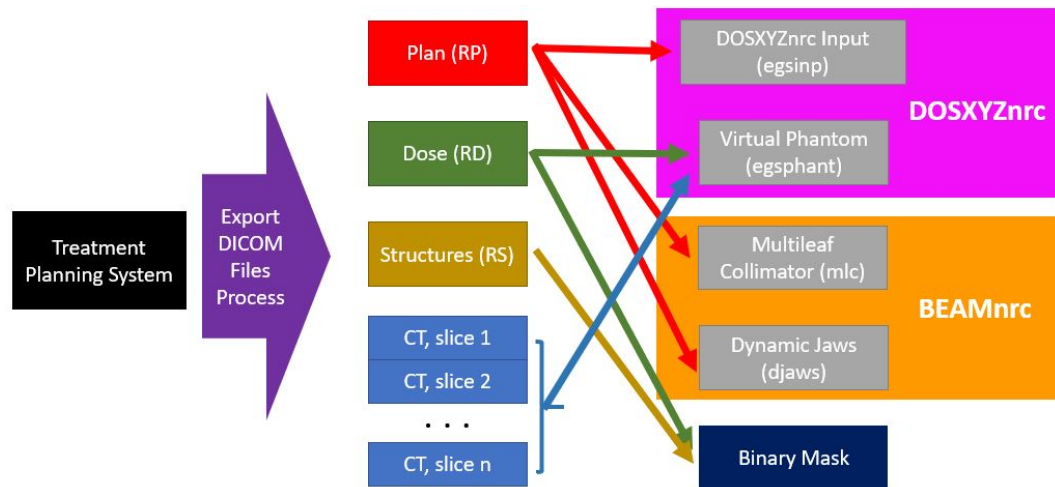


Figure 3.1: A flowchart showing the DICOM files needed to generate each type of EGSnrc input file or binary mask.

It should be noted that the accelerator model used for this dissertation was a Varian Trilogy. This model was developed as part of previous projects (31). Thus, its development was not part of the scope of the work done for this dissertation, so it will not be detailed here.

3.2.4.1 DOSXYZnrc Input File

The DOSXYZnrc input file (extension `egsinp`) contains parameters related to the treatment plan. This information is found in the RP file, and a Python script was written to obtain the necessary information and write a text file in the following format:

Block 1 - Header (one line)

Block 2 - Number of media in simulation ("0" if an `egsphnt` file is used)
(one line)

Block 3 - File path to the egspphant file (one line)

Block 4 - Cutoff energies (one line)

Block 5 - Dose output reporting options (one line)

Block 6 - Particle and source options (one line)

Block 7 - Control point by control point parameters: isocenter offsets (x, y, and z), gantry angle, couch angle, collimator angle, distance from isocenter to BEAMnrc scoring plane, MU index (i.e., meterset) (number of lines equal to the number of control points)

Block 8 - Beam spectrum options and virtual phantom options (one line)

Block 9 - Name of BEAMnrc accelerator, filename of BEAMnrc accelerator input file, name of PEGS file, optional non-EGSnrc inputs (unused, so "0") (one line)

Block 10 - Number of histories and miscellaneous options (one line)

Block 11 - MC transport options (multiple lines)

The various option choices used for the generation of input files here were set to their defaults as described by the DOSXYZnrc user manual (32). These options satisfied the needs of the work detailed in this dissertation, so they weren't changed.

3.2.4.2 MLC Leaf Position File

Leaf position information is stored in the RP file. Each control point contains a list of the coordinates of all MLC leafs with respect to the direction along which the leafs move. The positive direction is toward the B bank, so retracting (i.e., opening) a B bank MLC leaf or extending (i.e., closing) an A bank MLC leaf increases the coordinate value of its location. Conversely, extending a B bank leaf

or retracting an A bank leaf moves the leaf position negatively. Note: the leaf positions displayed in Eclipse follow a slightly different coordinate scheme. In Eclipse, the positive direction for each bank of leafs is toward the corresponding bank. That is, if leafs are moving outward, they are moving positively regardless of to which bank they belong. The Eclipse-style coordinates are automatically converted to traditional coordinates when exported via DICOM.

The coordinates stored in the RP file are *not* the physical coordinates of the MLC leafs. Rather, they are the coordinates of the leafs projected to isocenter. The MC code requires the physical locations of the MLC leafs. The method for converting the MLC leaf positions taken from the RP file is adapted from the method described in the MMCTP user manual (39).

Refer to Figure 3.2 for a schematic of the MLC leaf geometry. x is the physical distance of the leaf from the central axis whereas x' is the distance of the same leaf as projected at a plane located at isocenter (i.e., the coordinate given in the RP file). x can be represented as:

$$x = a + c - r \quad (3.1)$$

where a and c are represented by the pink and green , respectively, and r is the radius of curvature of the MLC leaf (8 cm), which is defined during modelling of the accelerator. Since the triangle containing lines a and z_{MLC} , which is the horizontal distance between the target and the center of the MLC leaf, and the triangle containing lines x' and SAD (i.e., source to axis distance, 100 cm) are similar,

$$\frac{a}{z_{MLC}} = \frac{x'}{SAD}$$

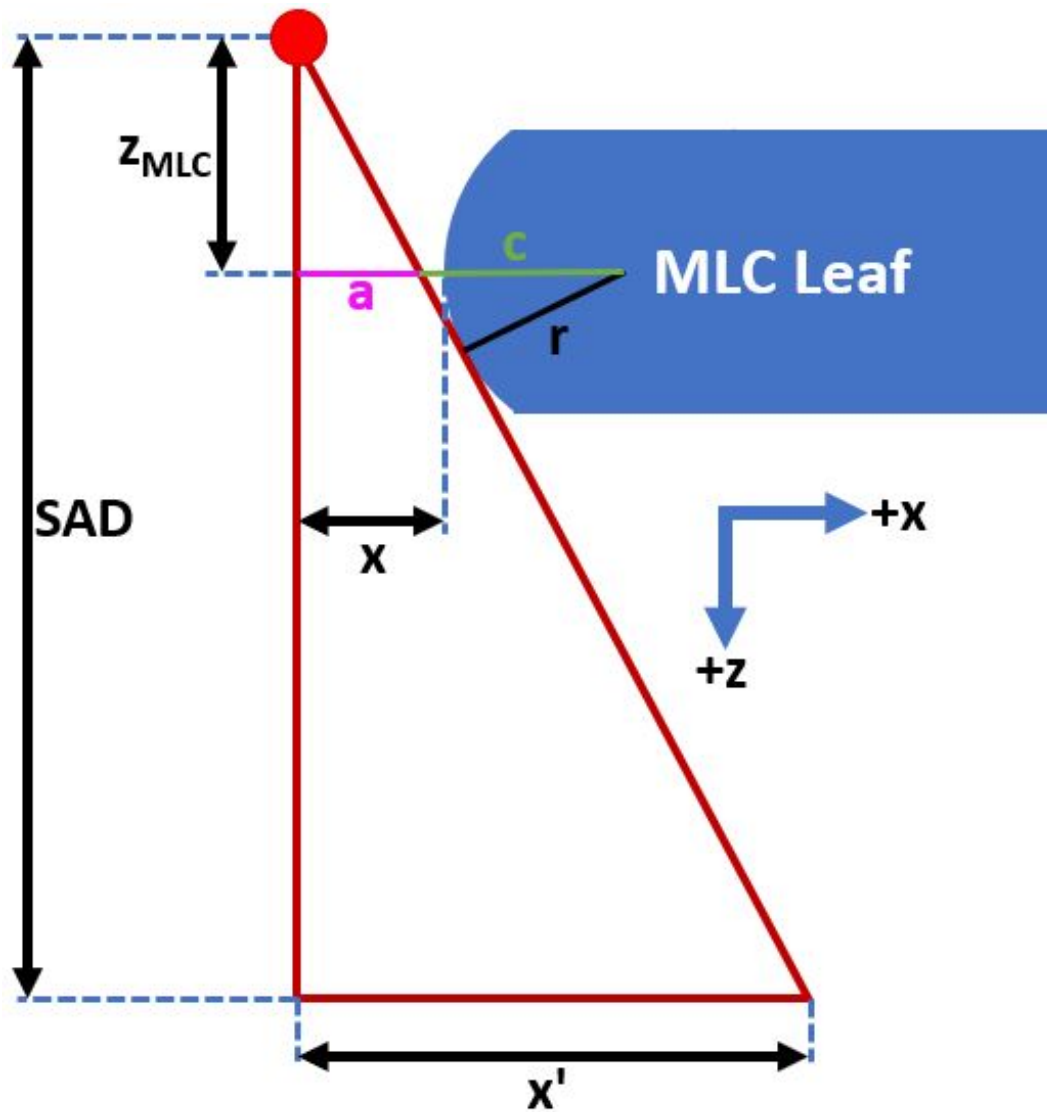


Figure 3.2: A schematic of the MLC leaf geometry as modelled by the BEAMnrc user code. The x ray source is indicated by the red dot, and the beam geometry is indicated by the red lines. Using the rule of similar triangles, physical horizontal distance between the central axis and the tip of the curve of the MLC leaf, x , can be represented in terms of the source to axis distance (SAD), the vertical distance between the target and the center of the leaf, z_{MLC} , and the coordinate of the leaf projected to a plane at isocenter x' .

$$a = \frac{z_{MLC} * x'}{SAD} \quad (3.2)$$

Likewise, the triangle containing lines r and c is similar to the triangle containing of lines a and z_{MLC} such that

$$\begin{aligned} \frac{c}{r} &= \frac{\sqrt{a^2 + z_{MLC}^2}}{z_{MLC}} \\ c &= r \frac{\sqrt{\left(\frac{z_{MLC} * x'}{SAD}\right)^2 + z_{MLC}^2}}{z_{MLC}} \\ c &= r \sqrt{\left(\frac{x'}{SAD}\right)^2 + 1} \end{aligned} \quad (3.3)$$

Combining 3.1 with 3.2 and 3.3:

$$x = \frac{z_{MLC} * x'}{SAD} + r \sqrt{\left(\frac{x'}{SAD}\right)^2 + 1} - r \quad (3.4)$$

A Python script was written to generate the MLC leaf position files used in a DOSXYZnrc MC simulation. The script generates these files in the following format:

Block 1 - Header (one line)

Block 2 - Number of control points (one line)

Block 3 - Meterset at control point #1 (one line)

Block 4 - MLC leaf positions at control point #1 (one line for each MLC leaf pair)

Block 5 - Meterset at control point #2 (one row)

Block 6 - MLC leaf positions at control point #2 (one line for each MLC leaf pair)

Block 7 to end - repeating the meterset at the next sequential control point followed by the MLC leaf positions at the corresponding control point.

The header is not read by DOSXYZnrc, but it is convenient for the user should they need to retrieve information from within. It is a string containing the type of file, "Dynamic MLC", patient pseudo identifier, arc number, and any other specifying information. The rest of the information is generated by extracting data from the RP file and converting it to parameters read by the EGSnrc codes. Since DOSXYZnrc simulates VMAT arcs independently, each arc has its own MLC file.

3.2.4.3 Dynamic Jaws Position File

The EGSnrc codes have the ability to simulate the jaws dynamically (i.e., the jaws can move during beam on); however, clinically the jaws remain fixed the entire duration of each field or arc. Because of this, the parameters of the jaws only need to be extracted and calculated once per arc. These parameters come from two sources: the modelling of the linear accelerator, which was done in previous projects, and the RP file.

To generate the dynamic jaws file, only two values need to be calculated per jaw (two pairs of jaws, so four jaws total). These values are defined as the distances of the top and bottom inner edges of the jaws from the central axis. Refer to Figure 3.3 for a schematic of how the jaws are modelled by the BEAMnrc code. Only the method for finding the x jaws values will be described in the following; however, finding the values for the y jaws is virtually the same. The only differences are the y jaws are located below the x jaws and move along an axis perpendicular to the axis the x jaws move along. Note: the movement of

the simulated jaws is different than the movement of real jaws. Simulated jaws only move along a single axis, and the shape of the jaws actually molds to conform to the outer edges of the beam. This is, of course, not possible in reality as the jaws are slabs of solid metal. Actual jaws move along an arc to attain a similar goal of conforming to the edges of the beam.

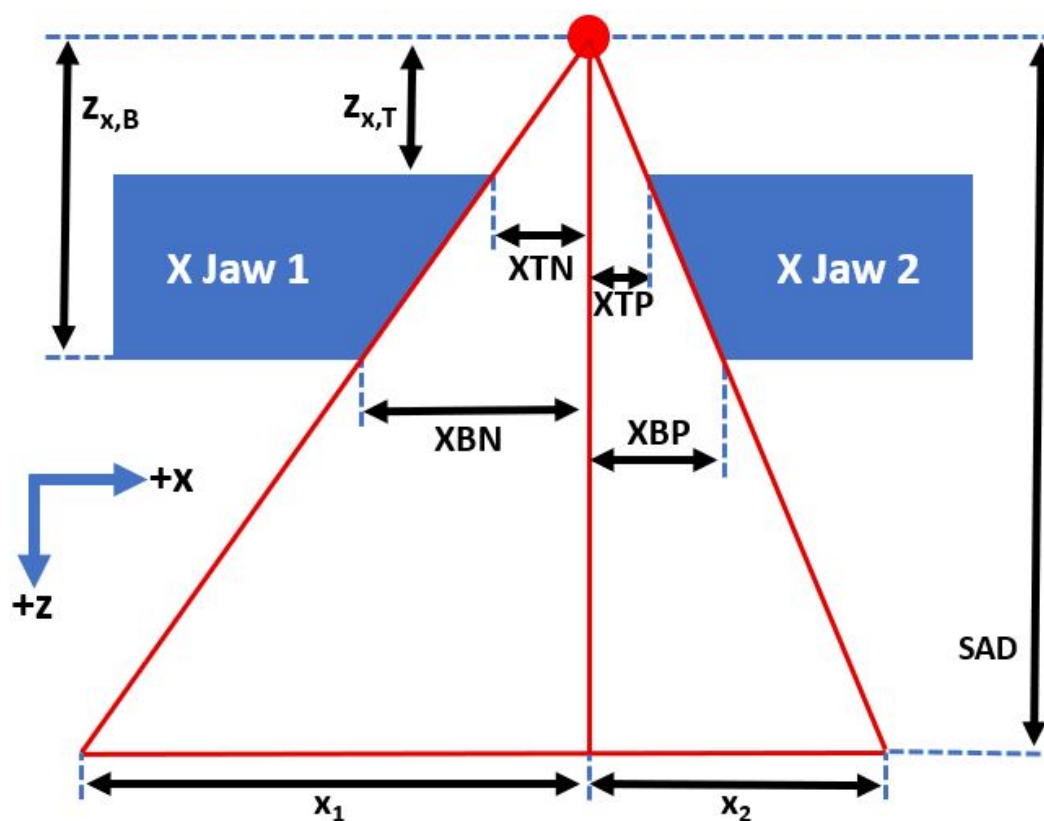


Figure 3.3: A schematic of the x jaws as modelled by the BEAMnrc user code. The x ray source is indicated by the red dot, and the beam geometry is indicated by the red lines. Using the rule of similar triangles, the values of XTN , XBN , XTP , and XBP , which are needed for the dynamic jaws input file, can be solved for in terms of $z_{x,T}$ and $z_{x,B}$, the vertical distances between the x ray source and top and bottom of the jaws, respectively; x_1 and x_2 , the coordinate of each jaw projected to a plane at isocenter; and SAD , the source to axis distance (100 cm).

Similar to the MLC leaf positions, the values given by the RP file are the coordinates of each jaw projected to a plane at isocenter, x_1 and x_2 . $z_{x,T}$ and

$z_{x,B}$ are the vertical distances from the top and bottom of the jaws to the x-ray source, respectively. These values are found when modelling the accelerator. On X jaw 1, which moves out in the negative (N) x direction, the distances between top (T) and bottom (B) inner edges of the jaws and the central axis are defined as XTN and XBN , respectively. Similarly, the X jaw 2, which moves out in the positive (P) x direction, are defined as XTP and XBP . A real set of jaws obviously doesn't change shape, but the way jaws move allow that inner edge to stay aligned with the widest part of the beam, a design that minimizes the beam's penumbra. Digitally, it is much more simple to change the shape of the jaws to correspond to the desired width of the beam. Given this geometry, the rule of similar triangles can be used to solve for all of the values in the djaws file. For example, to solve for XTN :

$$\frac{XTN}{z_{x,T}} = \frac{x_1}{SAD}$$

Applying this concept to all of the values needed for the dynamic jaws input file:

$$\begin{aligned} XTN &= \frac{x_1 * z_{x,T}}{SAD} & YTN &= \frac{y_1 * z_{y,T}}{SAD} \\ XBN &= \frac{x_1 * z_{x,B}}{SAD} & YBN &= \frac{y_1 * z_{y,B}}{SAD} \\ XTP &= \frac{x_2 * z_{x,T}}{SAD} & YTP &= \frac{y_2 * z_{y,T}}{SAD} \\ XBP &= \frac{x_2 * z_{x,B}}{SAD} & YBP &= \frac{y_2 * z_{y,B}}{SAD} \end{aligned}$$

The dynamic jaws input file is formatted in the following lines:

Line 1 - Header

Line 2 - Number of control points

Line 3 - Meterset of control point #1

Line 4 - $z_{y,T}, z_{y,B}, YTN, YBN, YTP, YBP$

Line 5 - $z_{x,T}, z_{x,B}, XTN, XBN, XTP, XBP$

Line 6 - Meterset of control point #2

Line 7 - Identical to Line 4

Line 8 - Identical to Line 5

Lines 9 to end - repeating the meterset at the next sequential control point followed by lines identical to Line 4 and 5.

3.2.4.4 Virtual Phantom

Independently generating a phantom file (file extension *egsphnt*) proved to be much more complicated than the other input files. In fact, the process of accurately and consistently generating virtual phantoms is still being developed by our group.

In the mean time, all phantom files are generated using the MMCTP program. For this process the RCT files are selected in the MMCTP application. A predefined ramp function, which is used to convert the Hounsfield Units (HU) found in the RCT files into densities the EGSnrc code can use, is also selected along with linear accelerator specifications. Once these inputs are complete, MMCTP is able to generate a single text file containing the virtual phantom. The virtual phantom is formatted in the following blocks:

Block 1 - Number of media used (e.g., air, tissue, lung, bone) (one line).

Block 2 - Each medium's name as listed in the PEGS file

Block 3 - Dimensions of the virtual phantom.

Block 4 - Bounds of the voxels in each dimension.

Block 5 - Voxel-by-voxel array indicating each voxel's composition with an integer (e.g., if voxel is listed as "1", it is composed of the first medium listed in Block 2).

Block 6 - Voxel-by-voxel array indicating each voxel's density, in g/cm^3 .

Each medium called by the phantom file is defined in the PEGS file. Contained within the PEGS file are a-priori calculations of cross-section (i.e., particle interaction probability with a medium) data for each medium (40). Thus, at each step in a particle's path, the MC code has a bank of interaction cross sections and their associated probabilities to call. It should be noted that MC simulations can be run "PEGSless" such that the interaction cross-section data would be calculated on the fly. Running a simulation PEGSless may add some additional time.

Voxel size is determined in the TPS and stored in the DICOM RD file. The voxel size is the same in the TPS, the virtual phantom, and the voxel-by-voxel dose file output once an MC simulation has been completed. Typical default voxel dimensions are 0.2 cm x 0.2 cm x 0.25 cm, though the dimensions can be set to other values by the user in the TPS.

3.2.5 Running EGSnrc Jobs on the Cluster

Although the MC simulations can be run locally (i.e., the computer on which the user is working), the process takes a very long time and ties up a considerable amount of computing resources. Instead, it is more practical to submit MC simulations as jobs to the UB CCR super-computing cluster where it can be split up into pieces that are run in parallel. Text files are needed to direct the SLURM to execute the codes. One such file can be submitted to the cluster without any

information other than the commands to run; however, submitting a job in a *job array* provides the SLURM scheduler with more information it can use to tailor the job's parameters. This leads to more efficient and reliable job execution.

Along with the EGSnrc user code input files and job arrays, scripts were written to automate the MC job submission process and do miscellaneous tasks. A majority of these scripts were formatted as SLURM job arrays. These automation scripts performed tasks such as file arrangement; the user does not need to manually move files once they are transferred to the cluster - that is included in the tasks along with submitting the job to the cluster.

Python scripts were written such that, concurrently when the EGSnrc input files are generated, job array text files and automation scripts are generated that will move input files to their designated directories and submit jobs to the SLURM. Once all (EGSnrc input and SLURM job array) files have been generated by the Python scripts and the user has moved them to the cluster, the user need only submit a single job array that will automate the following steps:

1. Make a copy of the linear accelerator information
2. "Build" the accelerator model using the beam.build executable provided in the EGSnrc code
3. Compile the new accelerator model and its libraries
4. Move the djaws and mlc position files to the new accelerator's directory
5. Append the new accelerator's text file with the correct directory paths of the djaws and mlc files.
6. Move the egsinp file to the DOSXYZnrc directory

7. Submit a job array to cluster containing the commands to run the parallelized MC simulation

EGSnrc comes with a built-in parallelizing module. This module did not work properly on its own. Some adaption was done by a UB CCR representative to make the parallelizing module work on the cluster's file system.

Alternatively, in the future it may be possible for the MC input files and job arrays to be produced on the cluster; however, currently the DICOM reading package is not available to the Python interpreter used on the cluster. It is currently more efficient to transfer a few input files and job arrays than it is to transfer the hundreds of anonymized DICOM files required to generate the MC input files.

```
#!/bin/bash
#SBATCH --cluster=MAE
#SBATCH --partition=scavenger
#SBATCH --qos=scavenger
#SBATCH --time=2:00:00
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=16
#SBATCH --job-name=Brain1_arcl_CP3AI27
#SBATCH --output=Brain1_arcl_CP3AI27.out
#SBATCH --mail-user=jdoesmc@yahoo.com
#SBATCH --mail-type=FAIL,END
#SBATCH --requeue

pwd; hostname; date

srun dosxyznrc Brain1 700icru batch=srun p=${SLURM_NPROCS}
```

Figure 3.4: An example of a job array read by the SLURM scheduler. The header contains information that sets the parameters for the job (e.g., job title, maximum time), and the body contains the commands that execute the job.

Job arrays are divided into two parts: the header and the body. An example of a job array is shown in Figure 3.4. The header of the job array contains information that is read by SLURM while the body contains the commands that

execute the job. The SLURM scheduler does not operate on a "first come, first served" basis. Instead, there is an algorithm that attempts to optimize when jobs are run. If the user submits a job without a job array, SLURM assumes it will need the maximum run time, so the scheduler may take longer looking for resources it doesn't actually need. Because the resources needed are defined in the parameters of the job array, submitting jobs in job arrays is much more efficient as the scheduler will only try to locate the resources defined in the job array parameters.

For the work detailed in this dissertation, *scavenging* was used heavily to reduce the overhead time between submitting a job and having it run on the cluster. When submitting a job, the scheduler will place it in a queue until computing resources become available. Depending on how heavily used the cluster's shared resources are being utilized, this overhead time could be very long (hours, even days). To counter this, there is a way of utilizing a partition in the cluster dedicated for faculty use. The resources of this partition may go unused for long periods of time. Because of this, there is an option to scavenge these resources and use them temporarily. The trade-off of scavenging is, although there is a drastic decrease in overhead time when scavenging these resources, should the primary owner of the resources need them at any point during the job's execution, the scavenging user's job will be terminated immediately. Due to the nature of the work presented in this dissertation, computing resources are only needed for relatively short periods of time, so the probability of the jobs being preemptively terminated are low.

3.2.6 DVH Calculation Using Binary Masks

Although DICOM files conveniently contain DVH data for each structure, these data are come from post-processing in the TPS. Information regarding the location of the voxels used for calculating the DVH's is not present. Essentially, there is no easy way of obtaining doses to specific structures given the data provided by the DICOM files. Because of this, a method for obtaining this information was developed.

A *binary mask* is defined as a matrix the same size as the dose matrix and with the same voxel bound coordinates containing only ones and zeros. These dimensions and bounds are obtained from the RD file. Each structure has its own binary mask. A voxel in a binary mask containing a one is composed of the particular structure while a voxel containing a zero does not. Thus, doing an element-by-element multiplication of the dose matrix and a binary mask of a particular structure results in a matrix containing only doses to that structure. Stripping the zeros and sorting the dose values provides the data used to generate a DVH curve. Dose-volume information can be obtained from this and used to evaluate an OF.

Although the binary mask information is not provided in the DICOM files, coordinates of the vertices of structure contours is contained in the RS file. Like the dose matrix, these vertices are stored slice-by-slice such that a 3-D structure is defined as a collection of 2-D polygons. The vertices correspond to the sides of one or more polygons defining the edge of the structure. Examples of these polygonal contours can be seen in Figure 3.5.

