

***SERA:
Simulation Environment For
Radiotherapy Applications
User's Manual
Version 1CO***

*D. E. Wessol
C. A. Wemple
F. J. Wheeler
G. J. Harkin
M. W. Frandsen
C. L. Albright
M. T. Cohen
M. B. Rossmeier
J. J. Cogliati*

June 2002

*Idaho National Engineering and Environmental Laboratory
Bechtel BWXT Idaho, LLC*



**SERA:
Simulation Environment for Radiotherapy Applications
User's Manual
Version 1CO**

**D. E. Wessol, C. A. Wemple, F. J. Wheeler,
G. J. Harkin, M. W. Frandsen, C. L. Albright,
M. T. Cohen, M. B. Rossmeier, J. J. Cogliati**

**Edited by
J. R. Venhuizen**

Published June 2002

**Idaho National Engineering and Environmental Laboratory
Bechtel BWXT Idaho, LLC
Idaho Falls, Idaho 83415**

**Prepared for the
US Department of Energy
Office of Science
Under DOE Idaho Operations Office
Contract DE-AC07-99ID13727**

ABSTRACT

This document is the user manual for the Simulation Environment for Radiotherapy Applications (SERA) software program developed for boron-neutron capture therapy (BNCT) patient treatment planning by researchers at the Idaho National Engineering and Environmental Laboratory (INEEL) and students and faculty at Montana State University (MSU) Computer Science Department. This manual corresponds to the final release of the program, Version 1C0, developed to run under the RedHat Linux Operating System (version 7.2 or newer) or the Solaris™ Operating System (version 2.6 or newer). SERA is a suite of command line or interactively launched software modules, including graphical, geometric reconstruction, and execution interface modules for developing BNCT treatment plans. The program allows the user to develop geometric models of the patient as derived from Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) images, perform dose computation for these geometric models, and display the computed doses on overlays of the original images as three dimensional representations. This manual provides a guide to the practical use of SERA, but is not an exhaustive treatment of each feature of the code.

IN MEMORIAM

Floyd J. Wheeler passed away before the final version of the software and manual were completed. Floyd was a leader and innovator in the development of computational methods for neutron transport in tissue, a widely respected member of the international Neutron Capture Therapy community, a great teacher and mentor, a dedicated and loving father, and one of the most decent, forthright, and helpful people anyone could hope to meet. His absence impoverishes us all.

ACKNOWLEDGEMENTS

As with many large endeavors such as this, there is a supporting cast of thousands lurking behind the scenes who are not only unrewarded but, worse, unrecognized. In many of these cases, these same people may and do make contributions more significant than do the authors appearing on that work. This effort is no exception. With regard to this effort, there is one such person that stands above all others, authors included, and that person is Merle Griebenow. Merle instituted the boron neutron capture therapy (BNCT) program at Idaho National Engineering and Environmental Laboratory (INEEL), and perhaps revived it in many other quarters. Merle, like Professor Hatanaka before him, was regarded as a heretic by many of his peers, except by those who really know him. We might say that being regarded as a heretic is perhaps really an honor, since, from the historic perspective, it's probably heresy, not necessity, that is the mother of invention.

Merle has an incredible capacity to observe and learn and then trust and support the people he recruits. For that, we are most grateful.

After Merle, we are deeply indebted to Dave Nigg, Jim Venhuizen, and Peggy Shiffer of the INEEL BNCT program office, and Denbigh Starkey, Computer Science Department Head at Montana State University, who had the unenviable and difficult position of leading interference for us, while we had the less difficult task of writing software. To the list of supporters above, we would certainly be remiss if we didn't give special recognition to Pat Gavin, Ron Dorn, Lowell Miller, Arlene Ackermann, Carol Atkinson and Greg Storr. Also, a great deal of thanks is due to the true victims of this software, Drs. A. Z. Diaz and Ruimei Ma of Brookhaven National Laboratory and Jacek Capala of Studsvik Medical in Sweden. Through their diligent and expert use of this software, Drs. Capala, Diaz, and Ma provided us with many insightful suggestions that ultimately added significant improvement to this software. Without the support of some outstanding students at Montana State University, including Lynn Voss, John Evans and Jeremy Cook, much of this work would not have been possible.

Select images shown as examples in this manual are provided courtesy of the Visible Human Project of the National Library of Medicine.

We would like to dedicate this effort to all BNCT candidates, patients, and their families.