A *point-in-polygon algorithm* (PIPA) determines whether a point is located within a polygon defined by a set of vertices, which are extracted from the DI-

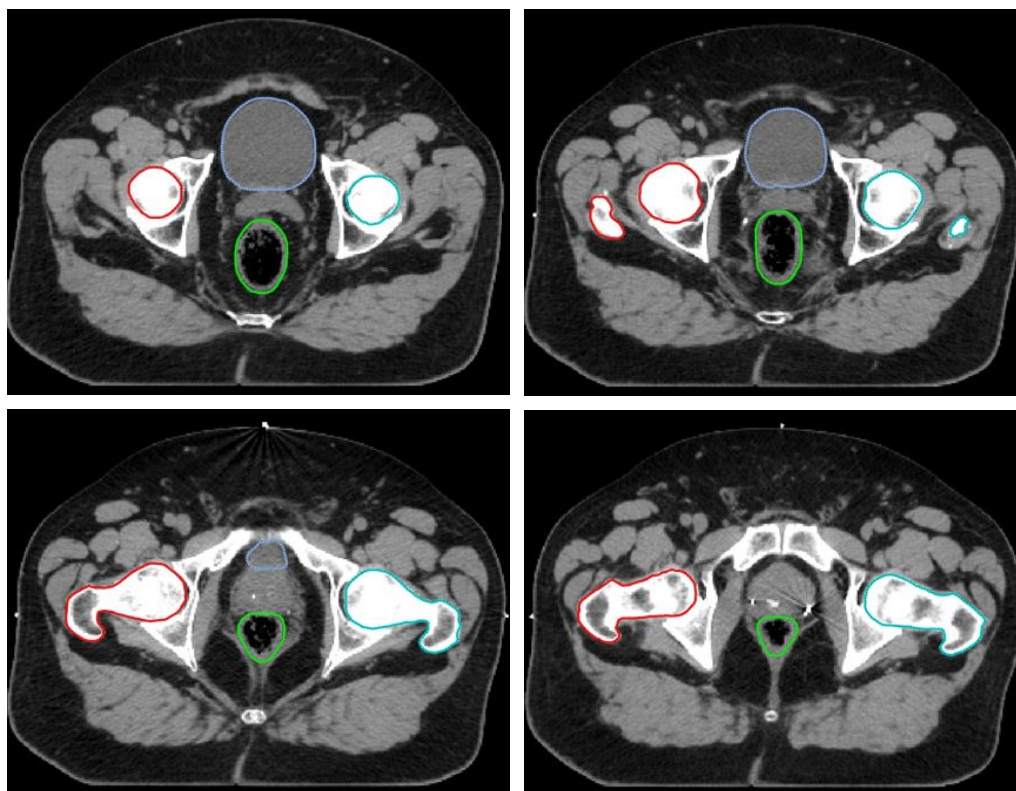


Figure 3.5: 3-D structures are defined as a collection of 2-D contoured polygons. Pictured are several slices from a prostate VMAT case with the bladder (blue), rectum (green), and femoral heads (red and cyan) contours.

COM RS file. Because the voxel bounds are defined by the RD file, the centers of the voxels are easily calculated. Each voxel coordinate along with the polygon of the structure's contour in that slice are fed into the PIPA, which returns either a zero (outside the polygon) or a one (inside the polygon). This 3-D array of ones and zeros is a binary mask. A geospatial PIPA (41) is used to determine the ones and zeros. The PIPA utilizes a ray casting method wherein a ray is cast from the point (i.e., the center of the voxel) in an arbitrary direction along the slice plane. Testing is done along the ray's path to determine whether it crosses a boundary of the polygon. If the number of boundary crossings, n_C , is odd (even), the point is inside (outside) the polygon. If the point is in the polygon,

its corresponding binary mask voxel value is set to one. Otherwise, it is zero. Repeating this process for all voxels for all structures produces a set of binary masks. Refer to Figure 3.6 for a schematic of this process.

An added level of complexity is when there are two or more polygons for a single structure on the same slice. These polygons can be either separate (e.g., both kidneys considered one structure, head and neck PTV's that include the lymph nodes), or they can be overlapping such that the interior of the structure is hollow (e.g., skin, skull). In the RS file, each contour polygon is stored separately and can be identified by slice; thus, a checker is built into the binary mask generator to make sure that, if a single structure has more than one contour on a single slice, all binary masks resulting from those contours are attributed to that slice. For the case of separate, non-overlapping contours, it is simply a matter of summing the mask's matrices together. For the case with overlapping contour polygons, the interior should not be included in the binary mask. To compensate for this, a check is included in the binary mask generator that tests the current value of the target voxel before assigning it a new value. If the value is already one, the value is set to zero as it is located in the interior portion of the structure.

The Computational Environment for Radiotherapy Research (CERR) (42) utilizes a similar ray casting method for obtaining binary masks used for its own DVH calculation. The CERR mask generator method's description was used as a stepping stone for development of our method, and masks produced by CERR were used in a comparison of the masks produced by the methods used in this project.

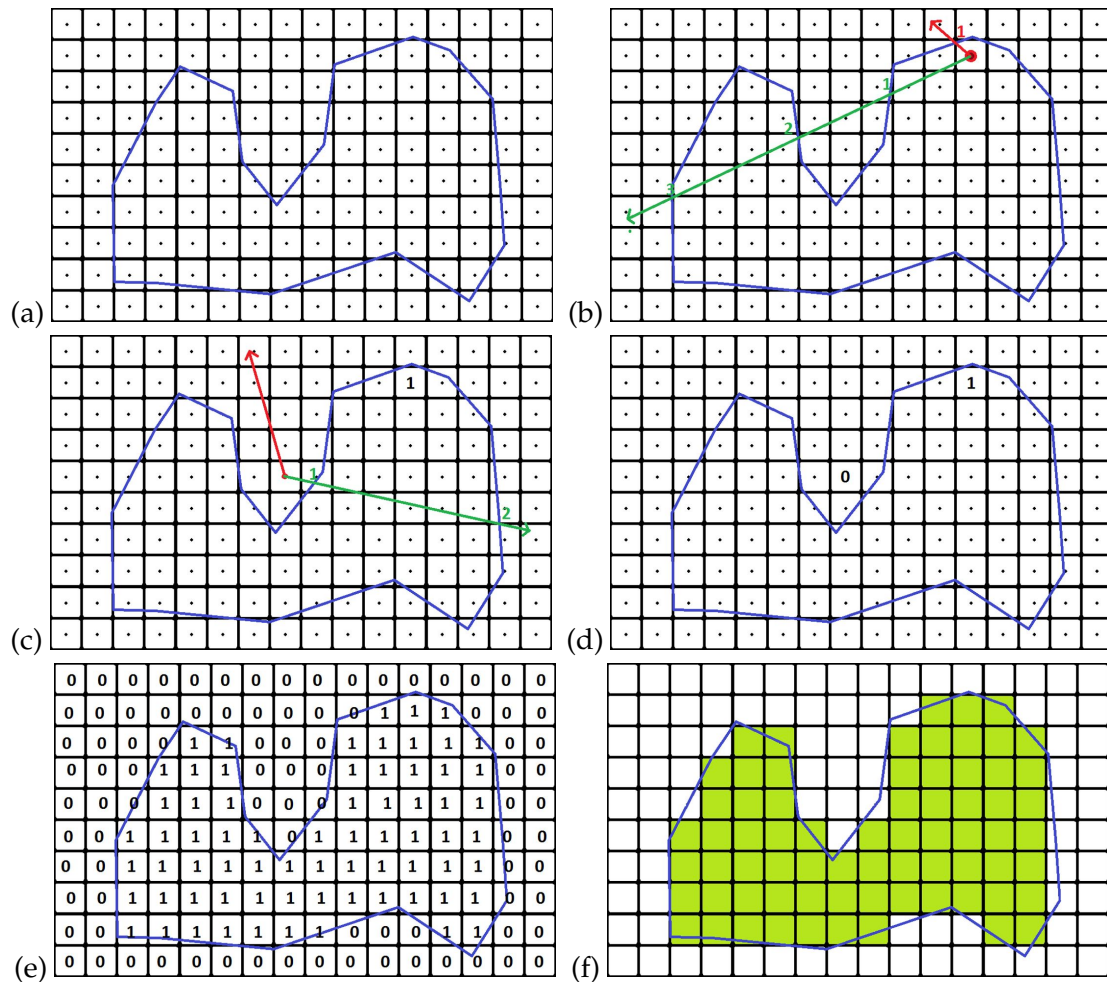


Figure 3.6: A step-by-step schematic of the PIPA process. Each square represents one voxel, and each voxel as a dot in the center denoting the test points used by the PIPA. In (a), a polygon is outlined by the blue contour. One or more rays are cast from that point in an arbitrary direction (b). The red ray intercepts the polygon's boundary once while the green ray intercepts three times; either of these rays meet the condition that indicates the target voxel is within the polygon, so it is set to one (c). Additionally, in (c) the red ray does not intercept the polygon boundary while the green ray intercepts twice; thus, indicating that the voxel is outside the boundary, so it is set to zero. This process is repeated until every voxel is assigned either a zero (outside polygon) or one (inside polygon) (e). Lastly, (f) the voxels containing ones are filled with green coloring to show that there is not an exact fit between the contour and the binary mask; some of the mask is located outside the contour while parts of zero voxels are contained within the contour.

3.3 Results

3.3.1 Anonymized DICOM Files

There were no issues with exported DICOM files. The only metrics available were searching the exported DICOM files for patient identifying information and then testing them for still containing the information required to produce the input files necessary for running EGSnrc MC simulations. Each file was found to contain all of the information necessary to generate the EGSnrc MC code input files as well as the binary masks used for DVH calculations while being successfully scrubbed of all patient identifying information. This indicated that the DICOM files were usable outside of a compliant network without violating any HIPAA controls..

3.3.2 EGSnrc Input Files and SLURM Job Arrays

Input files produced by MMCTP were used to compare the input files generated by the Python scripts written for the work included in this dissertation. This was done by using a text editing program option that directly compares the contents of two files side by side ("vimdiff" in a Unix shell). Once the generation scripts produced input files very similar to the ones produced by MMCTP, simulations were run, and the resulting outputs were used as the starting points for the work detailed in Chapter 4.

3.3.3 Binary Masks

Three indirect comparisons were performed (see Discussion for the reasoning behind these choices). For a quantitative comparison, the volumes of structures

reported by the TPS, CERR, and the binary mask generator were compared. For a qualitative evaluation, the binary masks were overlaid on CT slices of the anatomy with the contours to evaluate accuracy in positioning. Another qualitative evaluation involved comparing DVH curves extracted directly from the RD file and those produced using the binary masks and plan's dose matrix.

3.3.3.1 Comparison of calculated volumes

Eclipse directly reports volume measurements for each structure, and calculating the volumes for each structure with a binary mask was simply a matter of multiplying the voxel dimensions by the number of "one" voxels in the mask. Volume comparisons for structures taken from three brain, two head and neck, and two prostate VMAT cases were performed. See Figure 3.7 for a graphical representation of these results. The largest differences occurred for the very small structures (i.e volumes on the order of $<5.7 \text{ cm}^3$). Structures with binary mask generator calculated volumes of $>5.7 \text{ cm}^3$ were found to have less than 10% volume differences, with a steady decrease in the differences as the structure volumes increased. The maximum difference was 70.7%, which was an optic chiasm (Eclipse volume: 0.404 cm^3) belonging to a brain VMAT case. In every case, the binary mask calculated volume was greater than the Eclipse reported volume. The possible reason for this will be explored in the Discussion.

This analysis was also performed between the masks generated for the work detailed in this dissertation and the masks generated by CERR, as CERR's method was the basis for ours. Refer to Figure 3.8 for a graphical representation of these results. Although the trend of the volume differences was similar to the differences between the Eclipse reported volumes and the binary mask generator calculated volumes (i.e., smaller volumes having greater differences), the mag-

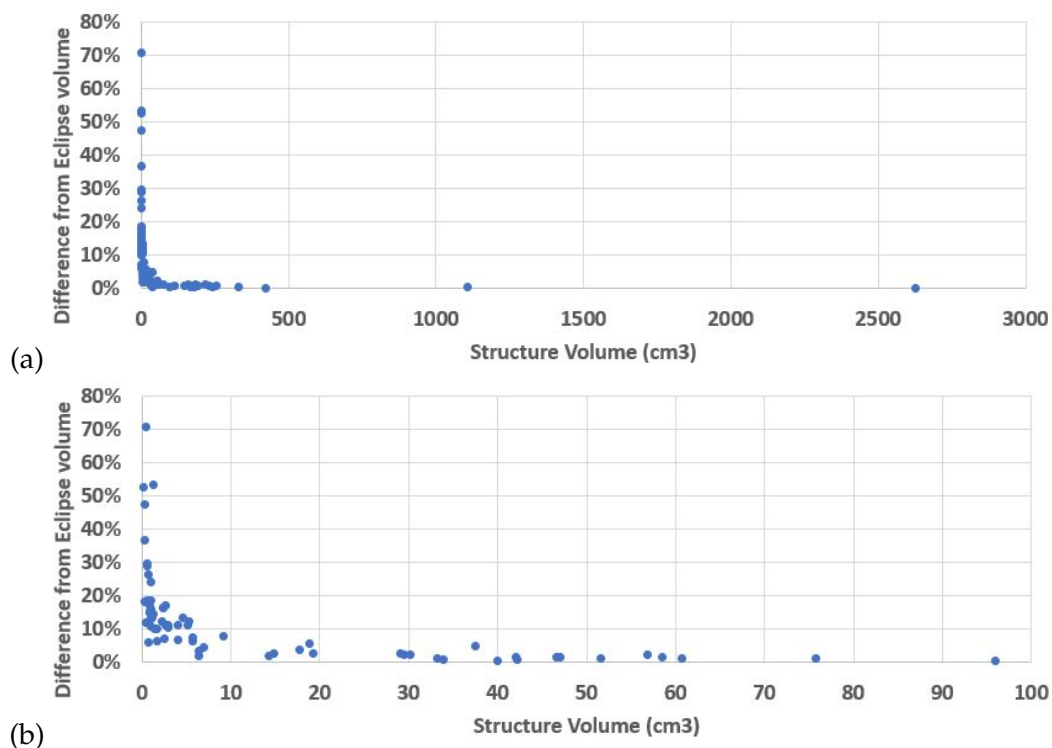


Figure 3.7: (a) Relative differences in structure volumes reported by Eclipse versus calculated using binary masks. (b) Zoomed in to show greater detail in differences of the very small volumes.

nitudes were reduced considerably. The maximum difference was 10.6%, which was a pituitary gland (Eclipse volume: 0.351 cm^3) belonging to another brain VMAT case.

3.3.3.2 Visual Comparison of Masks and Contours

A qualitative evaluation method of the binary mask generator was to overlay masks with their corresponding contours. Figure 3.9 shows the PTV contour (blue) and resulting binary mask (red) for a brain VMAT case. The binary mask generator was able to identify voxels within the polygon defined by the contour on this slice (along with all other slices, not shown). This was the most simple example of a contour/mask - only one polygon was defined for that structure

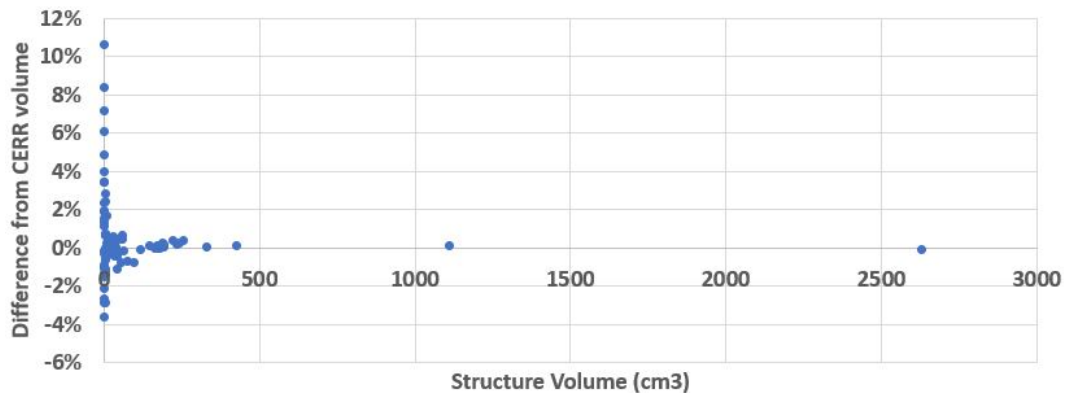


Figure 3.8: Relative differences in structure volumes reported by CERR versus the volumes calculated using binary masks.

on that slice.

The more complex contour/mask examples are shown in Figure 3.10. In Figure 3.10(a), the same structure consisted of two separate polygons on the same slice. In Figure 3.10(b), the structure is ring shaped; there was an exterior contour defining the outer edge of the structure and an interior contour defining the inner boundary, which did not contain the structure. The binary mask generator was able to produce masks as intended for both of these cases.

The final example of a contour overlaid with its corresponding binary mask can be seen in Figure 3.11. This shows the contour and mask of an optic chiasm - a very small structure ($<0.4 \text{ cm}^3$). This is the same structure mentioned earlier with the largest relative difference in calculated volume between the binary mask and Eclipse ($>70\%$). Seen in the figure are four slices of the optic chiasm contour (yellow line) and its associated binary mask (red). There was considerable coincidence between the contour and the mask, though, the mask appeared to be larger than the contour itself as it included voxels located outside the contour.

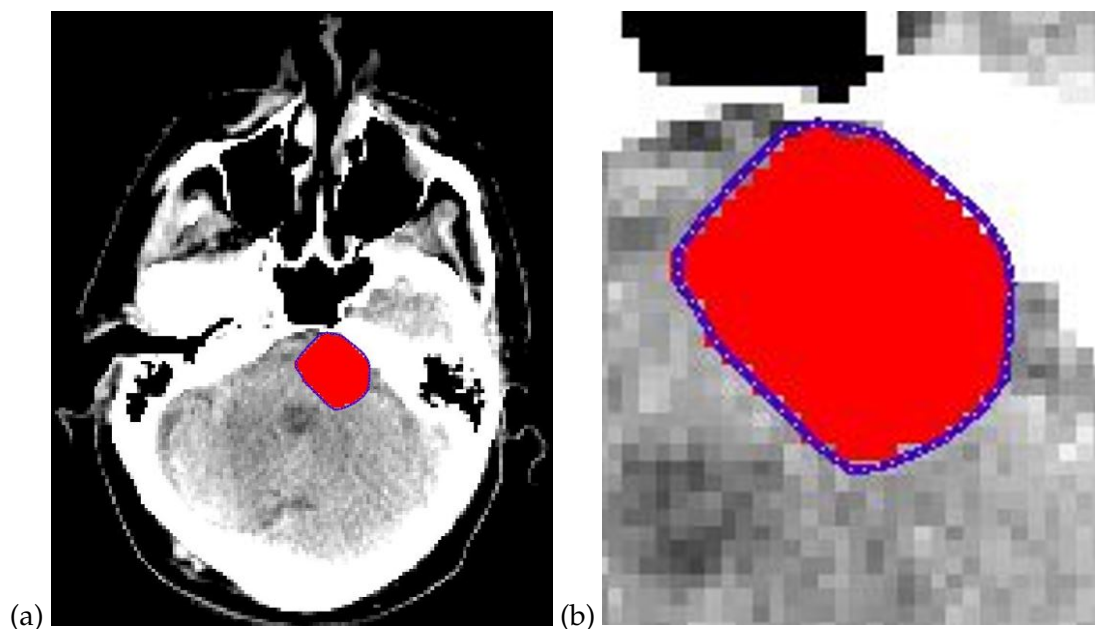


Figure 3.9: A CT slice of a brain VMAT patient containing a contour (blue line) of a PTV with the associated binary mask overlaid (red). The main difference between the contour and mask is apparent at the periphery. For this particular slice, the mask appears to be slightly smaller than the contour.

3.3.3.3 Comparison of DVH's

Another qualitative analysis of the performance of the binary mask generator was to compare the DVH's provided directly from Eclipse to the ones produced by applying binary masks to the dose array from Eclipse.

Two sets of structures are displayed in Figure 3.12, a brain VMAT case, and Figure 3.13, a prostate VMAT case. The former was composed of mostly very small structures ranging from 0.34 to 56.9 cm³ while the latter was composed of relatively larger structures ranging from 160.1 to 254.3 cm³. For the most part, all of the DVH curves produced with the binary masks tightly conformed to the curves from Eclipse, though there is a small deviation between the curves belonging to the right lens in the brain plan.

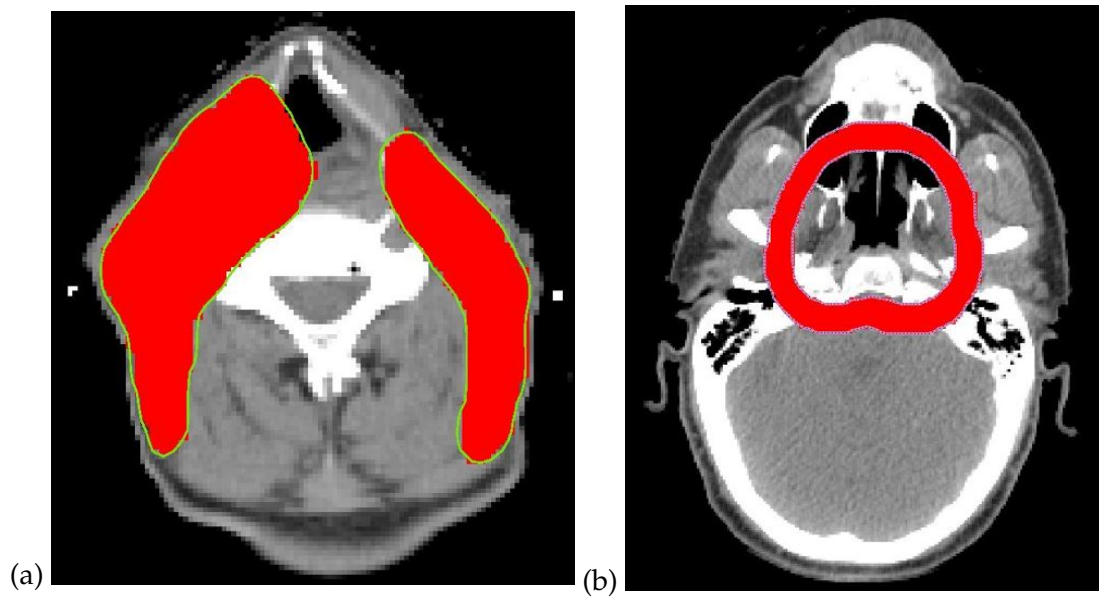


Figure 3.10: (a) A CT slice of a head and neck VMAT plan containing a two non-overlapping contours (yellow line) of a PTV with the associated binary mask overlaid (red). (b) A CT slice of a separate head and neck VMAT plan containing a hollow structure defined by two overlapping contours (pink line) with the binary mask is overlaid (red).

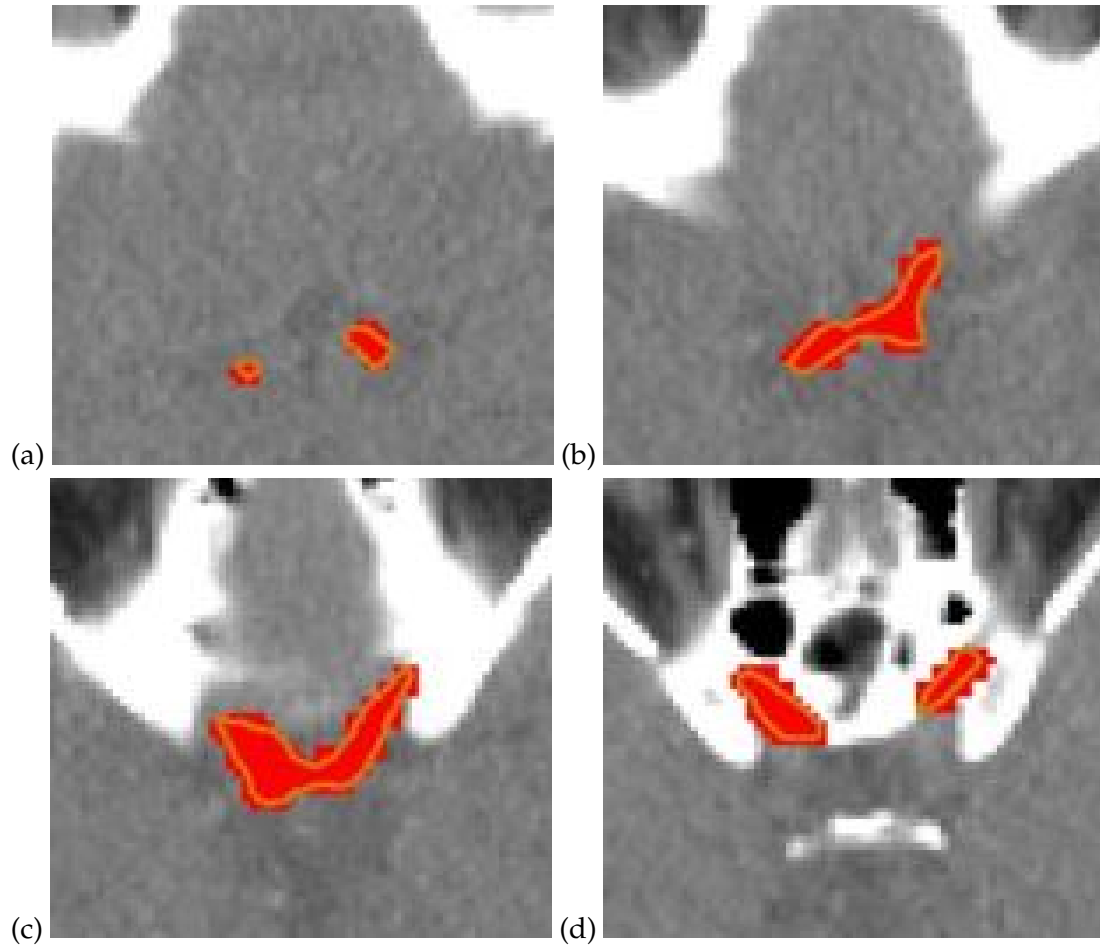


Figure 3.11: Four adjacent CT slices of a brain VMAT plan containing the contours (yellow lines) of an optic chiasm, a very small structure (0.4cm^3) with the binary mask overlaid (red)

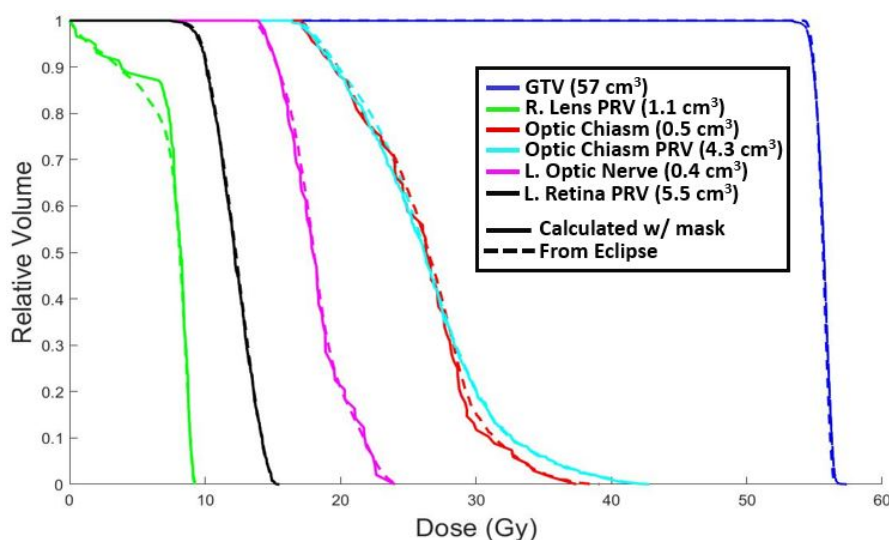


Figure 3.12: DVH with structures of a brain VMAT plan for the same Eclipse dose matrix. Particularly, the smallest structures defined for the plan (and the GTV) are included. The dashed curves are data directly from Eclipse while the solid curves were calculated using binary masks and the dose array extracted from the RD file..

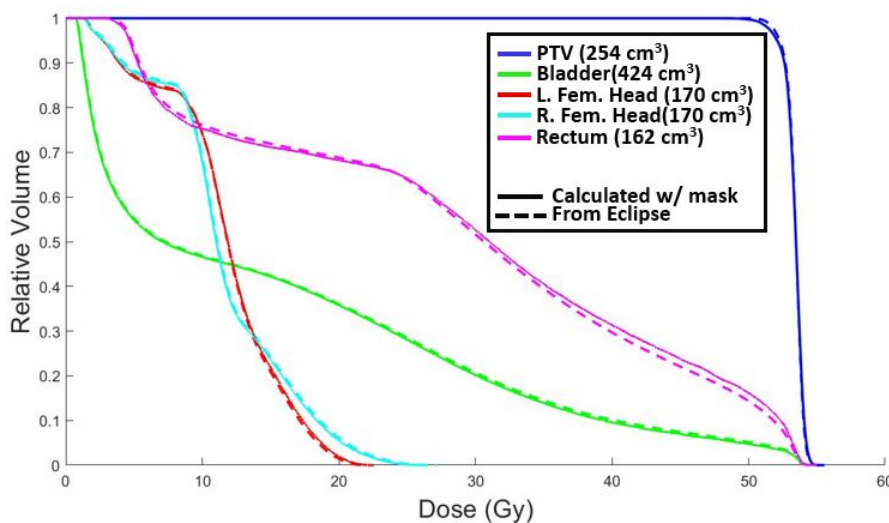


Figure 3.13: DVH with structures of a prostate VMAT plan for the same Eclipse dose matrix. These structures are relatively large when compared to structures in brain and head and neck plans. The dashed curves are data directly from Eclipse while the solid curves were calculated using binary masks and the dose array extracted from the RD file.

3.4 Discussion

3.4.1 EGSnrc Input Files and SLURM Job Arrays

One of the main motivations for this work was to automate the generation of input files to be used for running Monte Carlo simulations. To the inexperienced MC user, the organization of these input files is not intuitive; there are lines of number and data with virtually no labelling. In order to understand what each line of code represents, one must scour the user manuals, online message boards and forums, and hopefully be fortunate enough to know an experienced user who can render assistance. Monte Carlo is an extremely useful tool to have for data verification, but it can be unwieldy and frustrating to use.

Using a text editor feature, the input files generated by the Python scripts were found to be virtually identical to those generated by the MMCTP application. This provided an apples-to-apples metric for evaluating the effectiveness of the input file generators. An additional indirect method could be used to compare the simulation results produced as a result of the input files; however, this was not done as the simulation results would be confirmed during the work done in Chapter 4. Additionally, multiple users have used the input file generators developed here for their projects, and they have all reported successful results.

It should be noted that the input file generators created and detailed in this dissertation are currently somewhat limited in their scope of use. The beam modelling is only applicable to the Varian Trilogy linear accelerators currently at Roswell Park Comprehensive Cancer Center as of April 2020. Not included are the TrueBeam accelerators at that site, or any other linear accelerators at any

other clinics. Work is currently being done to expand this scope. In the same vein that this work simplified DOSXYZnrc simulation input file generation and submission, that work is looking to make accelerator modelling easier for the user. Since that work is still in progress, it will not be detailed in this dissertation. However, should a user at another clinic have their desired accelerator pre-modelled for BEAMnrc, adaptation of the input file generators developed in this work would be relatively simple.

3.4.2 Evaluating the Accuracy of Binary Mask Generation

Unfortunately, there is no direct way of evaluating the accuracy of the binary masks generated for the work detailed by in this dissertation. Ideally, we would be able to do a direct comparison with masks generated by the TPS, Eclipse, using some kind of test such as obtaining a Dice coefficient. Since Eclipse does not provide binary masks in the DICOM files and is a black box application, this is not possible. This is the reasoning for choosing the comparisons used in this dissertation.

When comparing the volumes reported by Eclipse to those calculated using the binary masks it was found that the binary mask volumes are greater in every case. This is likely because, during volume calculation, Eclipse is somehow able to compensate for the surface voxels only being partially included in the volume. It is apparent in the schematic in Figure 3.6 as well as Figure 3.9, both of which contain a contour overlaid with binary mask, that the surface voxels can include volumes located outside of the contour. Eclipse likely has a way of accounting for this, possibly with some kind of interpolation. Additionally, this voxel partial volume approximation is applicable along the slice axis, thus,

applying even more possibility in reduction of surface voxels' contribution to the volume of the structure.

The binary masks being the same dimensions as the dose array led to being able to display them like dose matrices. This made comparing the distribution of the binary masks and the shapes of the contours convenient and easy. Qualitatively, there appears to be tight overlap between the contours and masks, which indicates that the positioning of the masks by the generator is correct. The masks tend to be visually slightly larger than the contours. This confirms that the binary mask method overestimates the size of each structure when compared to Eclipse estimates.

Lastly, comparing the DVH's provided by Eclipse and those calculated using the binary masks is probably the most telling of the evaluations. The goal of generating binary masks was to produce dose-volume data to be used to evaluate an optimization function, so it was imperative that the data be accurate. Qualitatively, the easiest way to evaluate this is by comparing DVH curves produced by applying the binary masks to some standard. Here, we used the DVH curves from Eclipse since they are used clinically. The DVH's produced with the binary masks all conform closely to their Eclipse-generated counterparts, even for the structure with volumes that differed by $>50\%$ or more. This indicates that, even for structures with large relative differences between the reported Eclipse volumes and those calculated with the binary masks, the mask generator produced for this work is accurate. As such, it was used to generate dose-volume data used during the Enhanced Optimization process as detailed in Chapter 4.

Chapter 4

Enhanced Optimization of VMAT Plans Using Monte Carlo Generated Beamlets

Put that cookie down, NOW.

Howard Langston - Jingle All the
Way (1996)

4.1 Motivation

As mentioned in Chapter 2, treatment planning systems tend to use pencil-beam dose calculation algorithms while performing plan optimization. Although it is difficult to evaluate directly, the pencil-beam algorithm used by Eclipse during the optimization process will be much less accurate than the Monte Carlo (MC) or Analytical Anisotropic Algorithm (AAA) methods. Because of this, the optimization solution landscape may not be represented accurately.

A well-known trade-off with the high accuracy of MC methods is their large computational costs. They require extensive working memory and long execution times. Therefore, a complete optimization process using MC is impractical. However, we may use MC calculations to provide *perturbations* to a VMAT plan after it has been optimized by the TPS. These perturbations involve beamlet dose calculations, which may be performed by an MC method in a reasonable time frame (see Methods).

By applying these perturbations, we aim to generate VMAT plans that more closely conform to their optimization objectives. These objectives are set with the intention of sparing healthy tissues from receiving harmful amounts of radiation, which can lead to tissue damage or secondary cancers. By conforming more closely to these objectives, the risk of harming the patient while administering the radiation treatment is reduced.

4.2 Materials and Methods

4.2.1 Exporting VMAT Plan Files

In typical clinical workflow the VMAT plan is created and optimized by the planner and then approved by the physician. The EO can be performed before or after physician approval, and the resulting plan can be compared with the original for final approval. Figure 4.1 illustrates this workflow. The EO process begins with a VMAT plan which has been optimized by the TPS. In addition to the CT data, three DICOM files are exported: the dose file containing the dose matrix and corresponding DVH's, the plan file containing the control point information, and the structures file containing the contoured structure data.

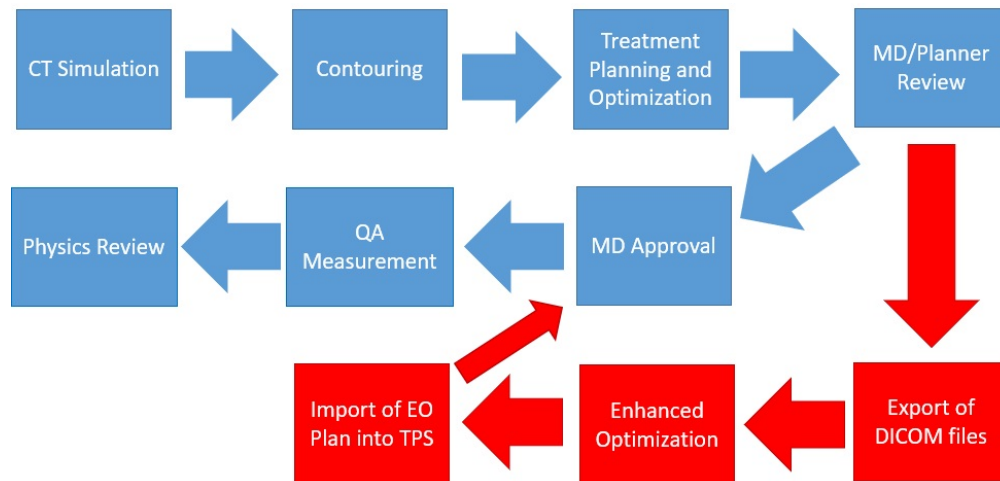


Figure 4.1: A flow chart of the clinical treatment planning process. The conventional workflow is indicated in blue, while the proposed EO addition is indicated in red.

4.2.2 Beamlet Calculation Using Monte Carlo

The Electron Gamma Shower, National Research Council (EGSnrc) toolkit is used for all MC dose calculations performed outside of the TPS. The BEAM-nrc module is used to model the linear accelerator, and the DOSXYZnrc module is used to create the patient/phantom model.

Perturbations to the original matrix are created by simulating *beamlets*. A beamlet is a change in dose resulting from moving a given MLC leaf into or out of the field by a small step, for a given control point. In this study, the step size used was 0.5 cm, although this value may be modified (see Discussion). Figure 4.2 shows the beam's eye view of a beamlet for a prostate plan, while Figure 4.3 shows a transverse view of this beamlet. Therefore, each beamlet corresponds to a small dose matrix, which can be calculated by simulating the modified VMAT plan (original plan, except with a single leaf moved in or out, in a single control point), and subtracting the original dose matrix. In practice, this may be accomplished using partial arcs, each comprising three adjacent

control points. Examples of the dose contributions of from running a partial arc can be seen in Figure 4.4. Any set of MLC perturbations can then be modeled by considering the original dose matrix and then adding or subtracting the corresponding beamlet matrices. This process is illustrated in Figure 4.5.

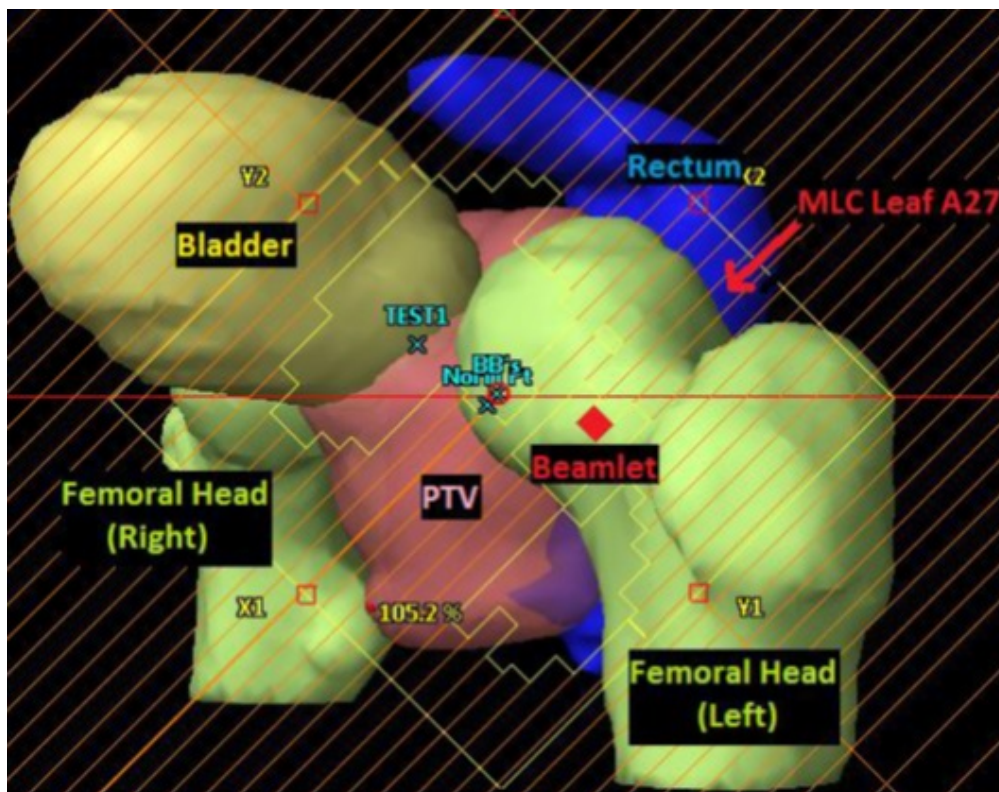


Figure 4.2: The beam's eye view of contoured patient anatomy for a prostate VMAT plan at a point along the arc (i.e., a control point). The red square represents the cross section of a beamlet generated by moving a single leaf indicated in the figure out by 0.5 cm.

4.2.2.1 Processing DOSXYZnrc Output - the 3ddose File

When a DOSXYZnrc simulation finishes successfully, a text file containing the dose matrix information is created. This file is known as the 3ddose file. 3ddose

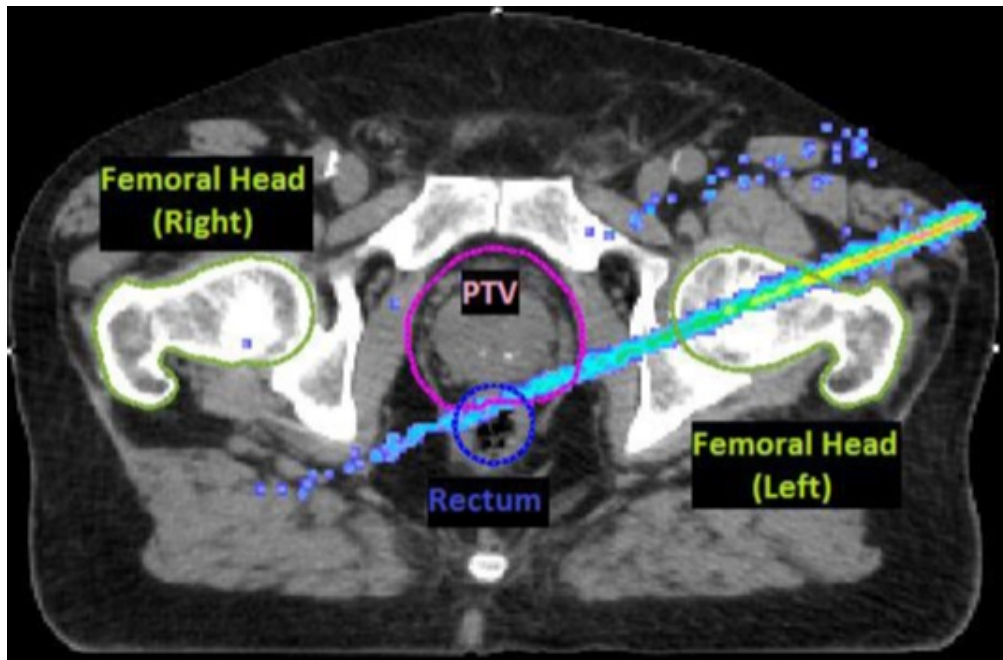


Figure 4.3: An axial cross section of the same anatomy shown in Figure 4.2 with the beamlet dose displayed as a relative color wash.

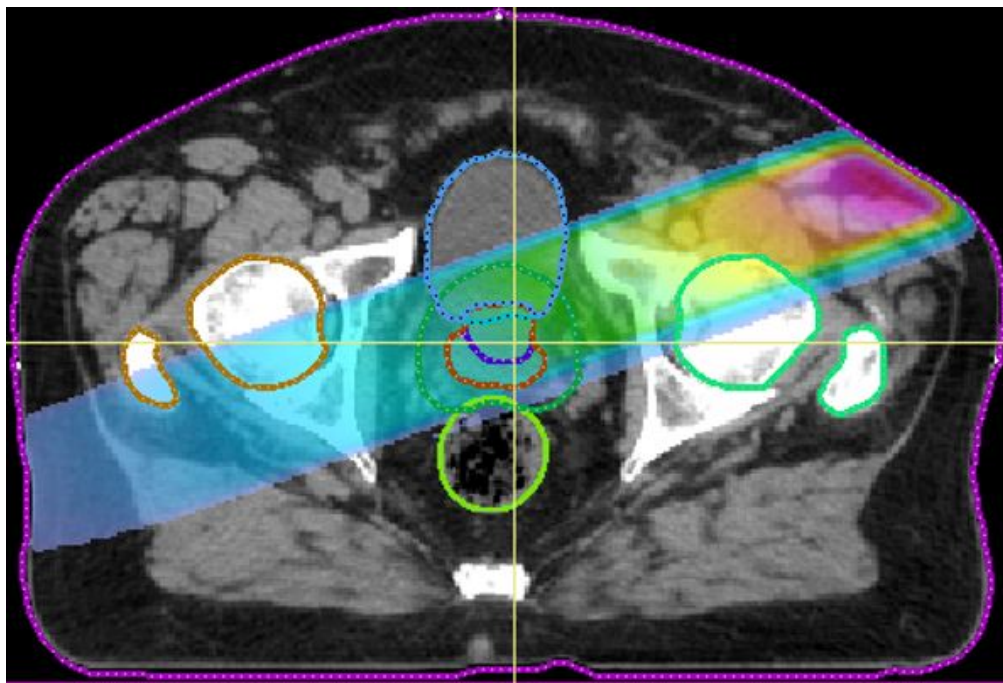


Figure 4.4: A CT slice of a VMAT prostate case with a color wash (blue-low dose to pink-high dose) of the dose contribution of a partial arc.

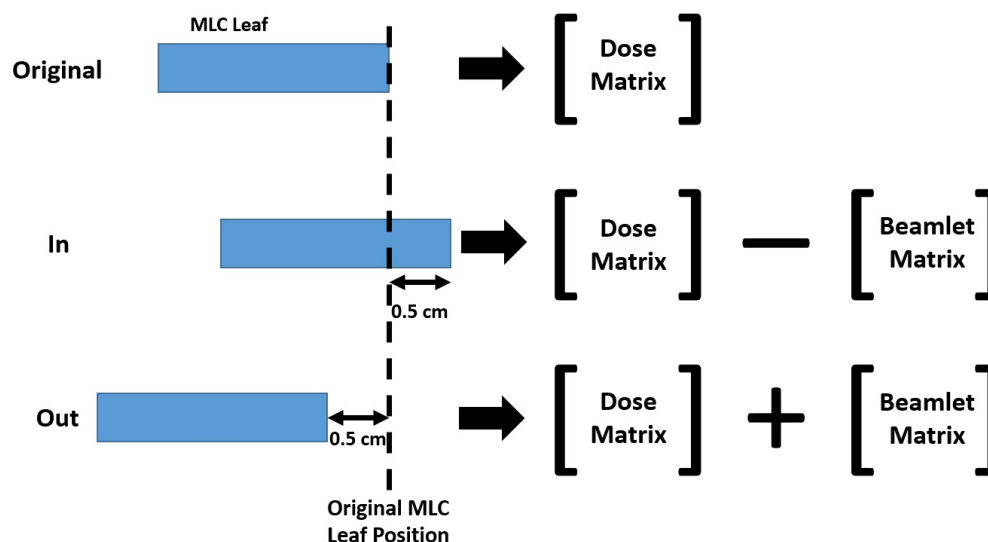


Figure 4.5: Perturbations to the TPS plan are performed by adding/subtracting a beamlet dose matrix to/from the current dose matrix, depending on whether the MLC leaf position is moved out of or into the field.

files are in the following format:

- Line 1 - Dimensions of the dose array
- Line 2 - Voxel bounds - x
- Line 3 - Voxel bounds - y
- Line 4 - Voxel bounds - z
- Line 5 - Voxel-by-voxel doses per arbitrary output
- Line 6 - Voxel-by-voxel relative uncertainties

The 3ddose file does not contain absolute doses. Instead, it reports doses per some arbitrary linac output, $d(v)$. Converting $d(v)$ to absolute dose, $D(v)$, is done using the following equation:

$$D(v) = d(v) * k * MU_{fx} * n_{fx} \quad (4.1)$$

where v is the voxel, k is the conversion factor found in Bhagroo's work (units:

arbitrary output per monitor unit) (6), MU_{fx} is the number of monitor units per treatment fraction, and n_{fx} is the number of fractions in the treatment.

k is linac dependent; however, finding it is simply a matter of running a MC simulation modelled after a standard setup. A simulated water block located at a source to surface distance (SSD) of 100 cm. The gantry and collimator angles are set to 0° , the field size is set to $10 \times 10 \text{ cm}^2$, and 100 MU's are delivered. k is the ratio of the output at standard depth according to the MC simulation and the expected dose, which is usually 100 cGy (as per TG-51 calibrations).

4.2.2.2 Exclusion of Unnecessary Beamlet Calculations

The input file scripts developed in the work described in Chapter 3 are used to generate the beamlet input files, but some additions were to be made to accommodate the plan parameter modifications needed to obtain beamlet dose perturbation information.

It is time consuming and ultimately unnecessary to calculate every possible beamlet. The following is a list of criteria used for excluding MLC leafs from having beamlets calculated unnecessarily for them:

1. Leafs covered by the jaws
2. Leafs that, if extended, would collide with the corresponding leaf on the opposite bank
3. Leafs that would be unable to extend or retract fast enough between control points

Python scripts were written to analyze the leaf parameters in the above list to determine if exclusion was necessary. Excluding leafs covered by the jaws was

simply a matter of comparing the Y jaws positions to the leaf bounds, both of which are stored in the RP file. If the Y jaws were over both bounds of the leaf, that leaf was excluded. A pair of leafs located opposite each other can collide if they are extended too far. To prevent these collisions from happening, the distance between the two leafs was calculated using the leaf position information extracted from the RP file. If that distance was less than the beamlet step distance, both of the leafs were excluded. Lastly, the maximum MLC leaf speed was obtained from the linear accelerator specifications. Taking into account the gantry speed, the current leaf speed between control points can be calculated for each leaf. If applying a change to the leaf's position leads to the leaf needing to exceed its maximum speed to achieve the new modified position, that leaf is excluded.