INSTALLATION INSTRUCTIONS

If you are running Solaris 2.6 and newer just make sure you have C and FORTRAN compilers, the Motif libraries, and the GL libraries installed.

If you are running Redhat linux 9 or Fedora Core 2 or 3, if you have installed it with the Workstation option, then everything you need to install sera should be already installed. The packages needed to install are the gcc and g77 compilers, the OpenMotif development libraries, and the OpenGL development libraries.

With both, systems, sera1/fullbuild script can be used to compile sera. When this script is done, it will copy the binaries to the directory sera1/Target/bin

Sera has several environmental variables that need to be defined and some Xresource files that need to be merged. A csh script to do this is located in sera1/Resources/Shared/serarc and a bash script to do this is located in sera1/Resources/Shared/serarc.sh One of these will be needed to be added to your .cshrc or .bashrc script, or they will need to be sourced each time before running sera programs.

Bash sourcing example: source .serash.sh

Csh sourcing example: source .serarc

If you are using the install script, the only thing you need to do manually is adding the line to source the shell script (i.e. .bashrc or .cshrc). Note that the install script will overwrite .serarc (or .serarc.sh) and the sera1 directory.

To run the install script: ./install

The install script has been tested on Redhat 8 and 9, and Fedora Core 2 and 3. It has been tested on Solaris 2.7. Other operating systems have not been tested, so additional work may be required to install Sera.

Troubleshooting

If the compiling fails, then (assuming that the fix is not obvious) the first thing to do is to track down which part of the compiling is failing. The fullbuild script has the commands that are run to compile sera. Try running each command manually and see which command first fails.

If sera3d fails and nothing else does then the problem is most likely with openGL. Note that it requires the Motif openGL widget, GLwMDrawA, which is currently built in the sera1/Shared/lib/libGLwM directory.

If the imake, make Makefile or make Makefiles commands fail, then the problem is probably with imake. The imake configuration is stored in the sera1/config directory.

If compiler flags need to be changed, or other changes need to be made, this can be done in the sera1/config/Local.def file and other files in the config directory.

Files Included

seralsrc.tar.gz	The source code for Sera
seralcvr.tar.gz	The CVS development history for Sera
verification.tar.gz	Test cases for seraMC
install	The install script
README	This file

CONTENTS

ABSTRACT	iii
IN MEMORIAM	iv
ACKNOWLEDGEMENTS	v
INSTALLATION INSTRUCTIONS	vi
1. INTRODUCTION	1
2. CONCEPTS AND NOTATIONS	2
2.1 Window Object	2
2.2 Input	4
3. GETTING STARTED	5
3.1 Main Menu	7
3.2 Image Formatter	8
3.3 Image Editor	9
3.4 Three-Dimensional Viewer	9
3.5 Dose Contours	10
3.6 Dose Plots	12
3.6.1 Dose-Depth Plotting	12
3.6.2 Dose-Volume Histogram Plotting	13
3.7 seraMC Interface	13
3.8 Dose Combination	14
3.9 Help	15
4. IMAGE FORMATS	16
4.1 Image Formatting	16
4.2 File Menu	17
4.2.1 Load QSH Selection	18
4.2.2 Load Single Images Selection	18
4.2.3 Unload Images Selection	20
4.2.4 Generate Raw Images Selection	20
4.2.5 Save QSH Selection	20
4.2.6 Launch Selection	20
4.2.7 Check Version Selection	21
4.3 Options Menu	21
4.3.1 Edit QHD Selection	21
4.3.2 Tool Bar Selection	22
4.3.3 Messages Selection	22
4.3.4 Reverse Images Selection	22
4.3.5 Manipulate Images Selection	22
4.4 Preferences Menu	22
5. IMAGE EDITOR	23
5.1 File Menu	24