An optional exclusion parameter takes into account whether or not the center of a leaf's tip is located inside the two-dimensional projection of the PTV in the beam's eye view. Refer to Figure 4.6 for an example of what this looks like. Mathematically, this is determined with a two-dimensional plane located at isocenter and orthogonal to the beam's direction. Using geometry, a three-dimensional point, located on the aforementioned two-dimensional plane, can be found with the following equations:

$$x = (x_o - d_{leaf}) \cos \theta \cos \alpha + L_{leaf} \cos \theta \cos \alpha \quad (4.2)$$

$$y = (y_o - d_{leaf}) \sin \theta \sin \alpha + L_{leaf} \sin \theta \cos \alpha \quad (4.3)$$

$$z = (z_o - d_{leaf}) \cos \alpha + L_{leaf} \sin \alpha \quad (4.4)$$

where $x_o, y_o,$ and z_o are the user-defined offsets of the isocenter, θ and α are the gantry and collimator angles, respectively, d_{leaf} isocenter-projected distance

from the beginning of the MLC bank to the center of the leaf, and L_{leaf} is the isocenter-projected distance of the leaf from the central axis. Once the three-dimensional coordinates, x, y , and z , are found, a Delaunay triangulation algorithm is used to determine whether or not those coordinates are inside a convex hull defined by the vertices of the PTV contours (43). If the point is not found to be within the convex hull, the center of the leaf is outside the beam's eye view of PTV, and the leaf is excluded from beamlet calculations.



Figure 4.6: Beam's eye view, including MLC leaves, of a PTV from a VMAT brain case. The leaves indicated overlap the PTV's projection in this beam's eye view. Other leaves would be excluded from beamlet calculations.

The MLC leaves remaining after the exclusion scripts were executed are recorded

in a dictionary-type data structure. These data are exported to a text file (file extension jBLinfo) for use in beamlet post-processing.

4.2.2.3 Generating Input Files and Automation Scripts for Beamlet Calculations

Each beamlet required its own set of input files and automation scripts. Additions to the file generation scripts were made to accommodate this. Thousands of files were necessary for each plan. A multiple tier scheme was devised in order to more efficiently automate the beamlet job submission process. Beamlet submissions were set to 500 jobs per batch. 500 was selected due to the limited storage capacity in our group's project space. Although this selection does not affect the precision or accuracy of the calculations, it does affect the time it takes to complete the beamlet calculation process. Being able to run all beamlets simultaneously instead of in batches is preferable; however, it is not possible given the current hardware resources available.

Each beamlet required its own virtual accelerator, so the input file generator scripts were written to keep track of this. Beamlets needed their own unique MLC position file, so those were accounted for as well, and automation scripts placed them in the correct accelerator when it was time for the beamlet to be calculated. Unique eginp files were generated for each beamlet as well; these contained the specific locations of the accelerator folders as well as the designations of the beamlets themselves so they could be accounted for in post-processing.

4.2.2.4 Choosing the Number of Histories for the Full Arcs and Beamlets

For a simple and relatively fast secondary dose check, as described in Bhagroo's work (6), 10^7 histories for each arc in VMAT plans was found to be sufficient for point dose evaluations. Clinically, tolerance for a secondary check is 5%, and Bhagroo's method for secondary dose check was found to be well within that tolerance. For the work presented in this chapter, though, these same small variances (3-5%) in dose distributions could potentially lead to relatively large misrepresentations of the objective function solution space. Because of this, some investigation into improving the accuracy of the MC data was performed.

There is an inherent trade off when increasing the number of histories simulated. Increasing the number of histories decreases the uncertainty in the data, but it also increases the amount of time the simulation takes to run. There is a diminishing return due to the inverse-square-root dependence of uncertainty on N . The bottom line is simply running as many histories as possible is inefficient. See Table 4.1 for a summary of the computing resources used, time to run, and mean uncertainty for a single brain VMAT plan arc simulated using DOSXYZnrc. Additionally, see Figure 4.7 for a log-linear plot of the number of histories related to the mean uncertainties. A bar graph showing the relative uncertainties corresponding to cumulative percentages of voxels for MC simulations run with different numbers of histories can be seen in Figure 4.8. The figure shows that increasing the number of histories reduces the relative uncertainty, which is expected. It should be noted that the 10^{10} histories simulation required a different method in order to be completed. The code that allowed the other simulations to be parallelized would not work, so parallelization had to be done manually. Since that method is uncommon, it will not be detailed

Table 4.1: A summary of the computing resources used and time required to run a single VMAT arc using DOSXYZnrc on the UB CCR cluster. Additionally, the mean uncertainty reported by each simulation is included along with the standard deviation.

Histories	Cores	Time (hr)	Mean Uncertainty
10^6	48	0.08	$20.87 \pm 14.89\%$
10^7	128	0.175	$12.26 \pm 9.3\%$
10^8	160	0.8	$3.46 \pm 2.61\%$
10^9	160	2.8333	$1.24 \pm 0.94\%$
10^{10}	1000	12.25	$0.39 \pm 0.3\%$

here. Figure 4.9 shows dose color washes for a single slice of a brain VMAT patient. In the figure, it that the data resulting from 10^6 histories is noisy and incomplete, which is indicated by the "graininess" of the color wash. As the number of histories increases, the graininess is reduced and data smooths out. Using 10^{10} histories as a reference, as it was the most histories we could run for a single simulation, differences in dose distributions were calculated. These can be seen in Figure 4.10, which further illustrates the amount of noise contribution to the dose distributions for each simulation. In short, using fewer histories leads to more noise in the dose data.

The noise in the data matters for optimization. As we are applying small perturbations to the plan during EO, the subtle voxel-by-voxel differences in doses introduced by noise in the data can substantially impact the optimization by potentially changing the solution landscape, which would lead the EO to find a different solution.

It was determined that 10^9 histories for the full arc simulations was a balanced compromise between simulation run time and data uncertainty. This size of simulation can be run in parallel with the beamlet histories, does not take long, and results in uncertainties of about 1.3%.

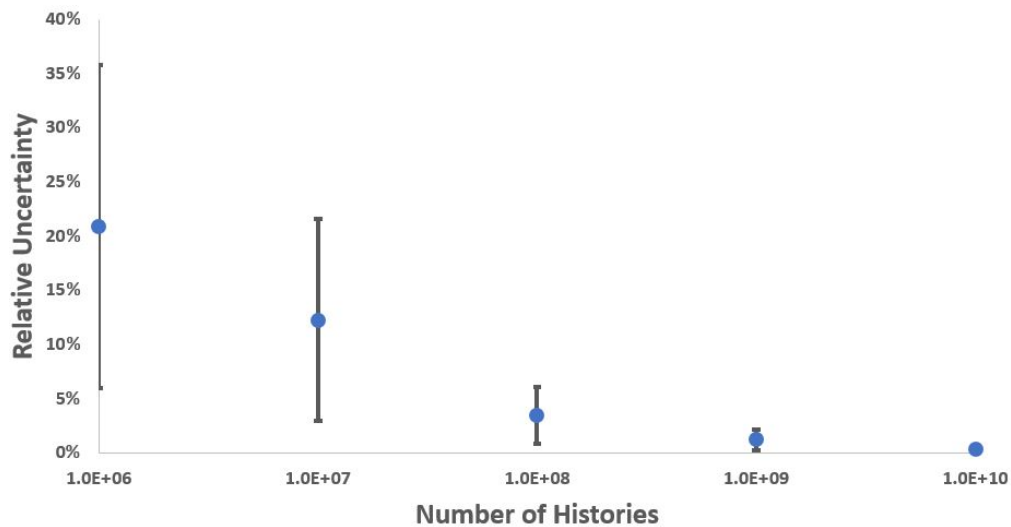


Figure 4.7: A log-linear plot of the number of histories in a single simulation related to the average relative uncertainties. The error bars represent the standard deviations of the average uncertainties.

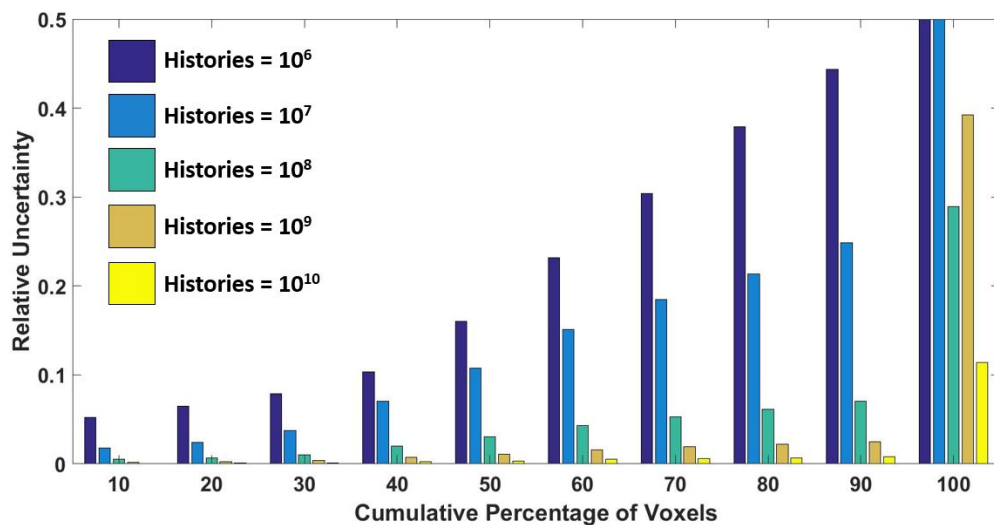


Figure 4.8: The relative uncertainties corresponding to cumulative percentages of voxels for MC simulations run with 10^6 - 10^{10} histories. Note: EGSnrc has an uncertainties cutoff for those greater than 0.5, voxels associated with that high of an uncertainty are set to zero.

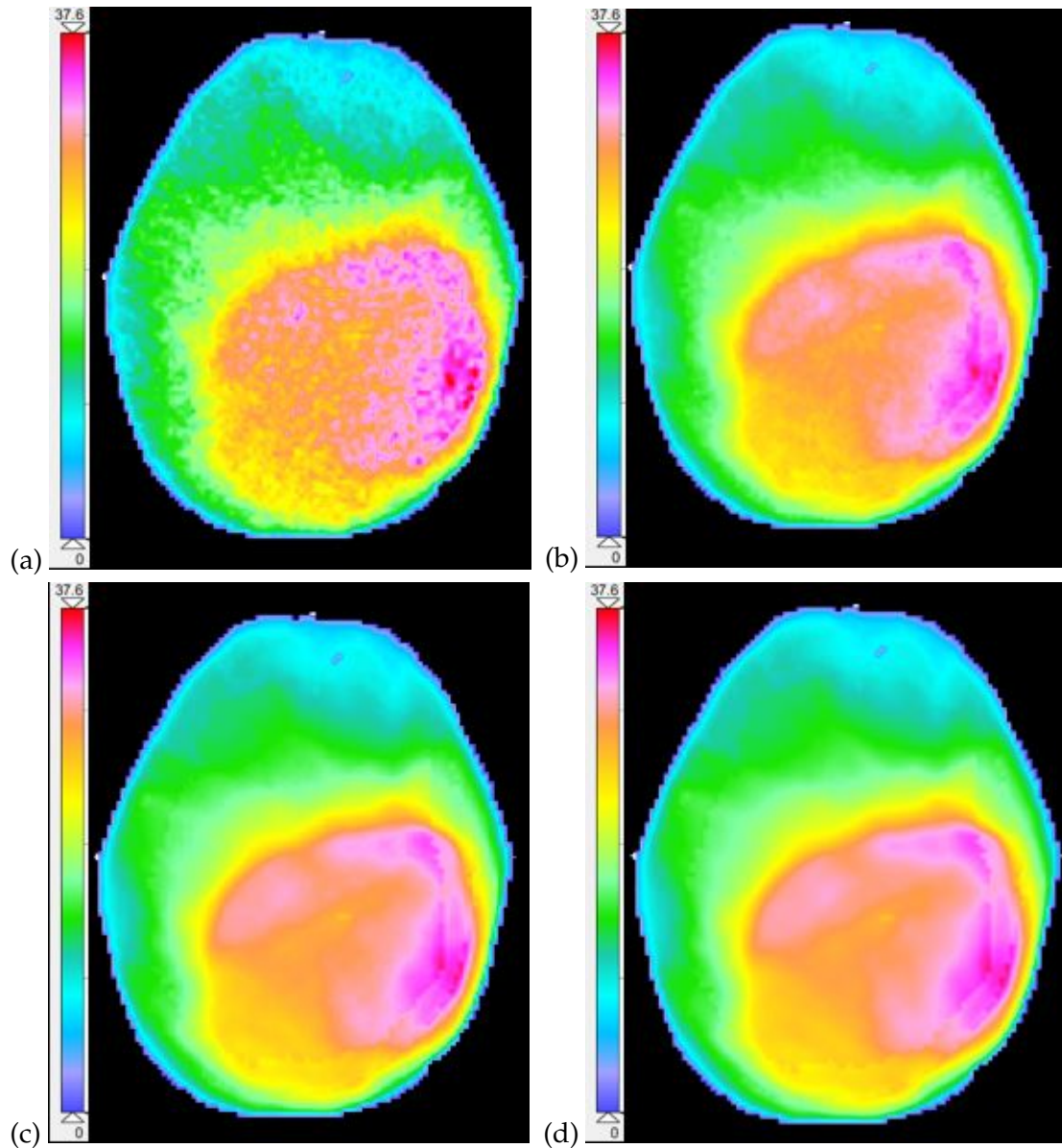


Figure 4.9: Relative dose color washes for a VMAT brain plan using (a) 10^6 , (b) 10^7 , (c) 10^8 , and (d) 10^9 histories. Noise is visible as "graininess" in each figure, where (a) has a considerable amount of graininess and (c) and (d) very grainy. Differences between these dose distributions and a reference dose distribution can be seen in Figure 4.10. The dose ranges from 0 to 37.6 Gy.

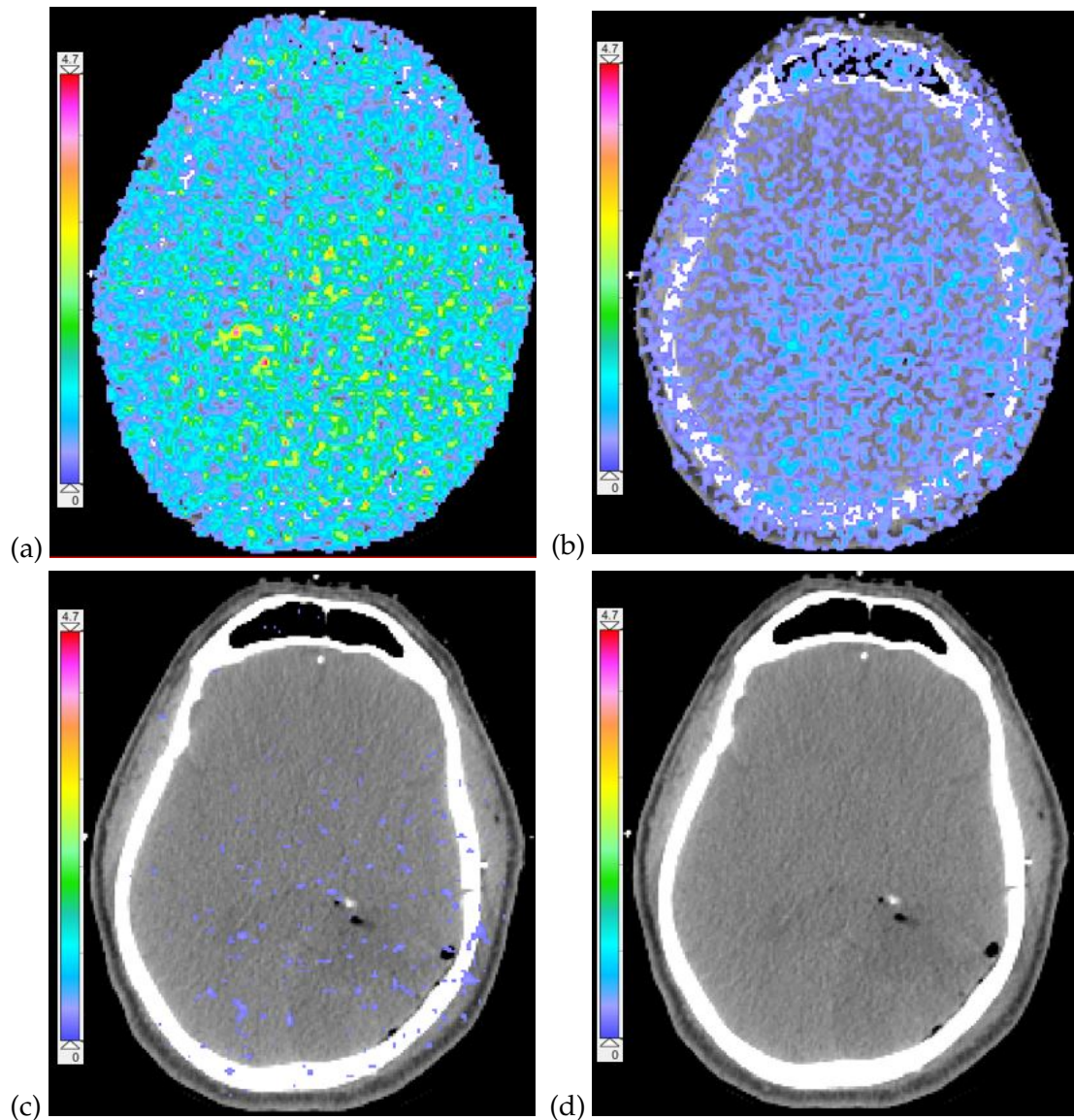


Figure 4.10: The noise contribution to MC dose calculations, which are shown as differences in dose, presented as color washes, for a VMAT brain plan using 10^{10} histories as a reference and (a) 10^6 , (b) 10^7 , (c) 10^8 , and (d) 10^9 histories. Ideally, there should be no color wash present, which indicates no difference to the reference. As the number of histories increases, the visible amount of noise decreases. The dose differences range from 0 to 4.7 Gy.

Choosing the number of histories for the beamlets required a different analytical approach. Originally, 10^6 histories was chosen as these simulations were relatively fast to run. In post-processing the data, it was discovered that the resulting beamlet data had a small amount of noise, which can be seen in Figure 4.11. The figure shows color washes of the dose distributions of the same beamlet - one calculated using 10^6 histories and the other with 10^7 . By itself, the noise in a 10^6 history beamlet does not contribute a significant distortion in the data. When hundreds of beamlets converge on the region of the PTV, though, the noise propagates, and that area is reported as receiving a higher dose (2.7 Gy for the plan shown in the figure) than what is correct. This is shown in Figure 4.12, which shows color washes of dose contribution from beamlets only. The area defined by the PTV contour is receiving substantial false increase in dose resulting from the propagation of the noise of the 10^6 history beamlets when compared to the plan with the MLC leaf positions modified to out 0.5 cm.

Unlike the 10^6 history beamlets, the 10^7 history beamlets contain vastly reduced noise. Because of this, there is virtually no propagation of noise contributing to false dose in the region where the beamlets converge, as evident in Figure 4.13. In fact, using 10^7 histories for beamlet calculations seems to slightly underestimate the dose to that region by about 0.4 Gy. The magnitude of this is much less than the overestimation of the dose from the 10^6 history beamlets, though, so using 10^7 histories was clearly the better choice.

The false dose contributions from noise to the target area is further evident in Figure 4.14. This figure compares the DVH curves for the PTV of a brain VMAT plan resulting from adding the sum of the 10^6 history beamlet's doses and the sum of the 10^7 history beamlet's doses to the unmodified dose matrix, separately, and includes a DVH of the target volume resulting from a plan where

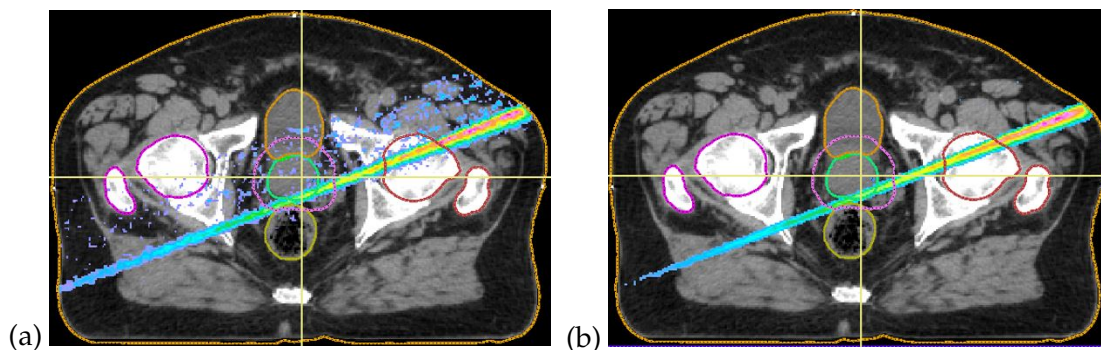


Figure 4.11: Relative dose color wash of a single beamlet in a prostate VMAT plan. (a) was obtained using 10^6 histories, and (b) was obtained using 10^7 histories. Noise is very apparent in (a) but is less apparent in (b). Increasing the number of histories ten fold decreased the variance in the data to the point where noise is not visible in the color wash. Note: noise visible outside the beamlet results from taking the difference between the unmodified and modified control point dose distributions, which is how beamlets are calculated.

all the included MLC leafs are moved out. There is relatively little difference between the modified plan dose distribution and the 10^7 history beamlets added to the original plan dose, but there is a notable difference between the modified plan dose distribution and the 10^6 history beamlets added to the original plan dose.

4.2.2.5 Running Modified Control Point MC Simulations on the Cluster

A VMAT plan typically consists of 2-4 arcs, each with 178 control points, with each control point involving 10-40 leaves. Therefore, there is a large number of beamlets associated with a VMAT plan. This number may be reduced by only considering the beamlets that irradiate the target structures. In addition, leaf motion that would violate leaf speed and collision constraints is excluded. The resulting number of beamlets required is generally 5,000-10,000. For this study, the MC beamlet calculations were parallelized on a computer cluster. This allowed simulations to be run in batches of 500, with further parallelization

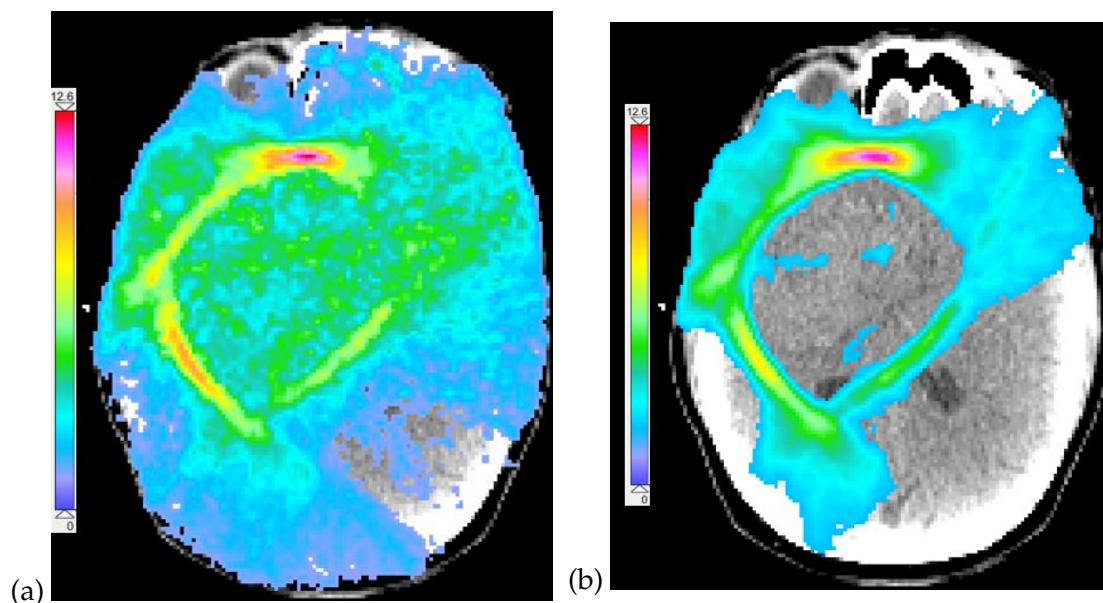


Figure 4.12: (a) Sum of the beamlets calculated for a brain VMAT plan using 10^6 for each beamlet. (b) Difference in dose matrices between the unmodified VMAT plan and the same plan modified with the non-excluded MLC leaves moved out 0.5 cm. Both (a) and (b) use the same color wash scale. The PTV is contoured in teal, which is the general area where the beamlets converge.

possible over multiple processing cores. The time required for a complete set of beamlet calculations (corresponding to one VMAT plan) was approximately 10 to 20 hours.

4.2.2.6 Beamlet Post-processing

Once all MC simulations are complete, the resulting 3ddose files take up an enormous amount of hard drive space. A single 3ddose file requires 200 MB, so thousands of them take up multiple terrabytes, which dwarfs most single PC storage capacities. To compensate for this, Matlab scripts were written to condense the beamlet data into convenient data structures.

A Matlab script imports the beamlet information file (jBLinfo) and stores the information as a set of variables. This information includes the designations

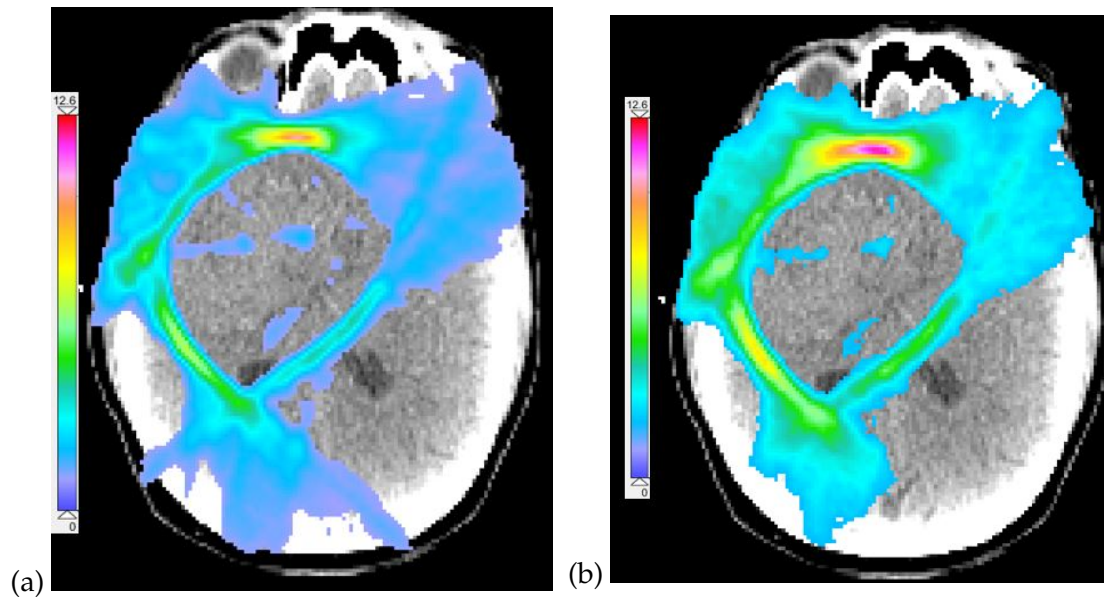


Figure 4.13: (a) Sum of the beamlets calculated for a brain VMAT plan using 10^7 for each beamlet. (b) Difference in dose matrices between the unmodified VMAT plan and the same plan modified with the non-excluded MLC leaves moved out 0.5 cm. Both (a) and (b) use the same color wash scale. The PTV is contoured in teal, which is the general area where the beamlets converge.

of each individual leaf included in beamlet calculations. Another Matlab script uses these variables to direct the data condensing process. During this process for a single beamlet, the dose data in the 3ddose files associated with the unmodified and modified control point is imported and stored as 3-D matrices. A simple element-by-element subtraction is done, which leaves only the perturbation dose information - the beamlet. An additional function was written to compensate for the noise resulting from this subtraction; a threshold dose is set such that any voxel in the beamlet matrix containing a dose below the threshold is set to zero. Since a vast majority of the voxels in the beamlet array are zeros, the array is converted to a sparse matrix where only the index of the voxel and the dose associated with said voxel are stored. This reduces the storage requirement for all of the beamlets calculated for a plan to just 25 MB, or about a tenth

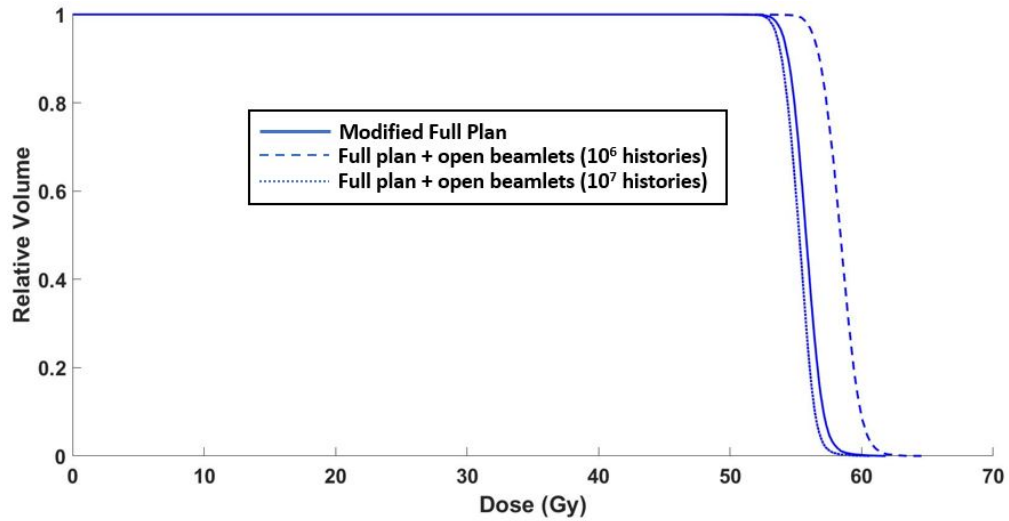


Figure 4.14: The DVH curves for a VMAT brain plan PTV resulting from all non-excluded leaves being moved out by 0.5 cm. The solid curve resulted from a single simulation - modifying the original VMAT plan parameters. The dashed and dotted curves resulted from adding the calculated beamlets to the original plan's dose matrix. In this example, the 10^6 history beamlets overestimate the dose to the PTV by about 2.7 Gy while the 10^7 history beamlets underestimate it by about 0.4 Gy.

the size of a 3ddose file.

4.2.3 Enhanced Optimization Objective Function

The original TPS dose distribution is modified by adding or subtracting beamlets, which corresponds to moving leaves out of or into their fields, respectively.

This permits a new *enhanced objective function* (EOF) to be defined:

$$f_E(\mathbf{x}) = \sum_j \sum_i W_j [(d_{j,min} - d_i(\mathbf{x}))^2 * H\{d_{j,min} - d_i(\mathbf{x})\} + (d_i(\mathbf{x}) - d_{j,max})^2 * H\{d_i(\mathbf{x}) - d_{j,max}\}] \quad (4.5)$$

This is similar to Eq.(2.1). The main differences are that n is the number of active leaves in the process, and

$$\mathbf{x} = [x_1, x_2, \dots, x_n] \quad (4.6)$$

is a vector of *ternary* leaf variables. In Eq. 4.5 the variable w_D can contain continuous values whereas here the variable \mathbf{x} contains discrete values. That is, each of the $x_i \in \{-1, 0, 1\}$ represents a possible leaf position modification. A -1 indicates a leaf moving into the field (i.e., the subtraction of a beamlet from the total dose matrix); a 0 indicates no change; and a 1 indicates a leaf moving out of the field (addition of a beamlet to the total dose matrix). Therefore, the ground state (i.e., the original and unmodified plan) is defined as:

$$S \equiv f_E(\mathbf{x}_0), \quad (4.7)$$

where $\mathbf{x}_0 = [0, 0, \dots, 0]$ is the value of the EOF applied to the original TPS-optimized plan (i.e., with no changes to the leaf positions). The EOF is a function of ternary, and thus discrete, variables, and therefore it may be optimized using a discrete optimization algorithm (see Discussion). In this study, we performed a simple *greedy* search, which iterated through each active leaf, one at a time, and calculated the EOF corresponding to the leaf moving in or out. A leaf position was saved if it resulted in a reduction (improvement) in EOF value, and rejected otherwise. In other words, we first check if

$$f_E(\mathbf{x}') < f_E(\mathbf{x}), \text{ for } \mathbf{x} = \mathbf{x}_0 \text{ and } \mathbf{x}' = [1, 0, \dots, 0] \text{ or } [-1, 0, \dots, 0]. \quad (4.8)$$

If that condition is met, we replace \mathbf{x} by \mathbf{x}' and check if

$$f_E(\mathbf{x}') < f_E(\mathbf{x}), \text{ for } \mathbf{x}' = [0, 1, \dots, 0] \text{ or } [0, -1, \dots, 0]. \quad (4.9)$$

This process iterates over every active leaf (i.e., 1 to n). The average run time of the greedy search was 30-45 minutes. More active leaves and increased number of objectives in the EOF led to longer greedy search times.

4.2.4 Evaluations

This study used retrospective comparisons in which actual clinical VMAT plans were subject to the Enhanced Optimization process. Seven VMAT plans (two adult brain, one 18-year-old pediatric brain, two head and neck, and two prostate) were selected at random from the Eclipse database. Table 4.2 shows information on the number of objectives and ternary EO variables of each plan. A standard evaluation was performed on every plan, employing the same OAR dose-volume objectives as in the TPS plans. In order to investigate further potential improvements, the pediatric brain plan was subject to additional evaluations in which the left cochlea PRV objective was reduced (i.e., made stricter). These specific objectives are shown in Table 4.3. This objective was selected because it was the only one not satisfied in the original TPS plan. In all cases, the PTV objectives were modified because the original planning required certain optimization strategies, which were not required in the EO process (see Discussion).

For each case, the results of the EO were used to modify the control point information in the plan DICOM file, which was then imported into Eclipse. This allowed the plan to be recalculated with any linac and leaf-motion mechanical constraints applied. The EO plan could then be compared to the original Eclipse

Table 4.2: Designations VMAT plans selected for this study and their associated number of arcs, objectives, and variables (i.e., the number of active leaves for which beamlets were calculated).

Plan	Arcs	Objectives	Variables
Brain 1	2	17	2733
Brain 2	2	23	3831
Ped. Brain	2	19	2714
H&N 1	2	16	4548
H&N 2	2	17	2974
Prostate 1	2	17	5099
Prostate 2	2	8	3234

Table 4.3: Objective modification types used for the pediatric brain case. The objective is the maximum dose to the left cochlea PRV.

Objective	Maximum
Unmodified	35.0 Gy
Type 1	33.5 Gy
Type 2	29.0 Gy

plan by reviewing DVH's and isodose curves. To simplify the comparison, the EO plan was normalized so that 95% of the PTV received the same dose as in the corresponding Eclipse plan. This was done to ensure equivalent target coverage between the original and EO plans as loss of this coverage can make a plan less clinically acceptable.

4.3 Results

Most of the evaluation results are presented with DVH curves corresponding to the clinical plans produced by the TPS (labeled "Original") and the plans produced by the enhanced optimization procedure (labeled "EO"). It is emphasized that all DVH's correspond to the *final, deliverable plans*, extracted from Eclipse. Some cases, particularly the brain and head-and-neck cases, involve

many OAR's. For easier visualization, a DVH is only displayed if the OAR's objective penalty changed after the EO. Structures with no DVH's displayed may be assumed to have had their objectives met before and after the EO.

4.3.1 Enhanced Optimization with Original Objectives

Table 4.4 presents a numerical summary of the plans evaluated, showing the objective score before and after the EO process, along with the time required for the greedy search to complete.

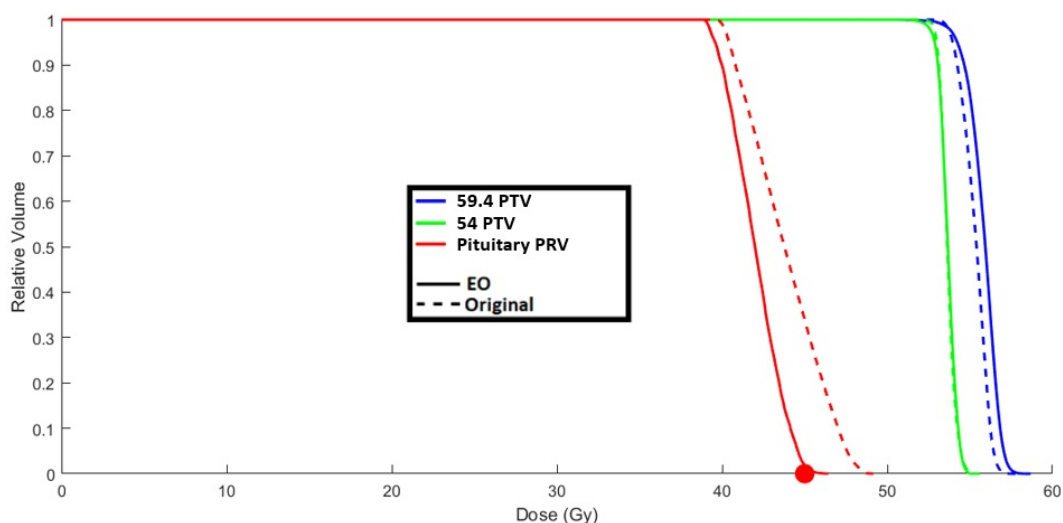


Figure 4.15: DVH curves for the first brain VMAT plan before (original) and after EO. The pituitary PRV improved, with a small increase to the 59.4 PTV. OAR's with objectives already met before and after EO are not shown.

The first brain VMAT plan had non-overlapping PTV's (Figure 4.15). The EO produced virtually no change to the 54 PTV, while slightly increasing the dose coverage to the 59.4 PTV. Improvement to the pituitary PRV DVH curve was apparent. Other structures saw some improvement to their DVH curves; however, their objectives were met prior to EO (i.e., they contributed no penalty score), and therefore are not shown in the DVH plot.

Table 4.4: Summary of the enhanced optimization results with original objectives. Penalties are calculated with MC data applied to Eq. 4.5, and dose values are extracted directly from Eclipse.

Brain 1							
OAR	<i>Objective</i>		<i>MC Penalties</i>		<i>Doses (calc'd by Eclipse)</i>		
	Dose (Gy)	Volume	Original	EO	Original (Gy)	EO	Diff. (Gy)
Pituitary PRV	45	0	21219	382	49.28	46.41	2.87

Brain 11							
OAR	<i>Objective</i>		<i>MC Penalties</i>		<i>Doses (calc'd by Eclipse)</i>		
	Dose (Gy)	Volume	Original	EO	Original (Gy)	EO (Gy)	Diff. (Gy)
R. Cochlea PRV	36.5	0	4110070	3769964	62.25	62.18	0.07
Brainstem PRV	54	0	27239	18862	57.75	57.59	0.16
R. Lacrimal	36	0	91774	5990	35.93	34.41	1.52
R. Lacrimal PRV	36	0	37206	24284	46.02	45.73	0.29
Pituitary PRV	45	0	33038	18201	53.25	51.94	1.31

Pediatric Brain 1							
OAR	<i>Objective</i>		<i>MC Penalties</i>		<i>Doses (calc'd by Eclipse)</i>		
	Dose (Gy)	Volume	Original	EO	Original (Gy)	EO	Diff. (Gy)
L. Cochlea PRV	35	0	2000	729	39.35	38.49	0.86

Head and Neck 4							
OAR	<i>Objective</i>		<i>MC Penalties</i>		<i>Doses (calc'd by Eclipse)</i>		
	Dose (Gy)	Volume	Original	EO	Original (Gy)	EO (Gy)	Diff. (Gy)
Anterior Tongue	74.5	0	11716	2944	76.11	75.32	0.79
R. Cochlea	35	0	1700	35	38.08	33.66	4.42
L. Parotid	59.5	0	30146	2447	62.69	61.11	1.58

Head and Neck 13							
OAR	<i>Objective</i>		<i>MC Penalties</i>		<i>Doses (calc'd by Eclipse)</i>		
	Dose (Gy)	Volume	Original	EO	Original (Gy)	EO (Gy)	Diff. (Gy)
L. Cochlea	30	0	0	0	29.29	26.63	2.66
Ring 80	36	0	11765	10403	36.48	35.52	0.96
Ring 50	21.5	0	84075	76533	21.51	21.17	0.34

Prostate 3							
OAR	<i>Objective</i>		<i>MC Penalties</i>		<i>Doses (calc'd by Eclipse)</i>		
	Dose (Gy)	Volume	Original	EO	Original (Gy)	EO (Gy)	Diff. (Gy)
Bladder	53.4	0	18132	5321	54.5	54.25	0.25
Penile Bulb	53.4	0	2355	46	53.82	53.68	0.14
Rectum	53.4	0	48408	11547	54.61	54.58	0.03

Prostate 11							
OAR	<i>Objective</i>		<i>MC Penalties</i>		<i>Doses (calc'd by Eclipse)</i>		
	Dose (Gy)	Volume	Original	EO	Original (Gy)	EO (Gy)	Diff. (Gy)
Rectum	74	0	128	27	74.31	74.13	0.18

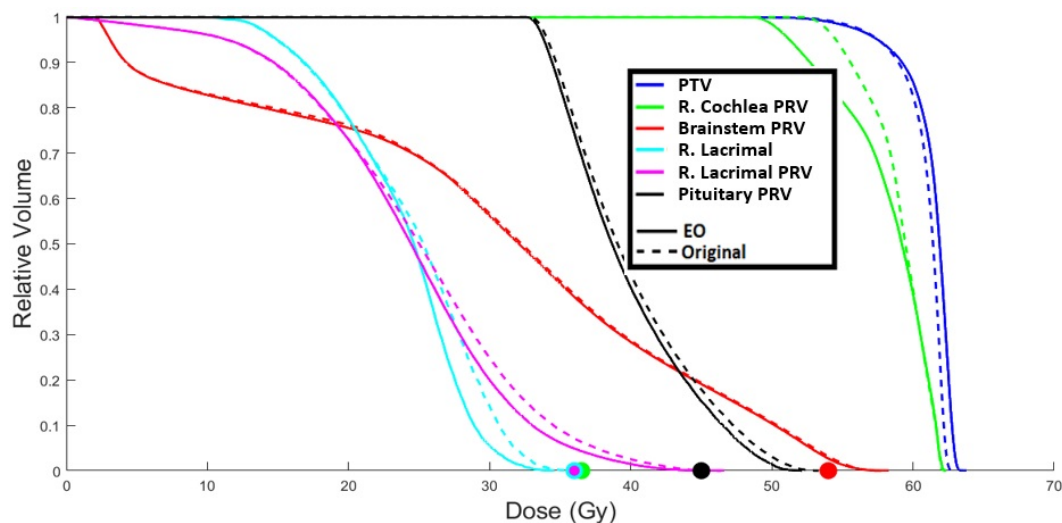


Figure 4.16: DVH curves for the second brain VMAT plan. The cochlea, lacrimal, lacrimal PRV, and pituitary were improved by the EO. Note that the lacrimal and lacrimal PRV dose objectives are identical, and are very close to the right cochlea PRV objective.

The second brain VMAT plan (Figure 4.16) also saw a small decrease in dose homogeneity of the PTV while having noticeable improvement to the right cochlea PRV. Additionally, the right lacrimal and its PRV were improved. The pituitary also improved slightly, and the brainstem was virtually unchanged.

The pediatric brain VMAT plan (Figure 4.17) had Eclipse objectives for the GTV instead of the PTV. There was a slight improvement to the left cochlear PRV, while the GTV was virtually unchanged. This plan was further investigated below.

The first head and neck VMAT plan had a 70 PTV (to be boosted in a future plan) inside a 56 PTV (Figure 4.18). There was a small improvement to the left parotid and larger improvements to the right cochlea. The 70 PTV was slightly less homogeneous.

The second head and neck VMAT plan (Figure 4.19) two “trick” planning structures, known as Ring 80 and Ring 50, to optimize the 80% and 50% isodose

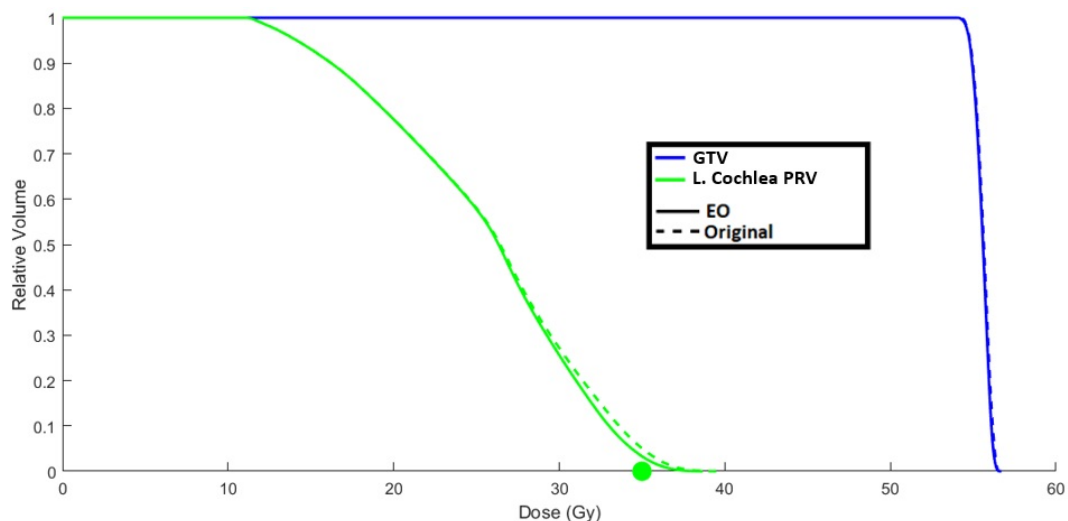


Figure 4.17: DVH curves for the pediatric brain VMAT plan. There was a slight improvement to the left cochlea PRV.

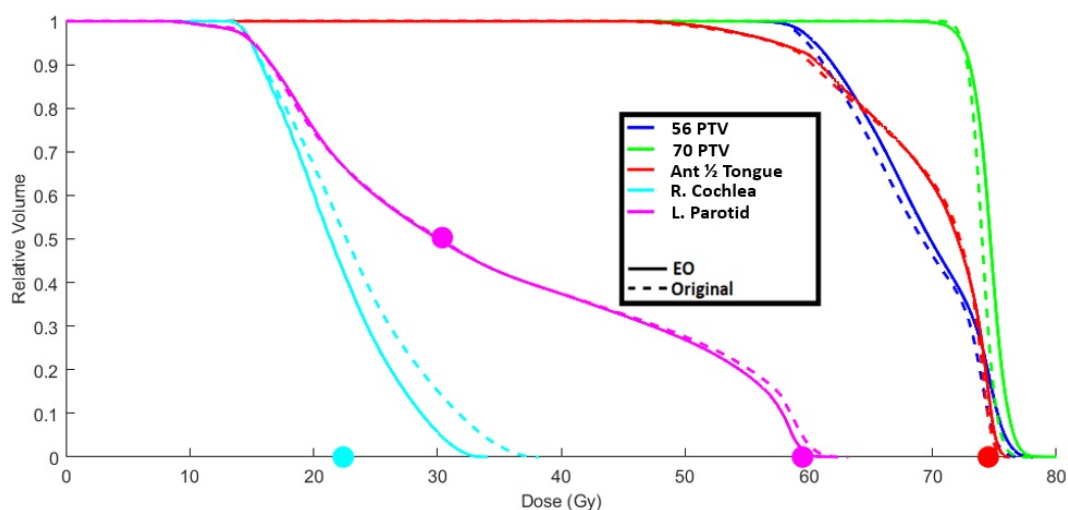


Figure 4.18: DVH curves for the first head and neck VMAT plan. There was an improvement to the right cochlea and a small increases in both PTV doses.

lines, respectively. These RING STRUCTURES ARE known as normal tissue objectives (NTO). The purpose of these NTO's is to add additional constraints to the optimization that ensure a rapid dose falloff outside the target. Traditional OARs were included in the optimization as well; however, only the NTOs received any penalties. Improving the scores to these structures resulted in a size-

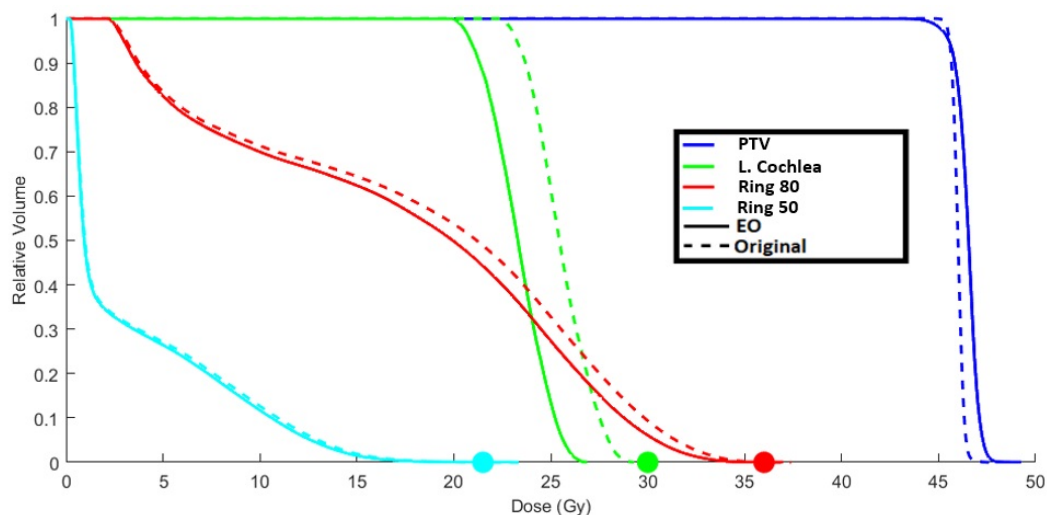


Figure 4.19: DVH curves for the second head and neck VMAT plan. Improvements to the NTO planning structures (Rings) produced improvements to the left cochlea.

able improvement to the left cochlea.

The two prostate cases are shown in (Figure 4.20) and (Figure 4.21). The DVH curves of both cases remained virtually unchanged by the EO process. This may be due to the Eclipse optimizer locating very stable minima for these relatively simple cases. The substantial reduction in objective scores are due to limitations in the Eclipse dose calculator (see Discussion).

4.3.2 Enhanced Optimization with Modified Objectives

The left cochlea PRV objectives of the pediatric brain case were modified in two ways, as shown in Table 4.3, with the results shown in Figure 4.22. The type 1 modification produced no change in the ternary optimization variables in Eq.(4.5). In other words, a reduction in maximum dose objective from 35 to 33.5 Gy for that single OAR did not produce a difference in penalty when the dose distributions were determined by the MC calculations, and therefore re-

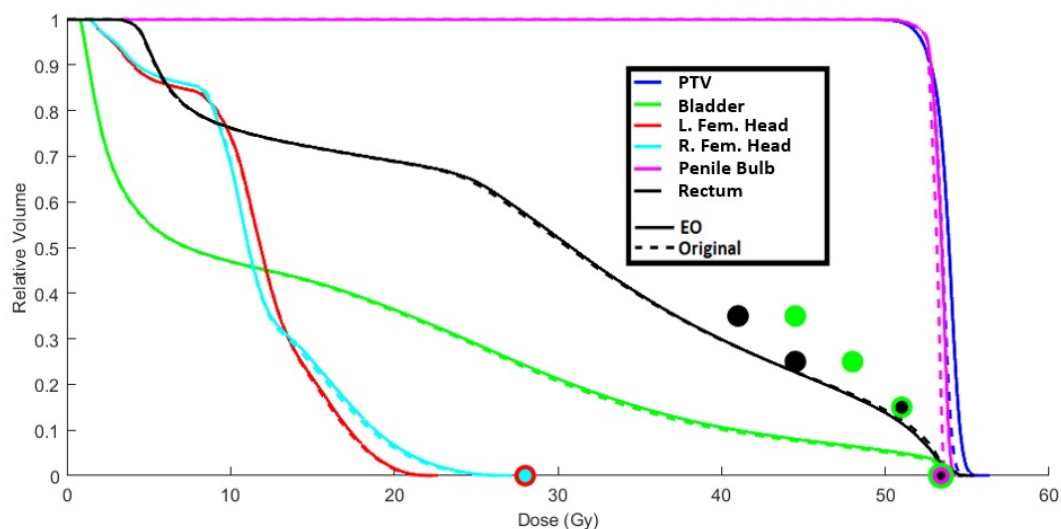


Figure 4.20: DVH curves for the first prostate VMAT plan, showing very little change with the EO. Note that the max dose objective for the bladder, rectum, and penile bulb are identical. The left and right femoral head dose objectives are also identical.

sulted in the same EO plan (see Discussion). The type 2 modification reduced the objective from 35 to 29 Gy, resulting in a noticeable improvement to the DVH. The isodose curves for these cases are shown in Figure 4.23. The PTV DVH curve was virtually unchanged for each type of objective modification. Table 4.5 shows the plans which were created for this case: Eclipse, EO Unmodified, EO Modified (Type 1), and EO Modified (Type 2). The scores of each plan, where applicable, are indicated for the various objective functions: Original, Modified (Type 1), and Modified (Type 2). Additionally, Table ?? shows LSKDJFLDKJFSLDJ

It should be noted that similar objective modifications were attempted for one of the prostate cases. They resulted in slight improvements to the rectum dose, as shown in Figure 4.24, but substantially less homogeneous PTV dose (see Discussion).

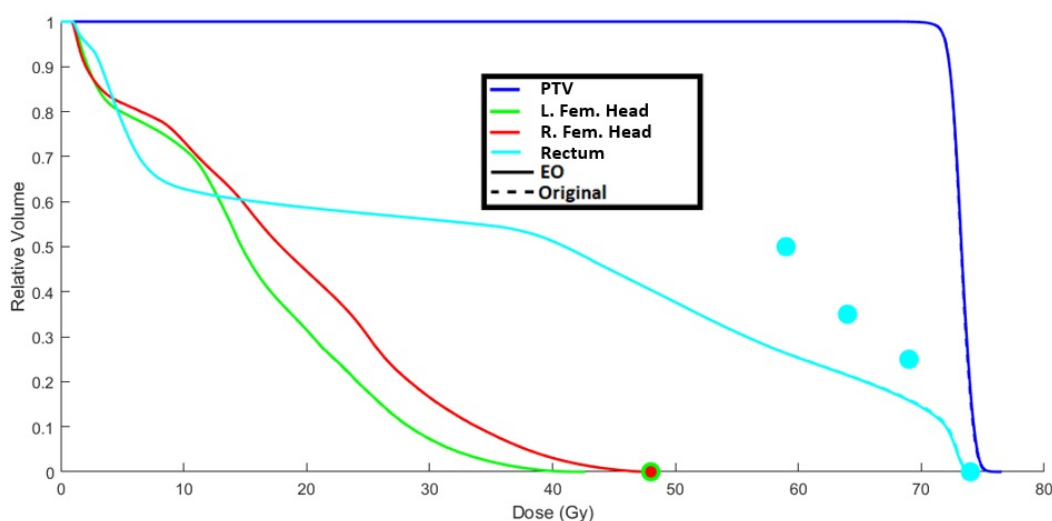


Figure 4.21: DVH curves for the second prostate VMAT plan, showing no change with the EO. Note that the femoral head objectives are identical.

Table 4.5: Objective function scores for pediatric brain plans. The EO plans correspond to different objectives on one OAR. "Unmodified" and "Type 1" resulted in the same plan after the greedy search was completed. The "Type 2" DVH was only different in high-dose region (see text). All EO plans had lower scores than the Eclipse plans.

Plan	Original Score	Modified Type 1 Score	Modified Type 2 Score
Eclipse	3109	24415	92495
EO Unmodified	1243	20137	72213
EO Mod. Type 1	1243	20137	72213
EO Mod. Type 2	1243	20137	69746

Table 4.6: Summary of the pediatric brain plan's enhanced optimization results with original, "Type 1" and "Type 2" objectives applied to the left cochlea PRV. Penalties are calculated with MC data applied to Eq. 4.5, and dose values are extracted directly from Eclipse.

Pediatric Brain - L. Cochlea PRV Modded Objectives							
Objective Set	Dose (Gy)	Volume	MC Penalties		Doses (calc'd by Eclipse)		
			Original	EO	Original (Gy)	EO (Gy)	Diff. (Gy)
Original	35	0	2000	729	39.35	38.59	0.76
Type 1	33.5	0	7462	3620	39.35	38.59	0.76
Type 2	29	0	74490	51985	39.35	38.22	1.13

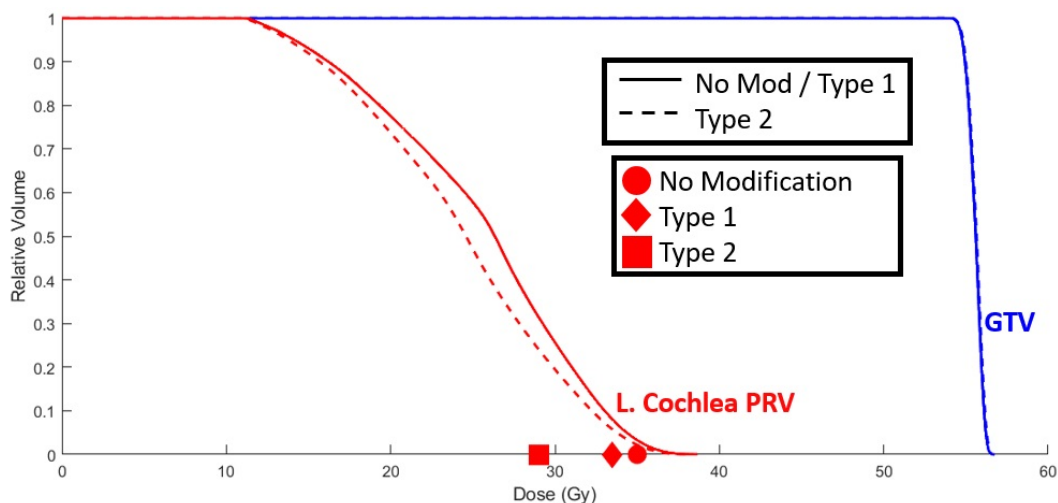


Figure 4.22: DVH curves for the pediatric brain case after the EO was applied. "No Mod" indicates the original objectives and subsequent EO plan, while "Type 1" and "Type 2" indicates modifications to the left cochlea objective to make it progressively stricter. Type 1 plan was the same as the No Mod plan, as explained in Discussion.

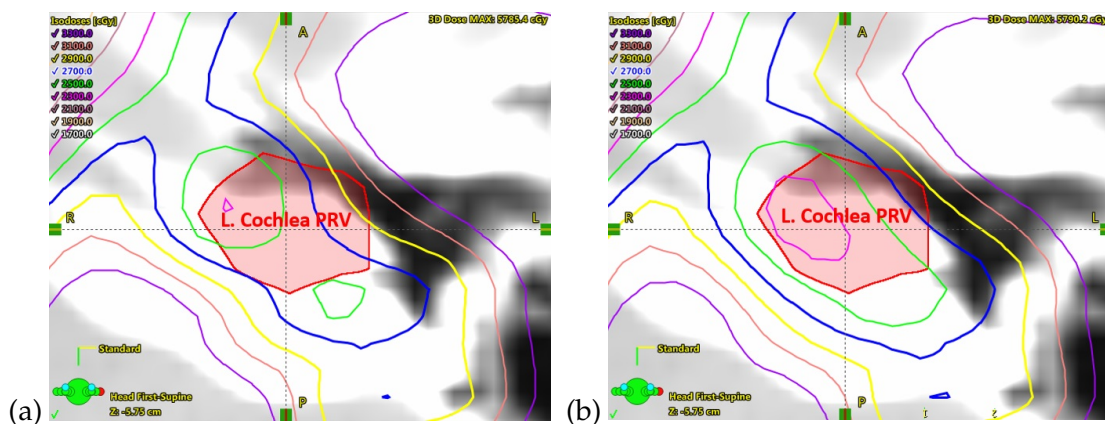


Figure 4.23: Isodose curves for the pediatric brain case, showing the left cochlea PRV, which is shaded. The no-modification and Type 1 plans (a) are identical. The Type 2 plan (b) shows sparing of the left cochlea PRV; there is reduction in area covered by the 27 and 29 Gy isodose lines in favor of the 23 and 25 Gy isodose lines. Note that the isodose curves indicate a lower dose toward the center of the structure.

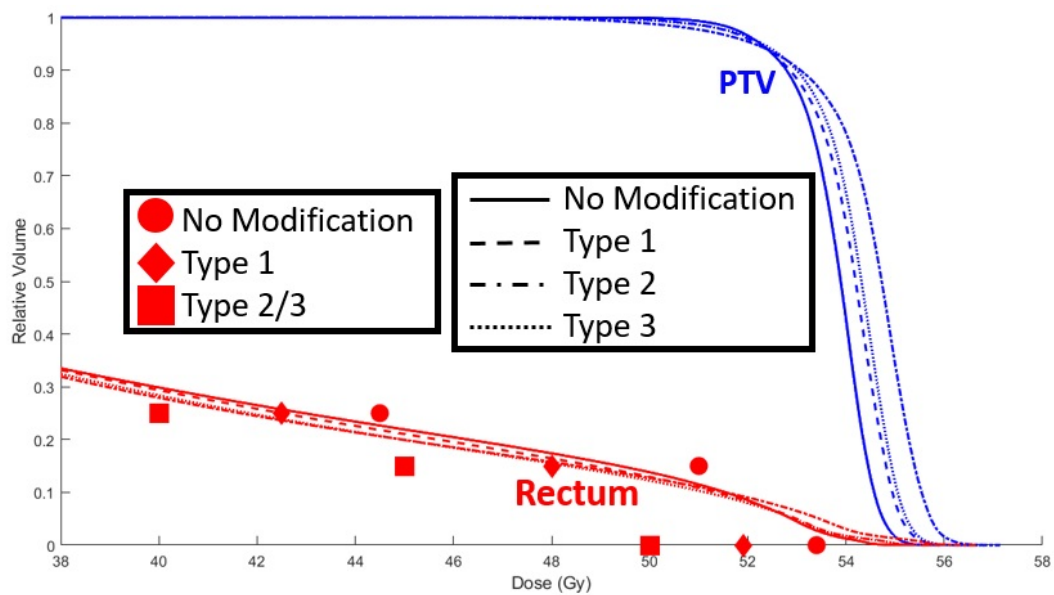


Figure 4.24: An enlargement of the rectum and PTV DVH curves for the prostate case after various modifications were made to the objectives. No Mod indicates the original objectives, while Type 1 and Type 2 objectives are progressively stricter. The Type 3 dose objectives were the same as Type 2, but with reduced priority weights. In every plan, reduction of the rectum dose resulted in compromised PTV dose homogeneity.

4.4 Discussion

4.4.1 Clinical Implications

In every evaluation performed, the enhanced optimization process was able to achieve a more optimal dose distribution, as indicated by the scores in Table 4.4. This was particularly true for the complex cases (e.g., brain and head and neck), whose Eclipse plans did not meet one or more of their objectives. However, an EO score improvement does not provide a complete picture of the plan quality. Both prostate case scores were substantially improved (by 46% and 79%), and yet their DVH's remained virtually unchanged, when Eclipse was used to calculate the EO DVH's. This is the result of differences in dose calculations between the Eclipse and MC algorithms, as explained below in 4.4.2.

From an optimization standpoint, the prostate cases are relatively simple. The targets are approximately spherical and are centrally located. The adjacent OAR's (bladder and rectum) are relatively large structures. This offers more flexibility to the optimizer in terms of potential locations within the OAR in which to spread dose. It also results in the OAR DVH being less volatile when subject to small changes to the dose distribution. The small dose perturbations used by the EO process will not substantially change the OAR DVH's. In contrast, brain and head and neck cases involve irregularly-shaped targets, which may be adjacent to small OAR's. Small changes to the dose may result in large changes to the OAR DVH's, and therefore to the plan score. Therefore, the EO may offer a greater benefit to complex plans, because the dose perturbations permit the system to explore the neighboring optimization space for improvements.

It is also apparent that relative raw score reduction had less impact on plan enhancement than absolute score reduction. For example, the score for the pediatric brain case was reduced by 60% (1866 points), whereas the second brain case was reduced by 6.5% (31215 points). However, the DVH's for these cases indicate that the second brain case benefited more from EO. This is also true in the second prostate case, whose absolute score reduction of 183 points (the lowest of the 7 cases), translated into a relative reduction of 78.6%, (the highest of the cases).

We believe EO has great clinical potential, especially for plans with an OAR which may be of particular concern. For example, a radiation oncologist in our center indicated that sparing of the cochlear structures in a pediatric brain patient is critical to a lifetime of preserved hearing function. This provided the motivation to apply the modified objective EO process to this particular structure. When such a case is identified, the planning may be performed as usual, and then the EO can be run overnight, for comparison with the original plan the next day. The process may be greatly streamlined and automated through the use of scripting.

The VMAT plans evaluated for this study all involved 6-MV energies and a Varian Trilogy linac. However, we have since employed the BEAMnrc module to model a TrueBeam linac with a high-definition MLC, and incorporated phase space files to support MC calculations with higher energies and flattening-filter free treatment modes. These new calculation capabilities will permit the investigation of EO applied to SBRT VMAT plans, in which DVH planning objectives are critical.

The beamlet calculation is the first step of the EO, but it is only required once, before the optimization is performed. Currently, the maximum time re-

quired to compute beamlets is approximately 18 hours for one plan. However, this includes time spent in the computer cluster's batch queue. A dedicated cluster and efficient parallelization of jobs would reduce this time, as would ever-increasing processor power. Once the beamlet matrices are calculated, the greedy search requires only 30-90 minutes, as shown in Table 4.4. The greedy search was selected due to its simplicity; however, any discrete optimization algorithm which can be applied to ternary variables may be used. Genetic algorithms and other methods are currently being investigated.

The beamlets act as perturbations to the original TPS-optimized plan. The width of the beamlets in the isocenter plane is determined by the width of the central leaves of the MLC, which was 0.5 cm in this study. This may be modified for other MLC types (e.g., 0.25 cm for high-definition MLC's). The length of the beamlets (i.e., in the direction of leaf travel) was selected to be 0.5 cm, although this value may be adjusted based on the plan type. For example, as discussed above, prostate plans are less sensitive to perturbations to the dose distribution, and may respond to larger beamlets in the EO process. Plans involving very small target structures and OAR's, such as those used for SBRT and SRS cases, may benefit from smaller beamlets. A further possibility is the use of variable beamlet sizes, in which various-length beamlets are calculated (e.g., 0.2 cm to 1.0 cm, in steps of 0.2 cm). This will increase the number of beamlets and their calculation time, and change the EO variable type from ternary to a larger discrete type. This is a possible area of future investigation.

As shown in this study, DVH objectives may be modified from their original TPS values. This has the effect of changing the objective function, and therefore the optimization space. However, it only requires a new greedy search, not a recalculation of the beamlet matrices; various combinations of objectives

and weightings may be investigated in this way. The modified pediatric brain cochlear objective did reduce the dose to the left cochlea. However, the prostate plan rectum objectives could not be reduced without incurring a substantial reduction to the homogeneity of the PTV dose. This demonstrates that the EO may open up a new area of treatment planning, in which a planner may acquire skill and experience in both the TPS and enhanced optimization processes.

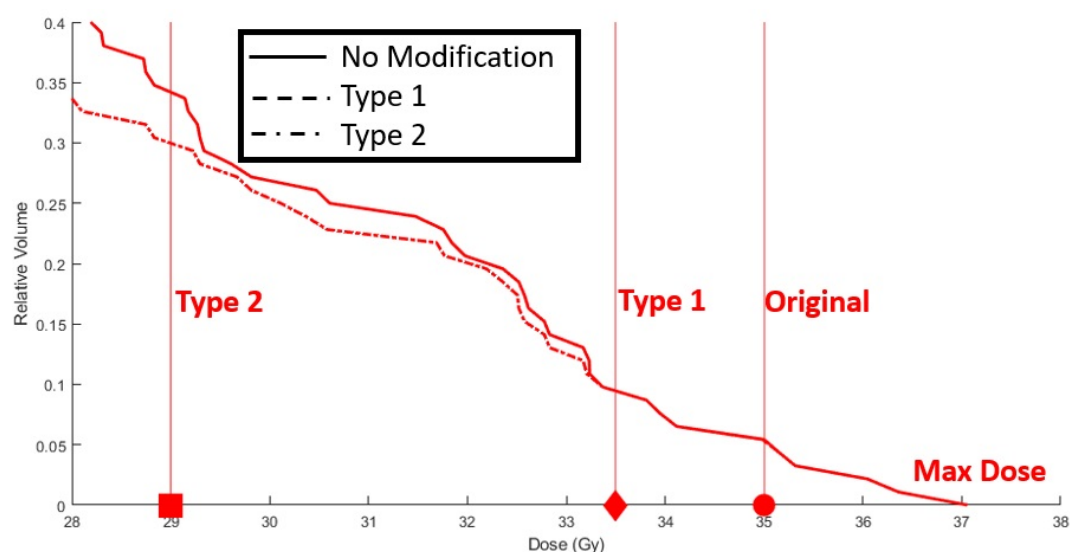


Figure 4.25: The MC-calculated DVH curves of the pediatric brain case's left cochlea PRV resulting from the original EO, type 1, and type 2 plans. The DVH's for the original EO and type 1 plans are identical, and the type 2 plan's DVH coincides with the other two at doses greater than about 33.1 Gy. Thus, the region under all three curves evaluated using the original and type 1 objective functions are identical and produce the same penalties.

4.4.2 Differences in Eclipse and MC Calculations

The EO process is based on doses calculated for VMAT plans using MC algorithms. In most cases, the plan dose values agree well with those calculated by the TPS(6). The EO also requires the calculation of DVH's for the VMAT plans, which also generally agree with the TPS DVH's. However, this study found

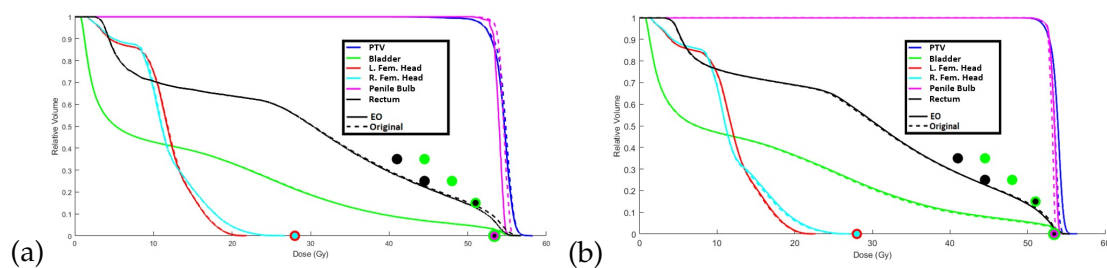


Figure 4.26: DVH curves for the same dose matrices calculated using the same plan parameters before and after EO by (a) MC and (b) Eclipse. Differences are especially apparent for the rectum and penile bulb. Note that the max dose objective for the bladder, rectum, and penile bulb are identical. The left and right femoral head objectives are also identical.

some instances in which there were differences between the EO and Eclipse DVH's. That is, after the EO was applied to the plan, the new plan with the modified control points was imported into Eclipse. The dose was calculated, and the DVH data was exported for comparison with the EO DVH data. Figure 4.26 shows a prostate plan comparison. Slight differences to the maximum doses to the rectum and penile bulb are evident. Since there are maximum-dose objectives to these OAR's, the plan is optimized according to the EO, but not necessarily according to Eclipse. Therefore, the Eclipse plan may still be selected by the radiation oncologist, even though the EO plan is in fact of higher quality.

Another issue that was encountered involved running the EO with the modified objectives for the pediatric brain case. As shown in Figure 4.25, the Unmodified (35 Gy), Type 1 (33.5 Gy), and Type 2 (29 Gy) maximum-dose objectives were investigated for the left cochlea PRV. The Unmodified and Type 1 objectives produced identical plans, as indicated by the DVH's. This was also verified by examining the values of x in Eq.(4.5). However, using the Type 2 objective resulted in a reduction in dose to the OAR. This plan was not found

during the Type 1 optimization, because it produces the same objective penalty as that of the Unmodified plan. Both Type 2 and Unmodified DVH's coincide at doses above about 33 Gy. This is the reason Table 4.5 has identical values for the EO Unmodified and EO Modified Type 1 plans. Therefore, a sufficient reduction in the objective is required to produce a change in the plan.

Chapter 5

Application of the Hamiltonian Engine for Radiotherapy Optimization on Prostate IMRT Plans

You've seen these movies where they say "Make my day" or "I'm your worst nightmare"? Well, listen to this one: Rubber baby buggy bumpers.

Jack Slater - Last Action Hero

(1993)

5.1 Motivation

Much research has been done in intensity modulated radiotherapy (IMRT) optimization algorithms. Implementation of these new algorithms tends to be in one of two groups. The first group, which is the most used commercially, are the algorithms that employ gradient-based methods in solving the IMRT optimization problem. These methods can reach a solution quickly; however, they have a tendency to report a non-optimal local minimum solution instead of the system solution (i.e., the global minimum) (44). The second group belongs to the stochastic methods, such as simulated annealing. During their search for a solution, these methods are capable of escaping local minimal due to the implementation of a randomizing element; however, they are much slower than their gradient-based counterparts (45).

The University of Toronto (Toronto, Ontario, Canada) and Fujitsu Ltd. (Tokyo, Japan) have been working together to develop quantum computing inspired hardware that can perform very rapid calculations. During this development process, they have reached out to various industries for potential applications for these innovations. One such idea was presented by our group: radiotherapy plan optimization. In response, the Hamiltonian Engine for Radiotherapy Optimization (HERO) hardware was developed for the IMRT optimization problem. HERO was designed to quickly and accurately solve quadratic unconstrained binary optimization (QUBO) problems. Because of this, the IMRT optimization problem was broken down into small parts so it could be represented as a QUBO problem.

Another novel hardware approach, involving quantum computing, was explored in a previous study (22). This approach, known as quantum annealing

(QA), similarly adapted the IMRT optimization problem so it could be solved using the proprietary hardware. The study ultimately concluded that, until improvements could be made to the QA hardware, the results were not clinically useful. Although the QA was able to optimize a prostate IMRT problem, it requires special hardware and considerations, namely cooling, and the process could only handle a very limited number of variables (around 70). Additionally, the time to reach a solution was relatively long (about 3 hours).

The HERO has some advantages over the QA that increases its potential clinical viability. The HERO utilizes conventional hardware, so it does not require special considerations such as extreme cooling; it can run at room temperature. Additionally, using the QUBO approach, the HERO is capable of accommodating many more variables (on the order of thousands). The HERO is also able to reach a solution to the IMRT QUBO problem very quickly (on the order of seconds).

For this work, seven prostate patient plans were selected, and each underwent optimization using techniques described in Materials and Methods. The prostate site was selected due to it presenting a relatively simple optimization problem due to its large, far separated structures. In the future the HERO's performance on treatment sites that present more complex optimization problems, such as the brain and head and neck, may be evaluated.

The HERO's potential applications does not end at IMRT optimization. In fact, this study was intended to be a stepping stone to potential VMAT optimization. VMAT optimization is significantly more complex optimization problem; however, with more adaptation, it is likely that, should the HERO be successful in optimizing IMRT plans, HERO could be applied to optimizing VMAT plans.

5.2 Materials and Methods

5.2.1 Using CERR to Obtain Pencil Beam Information for Optimization

As the TPS used clinically (i.e., Eclipse) does not have the option to export individual pencil beams (PB's) for dose calculation, another method was used. The Computational Environment for Radiotherapy Research (CERR) (42) has a built-in IMRT planning module. This module is incomplete as it lacks an optimization component; however, the options included allow for calculation of PB's using a quadrant infinite beam (QIB) algorithm. The QIB algorithm is a relatively simple way to calculate PB's that only requires a few table lookups and some arithmetic operations (46; 47).

PB parameters were selected to follow traditional prostate IMRT planning practices. Five 6 MV beams at gantry angles 0° , 72° , 144° , 216° , and 288° were planned. PB sizes are set to $0.25 \times 0.25 \text{ cm}^2$, which are the same dimensions as the Eclipse PB's. A default dose color wash can be seen in Figure 5.1. Here, each PB was weighed evenly (i.e., 1), and the fields themselves are considered "open" as they do not require MLC leaf movement.

The dose contribution of each PB is stored in its own three-dimensional matrix. A single PB matrix is known as an *influence*. This is not to be confused with *fluence*. A fluence is the collection of the weights of each PB in a single beam. The fluences of a plan can be combined into a single vector, \mathbf{w} . Depending on the optimization process, the weights in \mathbf{w} may be discrete or continuous. Thus, the dose matrix of the plan is the linear combination of the weights and influences

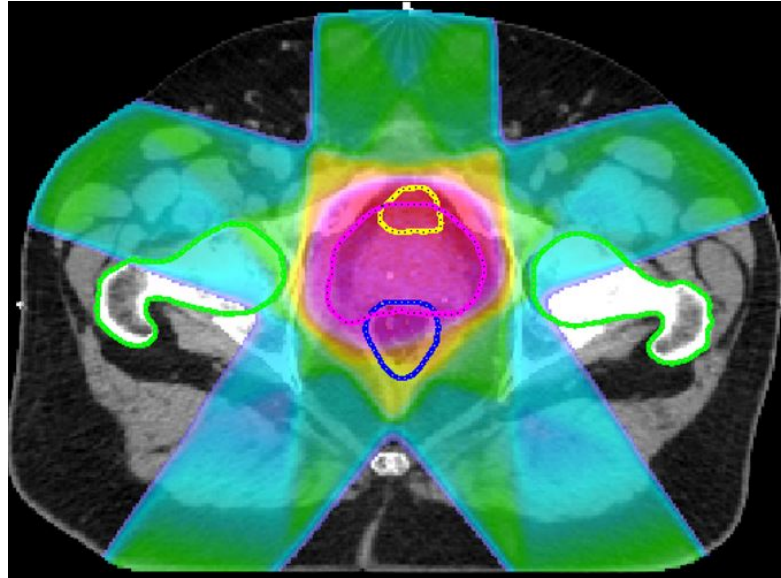


Figure 5.1: Dose color wash for all five beams of an IMRT prostate patient with default weights. The pink contour is the PTV, the yellow is the bladder, the blue is the rectum, and the green are the femoral heads.

(PB dose matrices) such that:

$$\mathbf{w} = [w_1, w_2, \dots, w_n]$$

$$D = w_1 D_1 + w_2 D_2 + \dots + w_n D_n \quad (5.1)$$

where D_i is the influence matrix of PB i and n is the number of PB's. There is one fluence per beam. By default, each weight is set to one such that:

$$\mathbf{w}_{default} = [1, 1, \dots, 1]$$

These fluences are graphically represented in Figure 5.2 (numeric) and Figure 5.3 (color wash). It is easiest to visualize a fluence from the beam's eye view, which is shown in the figures. A fluence map is represented by a grid, and each PB is an element in the grid. Some elements on the periphery of the grid are excluded

if they do not contain at least part of the two-dimensional projection of the PTV. A Matlab script assigns corresponding weight and influence an index so they are accounted for during the optimization process. It should be clarified that the PB's used here are different and independent of the beamlets described in Chapter 4. PB calculations done in this chapter are relatively simple and fast, and the sum of the PB's make up the entire IMRT plan dose distribution. The beamlets described in Chapter 4 are calculated using MC and are only used as perturbations to the plan's dose matrix.

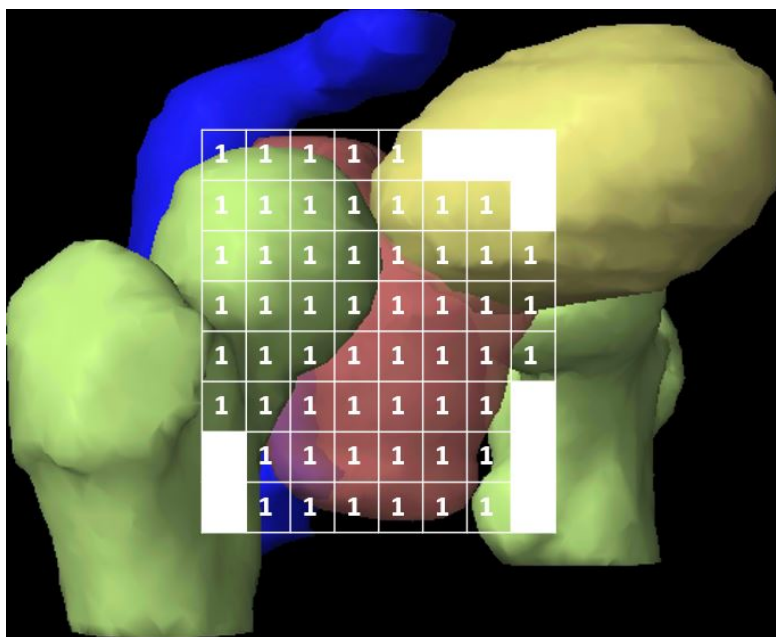


Figure 5.2: Beam's eye view of a single fluence (gantry at 288°) of an IMRT prostate patient with default weights. The fluence is represented by a grid pattern where each element corresponds to a single PB. The whited-out elements do not contain at least part of the two-dimensional projection of the PTV, so there are no PB's associated with those elements. The pink structure is the PTV, the yellow is the bladder, the blue is the rectum, and the green are the femoral heads.

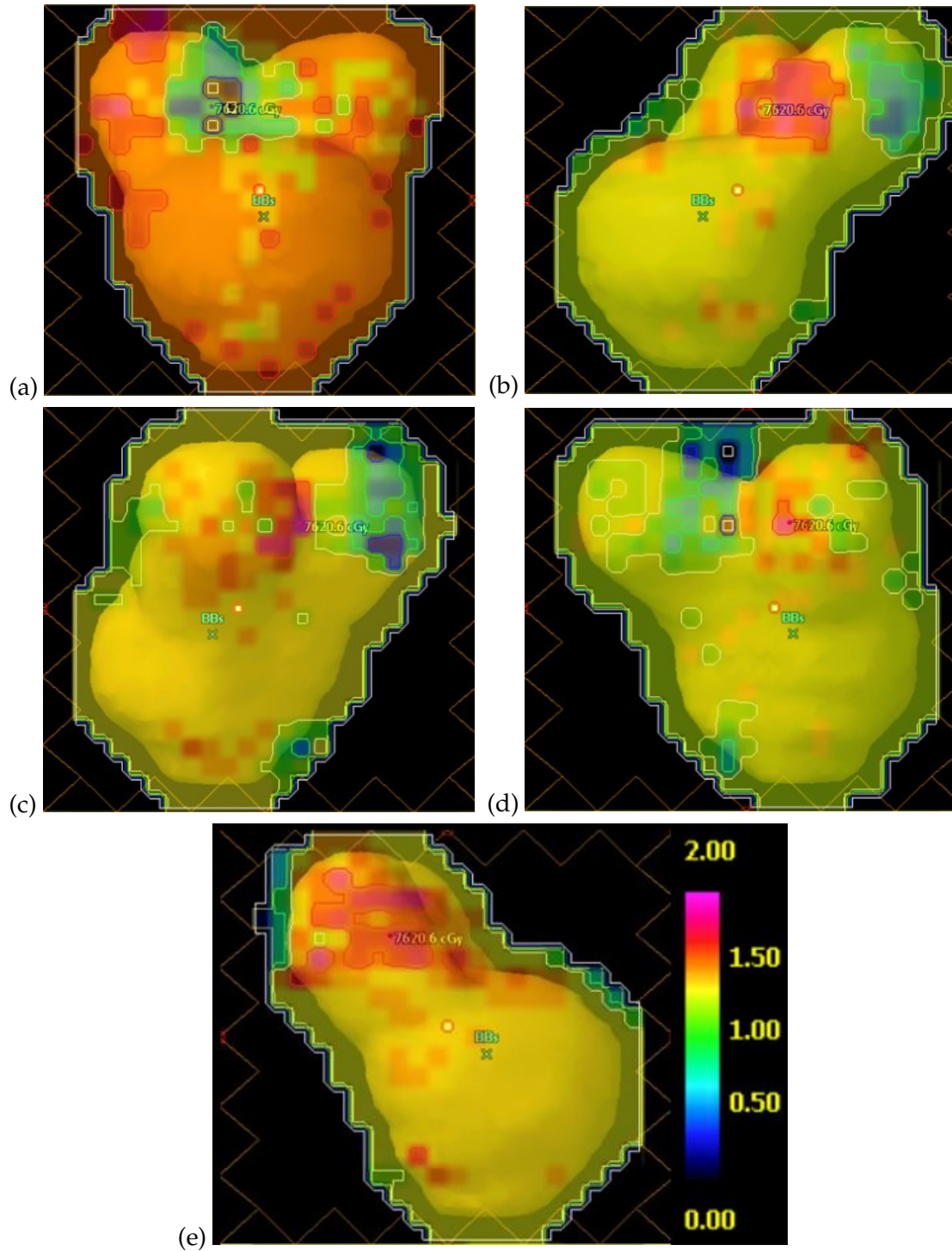


Figure 5.3: Beam's eye view of post-optimization fluence color wash of a prostate IMRT case at gantry angles (a) 0° , (b) 72° , (c) 144° , (d) 216° , and (e) 288° . Only the PTV structure is shown to illustrate that the fluence contours to the shape of the PTV projected to the corresponding beam's eye view.

5.2.2 Optimization of IMRT Plans with Eclipse

Prostate IMRT plans with parameters (e.g., fields, gantry angles) identical to the plans created by CERR were made in Eclipse. Following the typical treatment planning flow, objectives were entered into the built in optimization algorithm - Dose Volume Optimizer (D.V.O. - v15.1.51). As Eclipse is a black-box type application, the objective function used is unknown to the user. Eclipse's optimizer includes a graphical progress indicator, which can be observed for improvement progress. Upon completion, the resulting optimized fluences were exported to a text file.

5.2.3 Optimization of IMRT Plans with the HERO

Implementation of the HERO is detailed in the presentation at the 42nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC'20) (48). This was a collaborative effort with our group providing education on the concepts of radiation therapy as well as the optimization data to the other group, who performed the optimization process using the HERO. Thus, the specific inner workings of the HERO will not be detailed here as our group was not involved with the technical aspects of the HERO's implementation.

An objective function of the same form as Eq. 2.1 was adapted for use in a QUBO problem in the format of a quadratic pseudo-boolean function:

$$f(x_1, x_2, \dots, x_n) = - \sum_{i < j} W_{ij} x_i x_j - \sum_i b_i x_i \quad (5.2)$$

where the x 's are binary variables (i.e., either 0 or 1), W_{ij} is the strength of the

connection between x_i and x_j , and b_i is the bias term for x_i . The goal of QUBO is to minimize f .

5.2.4 Evaluations

As prostate IMRT plans were unavailable to use in a retrospective comparison, Seven VMAT prostate plan patients, selected at random from the Eclipse database, were replanned using standard RTOG IMRT protocols. The same set of objectives was used for all seven plans, which are shown in Table 5.1. There are two sets of objectives that were used, labeled "Obj1" and "Obj2". "Obj1" is the original set of objectives, which are identical to the objectives used in the work by Nazareth (22). "Obj2" introduces stricter constraints to the upper and lower PTV objectives. The reasoning for this will be explained in the Discussion. Each case had around 6000 PB's, the number of which was determined by CERR to accommodate the PB's with dimensions of 0.25 cm x 0.25 cm.

Each plan was optimized using Eclipse's DVO and the HERO method. Obj1 and Obj2 optimizations were done separately. Eclipse's DVO default termination settings (100 minutes or 1000 iterations, whichever came first) were set. These settings were not needed, though, as each run was visually monitored and terminated manually when no further improvements were made. This time was recorded, and then the optimization was resumed and a little more time allotted to observe if the DVO would be able to make any additional improvements. If it did, that time would be recorded as well. The HERO was set to halt once the score dropped to below 0.01% its initial value or after five minutes, whichever came first. For all evaluations, the HERO terminated due to score reduction.

Table 5.1: A summary of the objectives used for optimizing the prostate IMRT plans using SA, Eclipse's DVO, and the HERO.

Structure	Volume (%)	Obj1 Dose (Gy)	Obj2 Dose (Gy)	Priority Weight
PTV (Rx)	95	66.6	66.6	12
PTV (upper)	0	73.0	69.0	6
PTV (lower)	100	63.3	65.0	6
Bladder	70	40.0	40.0	1
Bladder	50	65.0	65.0	1
Rectum	55	40.0	40.0	1
Rectum	35	65.0	65.0	1
L. Fem. Head	10	50.0	50.0	1
R. Fem. Head	10	50.0	50.0	1

The resulting fluences from both Eclipse and the HERO were imported to Matlab and converted into a vector. This vector was then applied to the influence matrices calculated by CERR to obtain a global dose matrix. Binary masks (similar to the ones described in Chapter 3, but calculated by CERR itself) were multiplied element-by-element with the global dose matrix to obtain dose volume information for each structure.

Upon calculation of the dose matrices in every case, normalization was done such that 95% of the PTV received 66.6 Gy as this was considered to be the prescription. Because of this, the objective score contribution from that objective was zero in every case as the dose was always equal to the objective.

5.3 Results

Table 5.2 contains the dose values for each objective averaged over all seven prostate IMRT cases after optimization by Eclipse and HERO for both sets of objectives. Averaged over all cases, the HERO was able to provide lower doses to the OAR's than the Eclipse optimizer while still providing adequate coverage of the PTV.

Table 5.2: Dose-volume data for each objective averaged over all seven IMRT patients for each optimization approach. For each OAR objective, the average HERO result is lower than the corresponding Eclipse optimizer result, which indicates that, to varying amounts, the HERO produced more favorable OAR doses at the volume specified by the objective.

Structure (Objective)	Eclipse-Obj1	HERO-Obj1	Eclipse-Obj2	HERO-Obj2
PTV (95%, 66.6 Gy)	66.6 ± 0.0	66.6 ± 0.0	66.6 ± 0.0	66.6 ± 0.0
PTV (max, 73.0 Gy)	70.4 ± 0.3	72.7 ± 0.7	70.1 ± 0.4	69.8 ± 0.5
PTV (min, 63.3 Gy)	61.7 ± 0.6	62.7 ± 0.5	63.4 ± 0.7	64.1 ± 0.2
Bladder (70%, 40.0 Gy)	19.6 ± 13.8	17.6 ± 13.9	19.7 ± 13.7	17.9 ± 14.0
Bladder (50%, 65.0 Gy)	34.3 ± 14.5	33.3 ± 15.0	34.3 ± 14.4	33.5 ± 14.9
Rectum (70%, 40.0 Gy)	46.7 ± 16.6	35.5 ± 13.4	46.8 ± 16.6	36.6 ± 14.0
Rectum (50%, 65.0 Gy)	63.3 ± 2.9	60.2 ± 3.2	63.5 ± 2.8	60.4 ± 3.6
L. Fem. Head	33.1 ± 3.3	29.6 ± 3.6	33.0 ± 3.3	29.5 ± 3.5
R. Fem. Head	32.8 ± 2.7	29.3 ± 2.5	32.8 ± 2.7	29.1 ± 2.7

Figures 5.4 to 5.10 compare the DVH's for each case after the optimized weights were applied to the influences calculated by CERR, and the resulting dose matrices were used to obtain dose-volume information for each structure. Both the regular objectives (a) and strict objectives (b) DVH's are included for each case. In all cases the HERO optimization resulted in more favorable OAR DVH curves. That is, the overall dose distribution to the OAR's were reduced while the target coverage was kept the same. For the regular objectives, the HERO PTV DVH curves were less homogeneous than the Eclipse PTV DVH curves. For the strict objectives, the HERO PTV DVH curves were as homogeneous as the Eclipse PTV DVH curves.

Note that the doses were calculated with fluences that did not undergo MLC leaf sequencing. HERO fluence values ranged from 0.0 to 1.75 in steps of 0.25 due to the binary nature of the optimization while Eclipse fluences were in real number values. The HERO's low level of number values may be advantageous during the leaf sequencing step (see Discussion).

Figure 5.11 shows percent dosage (i.e., the overdose) relative to the associ-

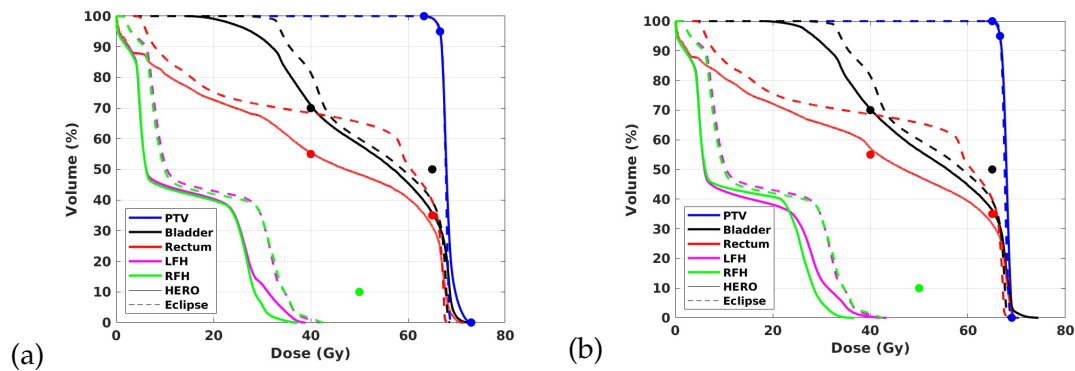


Figure 5.4: DVH's resulting from optimizing with Eclipse's DVO (dashed curves) and the HERO (solid curves) for the first prostate case. The results using the regular objectives (Obj1) are in (a) while the results using the strict objectives are in (b).

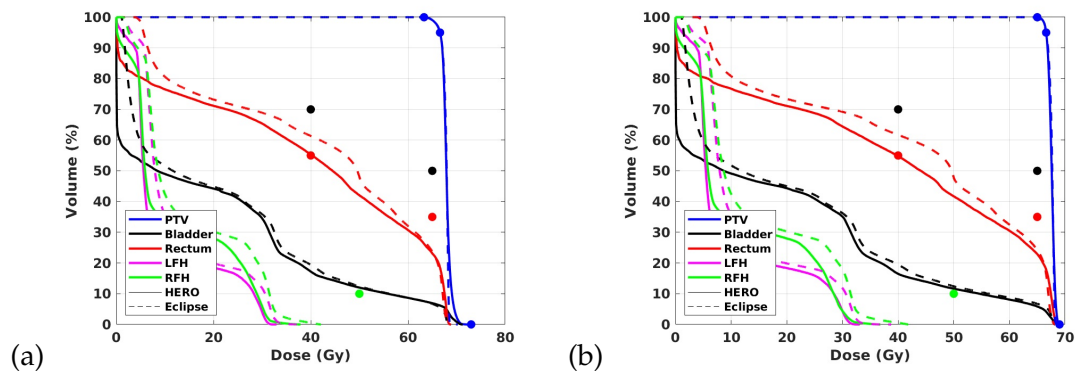


Figure 5.5: DVH's resulting from optimizing with Eclipse's DVO (dashed curves) and the HERO (solid curves) for the second prostate case. The results using the regular objectives (Obj1) are in (a) while the results using the strict objectives are in (b).

ated objective averaged over all seven prostate IMRT cases. Each plot represents a single structure. Excluded from this figure is the PTV 95% dose, the prescription, as each dose array was normalized to it. Red lines indicate the normal set of objectives while the blue dashed lines indicate the stricter objectives. With the exception of the minimum PTV objective, the objective is considered met for points below the red line. The error bars correspond to the standard deviation

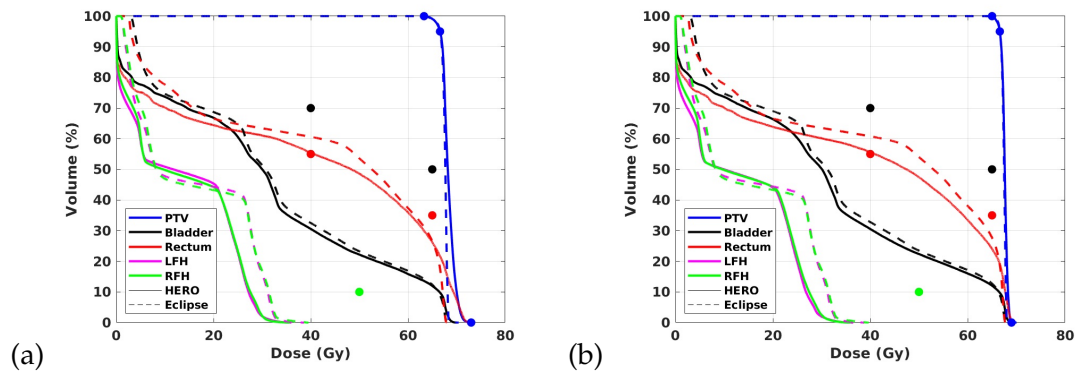


Figure 5.6: DVH's resulting from optimizing with Eclipse's DVO (dashed curves) and the HERO (solid curves) for the third prostate case. The results using the regular objectives (Obj1) are in (a) while the results using the strict objectives are in (b).

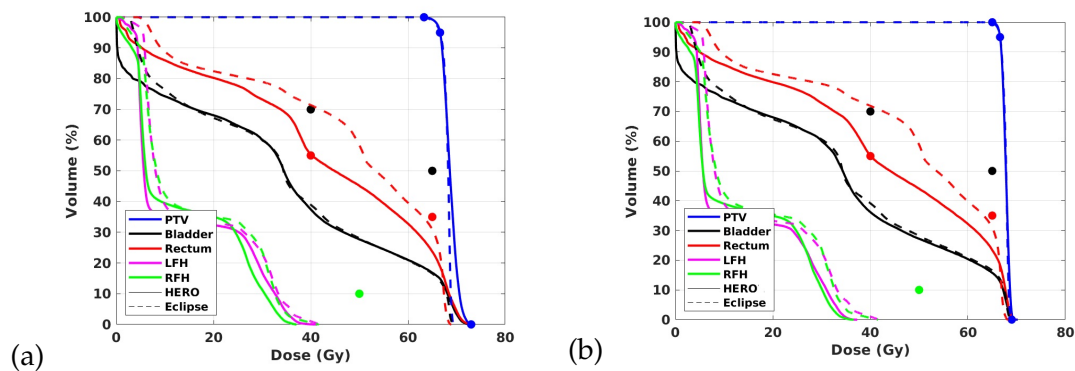


Figure 5.7: DVH's resulting from optimizing with Eclipse's DVO (dashed curves) and the HERO (solid curves) for the fourth prostate case. The results using the regular objectives (Obj1) are in (a) while the results using the strict objectives are in (b).

of the relative overdose over the seven patients.

Figure 5.12 contains log-log plots showing the time to best solution versus the objective score for each prostate IMRT plan optimized using each approach. Both the HERO and Eclipse were able to reach their respective best solutions in under one minute for the normal objectives. For the strict objectives, the HERO took one to two minutes to reach the best solution. The Eclipse optimizer

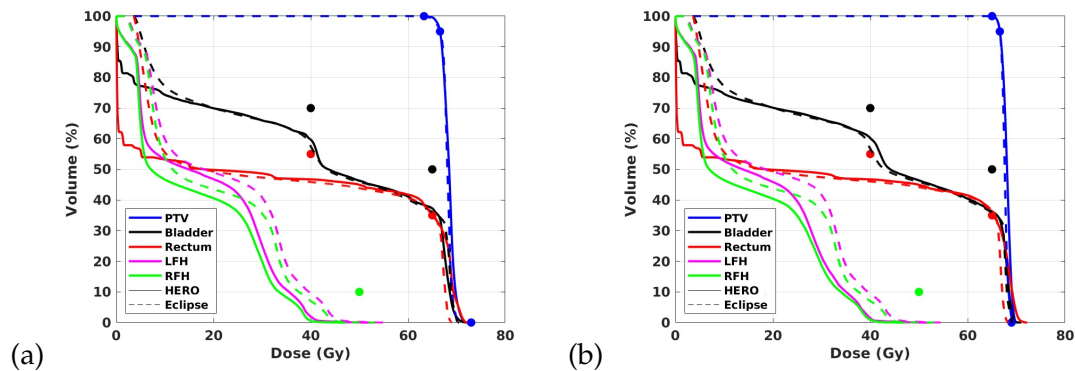


Figure 5.8: DVH's resulting from optimizing with Eclipse's DVO (dashed curves) and the HERO (solid curves) for the fifth prostate case. The results using the regular objectives (Obj1) are in (a) while the results using the strict objectives are in (b).

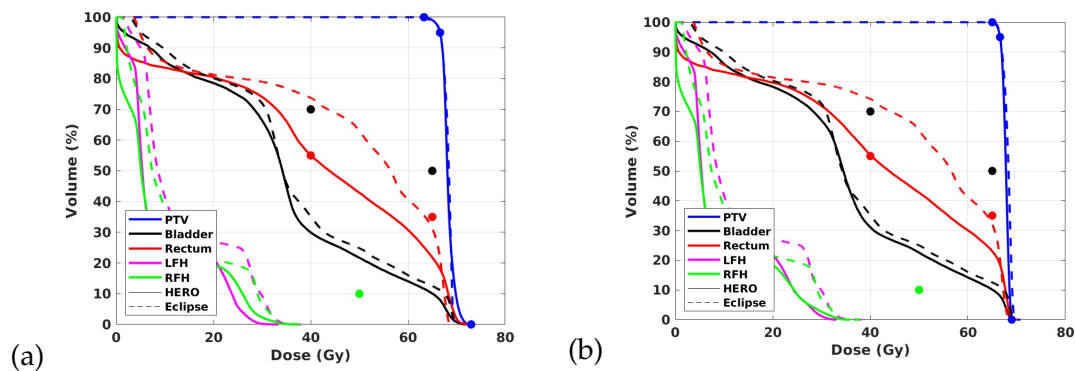


Figure 5.9: DVH's resulting from optimizing with Eclipse's DVO (dashed curves) and the HERO (solid curves) for the sixth prostate case. The results using the regular objectives (Obj1) are in (a) while the results using the strict objectives are in (b).

was allowed additional time, but no significant observable improvements to the score were made in that additional time. With the exception of the fifth case with strict objectives the HERO yielded lower objective function scores than Eclipse.

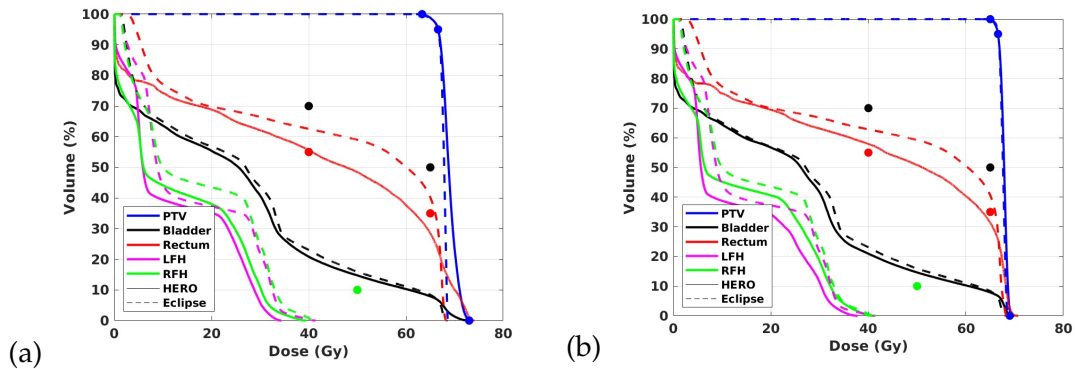


Figure 5.10: DVH's resulting from optimizing with Eclipse's DVO (dashed curves) and the HERO (solid curves) for the seventh prostate case. The results using the regular objectives (Obj1) are in (a) while the results using the strict objectives are in (b).

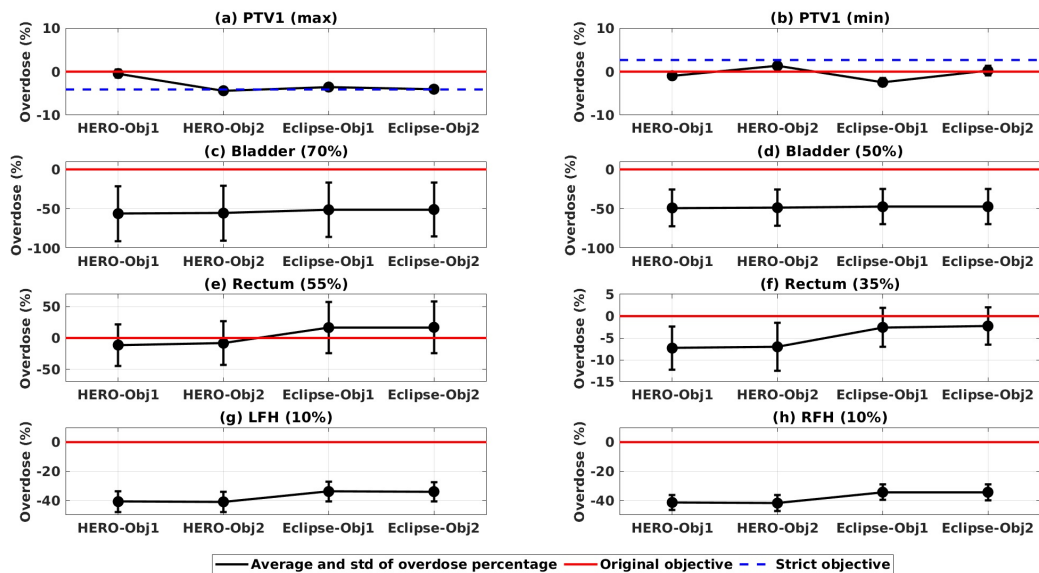


Figure 5.11: Plots comparing the percent overdoses relative to the corresponding objectives averaged over the seven prostate IMRT plans.

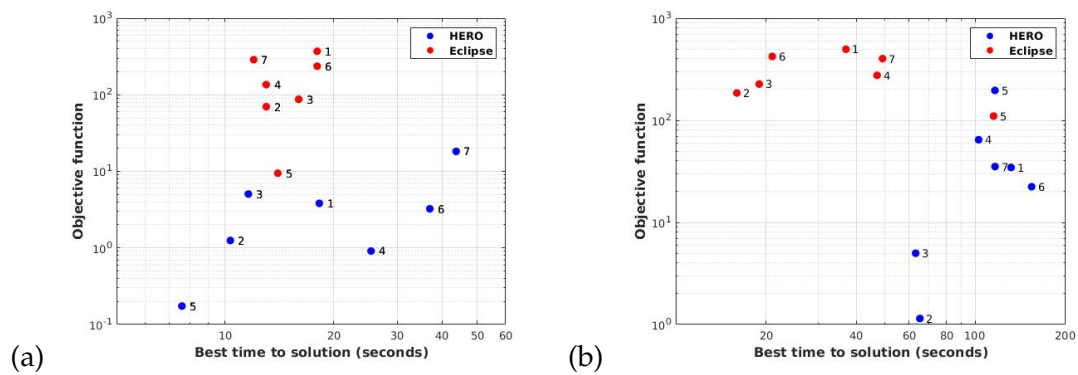


Figure 5.12: Log-log plots showing time to best solution versus objective function score for the HERO and Eclipse optimizations. (a) corresponds to the regular objectives while (b) corresponds to the strict objectives.

5.4 Discussion

5.4.1 Normal and Strict Objectives

A main staple clinicians look for when evaluating a plan based on its DVH is a homogeneous-looking PTV curve. That is, the PTV curve should, ideally, meet the prescription and then fall vertically down with a minimal tail. This indicates that the dose distribution in the PTV is relatively uniform (i.e., homogeneous). It was observed that when using the original objective set, the PTV DVH curves for each of the plans resulting from the HERO optimization had significant tails. It is uncertain why the Eclipse optimization did not result in similar curves, though, it is possible Eclipse's optimizer has a built in hidden function that prioritizes the homogeneity of the PTV DVH curve.

In an attempt to compensate for the PTV DVH curve tails resulting from the HERO optimization, stricter objectives were placed on the minimum and maximum doses. In each case, HERO took one to two minutes longer to find the best solution, but the resulting PTV DVH curves were more homogeneous without sacrificing the shape of the OAR DVH curves. Since clinicians prefer the homogeneous PTV curve, this would make the resulting plan more attractive to them, and they would be more likely to approve its use clinically.

5.4.2 Limitations in the Study

As mentioned before, the fluences compared in this study were ideal fluences. For these fluences to be useful clinically, the treatment planning system would need to perform a sequencing step. That is, the TPS performs an additional optimization step where MLC leaf motion is calculated that take into consider-

ation parameters such as the leaf speed. Leaf sequencing can potentially reduce the quality of a plan; however, since the HERO only provides solutions of PB weights ranging from 0 - 1.75 in steps of 0.25 (i.e., only eight different values), it is possible the fluences will not undergo significant reduction in quality. That is, the ideal dose distribution would not need to be sacrificed when transitioning from an ideal fluence to the actual sequenced fluence that the linac can deliver.

One issue we need to point out is that there was no way to make a direct comparison between the HERO and Eclipse optimization results as Eclipse does not provide PB data, nor does it have the option to apply the ideal fluence to a dose matrix. Because of this, Eclipse fluences had to be calculated using Eclipse's dose calculator, not the CERR PB's. The fluences were then exported from Eclipse, and the CERR PB's were applied to obtain the dose matrices used for this study. It is possible Eclipse's dose volume data could have more tightly fit the constraints than presented in this work. Given access to the PB influence matrices generated by Eclipse, this study could have been a direct comparison as we could have applied those data with the HERO instead of using the CERR PB influence matrices.

5.4.3 Comparison to Quantum Annealer Approach

The HERO showed promising results when optimizing IMRT plans, particularly when compared to similar work done by Nazareth (22) using a Quantum Annealer. Both methods used a QUBO approach, but the HERO was able to handle many more variables, and it does not require expensive hardware maintained at very cold (near absolute zero) temperatures like the Quantum Annealer. The lack of special hardware considerations makes clinical implementa-

tion of the HERO much more likely.

Chapter 6

Conclusions

Hasta la vista, baby.

T-800 - Terminator 2: Judgement

Day (1991)

6.1 A Framework for Parallelized Monte Carlo Simulations

A summary of the user's experience, from anonymized DICOM file export to MC job submission, can be found in Figure 6.1. This process is relatively uncomplicated and spares the user from having to devote a lot of time into learning the MC system and troubleshoot inevitable issues that will arise.

As of writing this dissertation, the development of an automated process of generating virtual phantoms independent of MMCTP is nearing completion. A set of scripts has been developed that produces virtual phantoms that, visually, look virtually identical to those produced by MMCTP. Currently, refinements

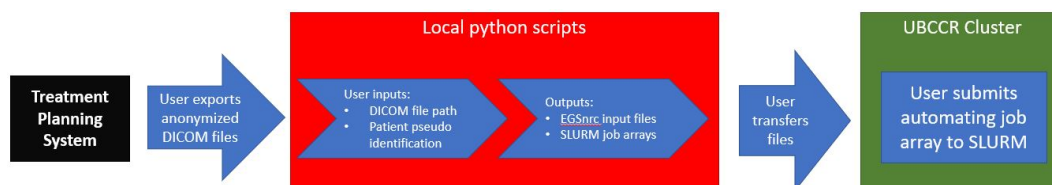


Figure 6.1: A flowchart summarizing the process the user follows, from exporting anonymized DICOM files to job array submission.

are being made to the scripts, but the process itself is, for the most part, functional. Commissioning of this process will be performed very soon.

Additionally, more is being done to make the process of generating EGSnrc input files and SLURM job arrays easier for the user. Graphical user interfaces (GUI's) are being created with the goal that the user will no longer need to do inputs manually in the Python script. This falls in line with more traditional computing as most commercial applications do not require the user to amend the script. Additionally, as executables the GUI's will be self contained, and the user will not be required to have a Python interpreter installed on their workstation.

Lastly, more refinement is being done to increase the automation of EGSnrc job submission. The final objective, which is the culmination of this work, is to have the user direct the GUI to the directory containing DICOM files and make a few inputs and click 'execute'. After that, input files and job arrays will be generated and submitted to the cluster. Once complete, the user can access the results. This will make obtaining MC data for eComp and VMAT plans very simple and user friendly. In addition, in line with Bhagroo's (6) work, a feature will be added that automatically performs a secondary check on dynamic plans, thus providing an easy to use, quick, and clinically viable alternative to other secondary checks (e.g., MU Check).

6.2 Enhanced Optimization of VMAT Plans Using Monte Carlo Generated Beamlets

We have developed *Enhanced Optimization*, a novel method for improving the quality of VMAT plans. The EO process begins with the TPS-produced plan, and applies small perturbations in the form of beamlets. This permits the objective space in the neighborhood of the TPS plan to be explored for locations of lower minima. Since the beamlets are calculated using an MC dose calculation algorithm, the scores computed during the EO search are more accurate than those computed by the TPS optimizer. The resulting plan is then imported into the TPS in order to determine the final, deliverable dose, and to compare the EO and original plans.

We evaluated the EO on seven clinical VMAT plans, and obtained improvements in the brain and head and neck cases. The prostate cases were also improved to a smaller extent. Modifying the DVH objectives resulted in further improvements in certain cases, which may be useful when a critical OAR objective is difficult to meet. The aim of the EO was to achieve relatively small improvements to the clinical plans without the need to replan, the EO may open up a new type of treatment planning, involving a combination of DVH objectives and more sophisticated search algorithms.

The development of the EO is an ongoing process, and much work needs to be done before it is clinically viable. The main issue to address is the time it takes to calculate beamlets, which essentially creates a bottleneck in the entire process. We currently propose running the EO overnight. At this point it is automated enough to do a majority of the beamlet calculations on its own. In the future, though, there are several things we will investigate to reduce the

runtime of the EO.

One of the causes of this bottleneck is the utilization of shared computing resources. A lot of time is spent on overhead; that is, waiting for resources to free up before a job can begin. We are also subject to limitations such as maximum simultaneous running jobs as well as finite project storage space. If our group was allotted dedicated computing resources, we would be able to bypass the restrictions.

Another proposed way of reducing the time it takes to complete the EO process is the development of a more efficient method of beamlet selection. Currently, the selection criteria for beamlet calculation does not account for the impact of individual beamlets on the optimization. It is apparent that most of the beamlets calculated go unused. Developing a way to predict which beamlets contribute more to the optimization process will allow us to reduce the number of beamlets selected for calculation and, thus, reduce the time needed to complete the EO process.

6.3 A Hamiltonian Engine for Radiotherapy Optimization

A novel method of IMRT plan optimization has been developed. This method employs quantum computing inspired hardware; thus, it can perform operations like quantum computing hardware, but it does not require the special considerations such as near-zero cooling. The computing hardware used in this study is able to run with the same considerations as conventional computing hardware.

The HERO involves building a QUBO model of the IMRT optimization problem in order to solve for optimal beamlet weights, which are presented as fluences. By comparing the HERO method to clinically used treatment planning software, Eclipse, we have shown that the HERO method is able to produce higher-quality dose distributions, based on the DVH data, in a very short amount of time.

We are confident this novel platform may be scaled to VMAT optimization in the future, and a preliminary investigation into this is currently underway involving the EO. A main ambition of this collaboration is to develop a system composed of conventional hardware that can very quickly (i.e., within seconds-minutes) perform VMAT optimization. This has a myriad of clinical usefulness, not least of which is adaptive radiotherapy (ART) where a treatment plan is adapted to the current situation while the patient is on the treatment table.

Appendix A

Code and Utilities

Matlab and python scripts related to the projects detailed in this dissertation have been stored in the following repository at github.com:

<https://github.com/mathewsmedphys/dissertation>

Bibliography

- [1] A. F. Bielajew. *Fundamentals of the Monte Carlo Method for Neutral and Charged Particle Transport*. U of Michigan Print, Ann Arbor, MI, 2001.
- [2] I. J. Chetty, B. Curran Curran, J. E. Cygler, J. J. Demarco, G. Ezzell, B. A. Faddegon, I. Kawrakow, P. J. Keall, H. Liu, C. M. Charlie Ma, D. W. O. Rogers, J. Seuntjens, D. Sheikh-Bagheri, and J. V. Siebers. Report of the aapm task group no. 105: Issues associated with clinical implementation of monte carlo-based photon and electron external beam treatment planning. *Med. Phys.* 34 (12), pages 4818 – 4853, 2007.
- [3] A. M. Bergman, K. Bush, M.-P. Milette, I. A. Popescu, K. Otto, and C. Duzenli. Direct aperture optimization for imrt using monte carlo generated beamlets. *Med. Phys.*, 33(10):3666–3679, 2006.
- [4] R. L. Stern, R. Heaton, M. W. Fraser, S. Murty Goddu, T. H. Kirby, K. L. Lam, A. Molineu, and T. C. Zhu. Verification of monitor unit calculations for non-imrt clinical radiotherapy: Report of aapm task group 114. *Med. Phys.*, 38(1):504–530, 2011.
- [5] A. M. Bergman, E. Gete, C. Duzenli, and T. Teke. Monte carlo modeling of hd120 multileaf collimator on varian truebeam linear accelerator for veri-

- fication of 6x and 6x fff vmat sabr treatment plans. *J of App Clin Med Phys*, 15(3):148–163, 2014.
- [6] S. Bhagroo, S. B. French, J. A. Mathews, and D. P. Nazareth. Secondary monitor unit calculations for vmat using parallelized monte carlo simulations. *J of App Clin Med Phys*, 20(6):60,69, 2019-06.
- [7] K. Otto. Volumetric modulated arc therapy: Imrt in a single gantry arc. *Med. Phys.*, 35(1):310 — 317, 2007.
- [8] A Boyer, P Biggs, J Galvin, E Klein, T Losasso, D Low, K Mah, and C Yu. Report of task group no. 50 radiation therapy committee: Basic applications of multileaf collimators. , 2001.
- [9] H. E. Romeijn, R. K. Ahuja, J. F. Dempsey, and A. Kumar. A column generation approach to radiation therapy treatment planning using aperture modulation. *SIAM Journal on Optimization*, 15(3):838–862, 2005.
- [10] About - <https://www.dicomstandard.org/about/>.
- [11] Key concepts - <https://www.dicomstandard.org/concepts/>.
- [12] Pydicom - <https://pydicom.github.io/>.
- [13] Image processing toolbox - <https://www.mathworks.com/products/image.html>.
- [14] T. Bortfeld. Clinically relevant intensity modulation optimization using physical criteria. *International Conference on the Use of Computers in Radiation Therapy*, 1997.
- [15] *Eclipse Photon and Electron Algorithms Reference Guide*. Varian Medical Systems, Inc., 2019.

- [16] R E Drzymala, R Mohan, L Brewster, J Chu, M Goitein, W Harms, and M Urie. Dose-volume histograms. , 21(1):71–78, May 1991.
- [17] F. Jiang, H. Wu, H. Yue, F. Jia, and Y. Zhang. Photon optimizer (po) prevails over progressive resolution optimizer (pro) for vmat planning with or without knowledge-based solution. *J of App Clin Med Phys*, 18(2):9,14, 2017-03-01.
- [18] D. A. Binny, T. Kairn, C. M. Lancaster, J. V. Trapp, and S. B. Crowe. Photon optimizer (po) vs progressive resolution optimizer (pro): a conformality- and complexity-based comparison for intensity-modulated arc therapy plans. *Med. Dosim*, 43(3):267–275, 2018.
- [19] J. L. Bedford. Treatment planning for volumetric modulated arc therapy. *Med. Phys.*, 36(11):5128,5138, 2009-11.
- [20] H. Liu, B. Sintay, K. Pearman, Q. Shang, L. Hayes, J. Maurer, C. Vanderstraeten, and D. Wiant. Comparison of the progressive resolution optimizer and photon optimizer in vmat optimization for stereotactic treatments. *J of App Clin Med Phys*, 19(4):155,162, 2018-07.
- [21] Q. Lyu, D. O' Connor, D. Ruan, V. Yu, D. Nguyen, and K. Sheng. Vmat optimization with dynamic collimator rotation. *Med. Phys.*, 45(6):2399,2410, 2018-06.
- [22] D. P. Nazareth and J. D. Spaans. First application of quantum annealing to imrt beamlet intensity optimization. *Phys. Med. Biol.*, 60(10):4137 – 4148, January 2015.

- [23] D. Craft, D. Mcquaid, J. Wala, W. Chen, E. Salari, and T. Bortfeld. Multicriteria vmat optimization. *Med. Phys.*, 39(2):686,696, 2012-02.
- [24] M. Mahnam, M. Gendreau, N. Lahrichi, and L.-M. Rousseau. Simultaneous delivery time and aperture shape optimization for the volumetric-modulated arc therapy (vmat) treatment planning problem. *Phys. Med. Biol.*, 62(14):5589,5611, 2017-07-21.
- [25] Y. Niu et al. Improving imrt-plan quality with mlc leaf position refinement post plan optimization. *Med. Phys.*, 39(8):5118 — 5126, 2012.
- [26] F. M. Khan and J. P. Gibbons. *Khan's the Physics of Radiation Therapy*, pages 426 — 428. LWW (PE), 5 edition, 2014.
- [27] X. Jia et al. "development of a gpu-based monte carlo dose calculation code for coupled electron-photon transport.". *Med. Phys.*, 55(11):3468, May 2010.
- [28] J. S. Li et al. Validation of a monte carlo dose calculation tool for radiotherapy treatment planning. *Phys. Med. Biol.*, 45:2969 — 2985, June 2000.
- [29] Health information privacy - <https://www.hhs.gov/hipaa/index.html>, Jan 2019.
- [30] D. W. O. Rogers, B. R. Walters, and I. Kawrakow. Beamnrc users manual, 2011.
- [31] S. B. French. Resolution adaptation and secondary optimization of volumetric modulated arc therapy, 2019.
- [32] B. R. Walters, I. Kawrakow, and D. W. O. Rogers. Dosxyznrc users manual, 2011.

- [33] University at buffalo center for computational research - http://www.buffalo.edu/ccr/support/research_facilities/general_compute.html.
- [34] Slurm workload manager - <https://slurm.schedmd.com/overview.html>.
- [35] A. Alexander. Monte carlo treatment planning with modulated electron radiotherapy: Framework development and application, 2012.
- [36] A. Alexander. Mmctp: A radiotherapy research environment for monte carlo and patient-specific treatment planning, 2007.
- [37] A. Alexander, F. DeBlois, G. Stroian, K. Al-Yahya, E. Heath, and J. Seuntjens. Mmctp: A radiotherapy research environment for monte carlo and patient-specific treatment planning. *Physics in Medicine and Biology*, 52(13):N297–N308, 2007.
- [38] A. Alexander, J. Seuntjens, and F. DeBlois. Mmctp - <https://andrewwalexander.github.io/mmctp/>.
- [39] Andrew Alexander. McGill monte carlo treatment planning users manual.
- [40] D. W. O. Rogers, I. Kawrakow, J. P. Seuntjens, B. R. B. Walters, and E. Mainegra-Hing. Nrc user codes for egsnrc, Jan 2020.
- [41] J. Lawhead. Point in polygon - <http://geospatialpython.com/2011/01/point-in-polygon.html>, Jan 2011.
- [42] J. O. Deasy and A. P. Apte. A computational environment for radiological research features.

- [43] `scipy.spatial.delaunay.find_simplex`, https://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.spatial.delaunay.find_simplex.html, 2008.
- [44] X. Zhang, H. Liu, X. Wang, L. Dong, Q. Wu, and R. Mohan. Speed and convergence properties of gradient algorithms for optimization of imrt. *Med. Phys.*, 31(5):1141–1152, 2004.
- [45] T. Bortfeld. Optimized planning using physical objectives and constraints. In *Seminars in Radiation Oncology*, volume 9, pages 20–34. Elsevier, 1999.
- [46] Anders Ahnesjö, Mikael Saxner, and Avo Trepp. A pencil beam model for photon dose calculation. *Med. Phys.*, 19(2):263–273, 1992.
- [47] E Kalinin and J Deasy. A method for fast 3-d imrt dose calculations: the quadrant infinite beam (qib) algorithm. *Med. Phys.*, 30(6):1348–1349, 2003.
- [48] M. Malekmohammadi, K. Dabiri, J. Mathews, D. P. Nazareth, H. Tamura, and A. Sheikholeslami. A hamiltonian engine for radiotherapy optimization, conference: 2020 42nd annual international conference of the iee engineering in medicine and biology society (embc) in conjunction with the 43rd annual conference of the canadian medical and biological engineering society.

Vita

Joshua Allen Mathews

Joshua "J" Mathews grew up in the Pacific Northwest, in the United States. He began his medical physics education at Western Washington University, where he got his B.S. in Physics. Concordently, J worked for the Northwest Medical Physics Center as an intern where he had the opportunity to experience the daily responsibilities as a medical physicist. J received his M.S. from the Oregon Medical Physics program and his Ph.D. from the University at Buffalo. During his time at U.B., J worked as a medical physicist assistant where he obtained more firsthand clinical experience. J will be continuing his education as a medical physics resident at the Ohio State University.

Colophon

The design of this dissertation was based on the LaTeX template by Jalil Modares and Nicholas P. Nicoletti, which was adapted from a Penn State University template. Additional commands were added as needed. The text was typeset in Computer Modern Roman 12pt. Figures were drawn using Microsoft Powerpoint. This work was printed in a 8.5" x 11" (letter) format and was bound by ProQuest LLC, Publishers, Ann Arbor, Michigan, USA.