5.1.1.	Open Selection	24
5.1.2.	Save Selection	24
5.1.3.	Launch Selection	24
5.1.4.	Remove Bodies Selection	24
5.1.5.	Restart Program Selection.....	25
5.1.6.	Exit Program Selection	25
5.2.	Edit Menu	25
5.2.1.	Undo	25
5.2.2.	Relabel Region Selection.....	25
5.2.3.	Remove Bodies from Image Range.....	26
5.2.4.	Color Map Selection	26
5.2.5.	Constraint Marker Selection	27
5.2.6.	Fiducial Marker Selection.....	27
5.2.7.	Sort Images by Z-value Selection.....	28
5.2.8.	Reverse Image Set Selection.....	28
5.2.9.	Slice Orientation.....	28
5.3.	View Menu	28
5.3.1.	Properties Toolbar Selection.....	29
5.3.2.	Edit Regions Selection.....	29
5.3.2.1.	Setup Mode.....	30
5.3.2.2.	Threshold Mode.....	30
5.3.2.3.	Manual Draw Mode	34
5.3.2.4.	Fill Mode.....	35
5.3.2.5.	3D Grow Mode.....	35
5.3.2.6.	Copy Body Mode.....	36
5.3.2.7.	Make Margin Mode	37
5.3.2.8.	Wand Mode	38
5.3.3.	Context Help Selection	39
5.3.4.	Bottom Information Selection.....	39
5.3.5.	Cross-Section Tool Selection.....	39
5.3.6.	Intensity Histogram Selection.....	40
5.3.7.	Region Volumes Selection.....	40
5.4	Preferences.....	41
5.4.1.	Preferences Menu	41
5.4.2.	Autosave Option.....	42
5.5.	Suggested Modeling Steps for BNCT Treatment Planning.....	42
5.5.1.	Suggested Modeling Steps Using CT Image Sets.....	43
5.5.2.	Suggested Modeling Steps Using MRI Image Sets.....	43
6.	THREE-DIMENSIONAL RECONSTRUCTION	44
6.1.	The Menubar.....	44
6.1.1.	The File Menu	44
6.1.2.	The Options Menu.....	45
6.1.3.	The Preferences Menu	46
6.1.4.	The Help Menu.....	46
6.2.	The Message Bar	46
6.3.	The Main Rendering Window	46
6.4.	The Tools.....	46
6.4.1.	Viewing Panel	48

6.4.2.	Color Panel.....	48
6.4.3.	Transparency Panel.....	49
6.4.4.	Lighting Panel.....	50
6.4.5.	Axes Panel.....	50
6.4.6.	Clipping Panel.....	50
6.4.7.	Polygon Panel.....	52
6.4.8.	Particle Tracking Panel.....	53
6.4.9.	Slice Panel.....	53
6.4.10.	Contour Panel.....	54
6.4.11.	Mouse Panel.....	55
6.4.12.	View Parameters Panel.....	55
6.4.13.	Beam Panel.....	56
6.4.14.	Texture Mapping Panel.....	57
6.5.	The Preferences.....	58
6.5.1.	Window Sizes.....	59
6.5.2.	Color.....	59
6.5.3.	Axes.....	59
6.5.4.	Mouse.....	59
6.5.5.	Particle Paths.....	60
6.5.6.	Texture.....	60
6.5.7.	Miscellaneous.....	60
7.	TRANSPORT MODELING.....	62
7.1.	Introduction.....	62
7.2.	Menus.....	62
7.2.1.	Files Menu.....	62
7.2.2.	Run Menu.....	63
7.2.3.	Options Menu.....	64
7.3.	Run Title Panel.....	65
7.4.	Files Panel.....	65
7.4.1.	Save Directory.....	66
7.4.2.	CG Geom File Name.....	66
7.4.3.	Patient Geom File Name.....	66
7.4.4.	Old RST File Name.....	67
7.4.5.	New RST File Name.....	67
7.4.6.	Source File Name.....	67
7.4.7.	Material File Name.....	67
7.4.8.	Cross Section File Name.....	67
7.4.9.	seraMC Run Script.....	67
7.4.10.	Ultrafast Mode Files.....	67
7.5.	Control Panel.....	68
7.5.1.	Nbatch.....	68
7.5.2.	Nhist.....	68
7.5.3.	Run Directives.....	68
7.5.4.	Run Date.....	68
7.6.	Positioning Panel.....	69
7.6.1.	Target Point Location.....	69
7.6.2.	Distance to Target Point.....	69
7.6.3.	Polar Angle.....	69

7.6.4.	Azimuthal angle	69
7.7.	Tally Panel	69
7.7.1.	Origin	69
7.7.2.	Delw	70
7.7.3.	Wncut	70
7.8.	Auxilliary Panels	70
7.8.1.	Beam Optimization and Automatic Positioning	70
7.8.2.	Edit Isotopes	72
7.9.	Edit Directives	72
7.10.	Monitor Panel	74
8.	STATISTICAL DOSE COMBINATION	75
8.1.	Introduction	75
8.2.	Menus	76
8.2.1.	File Menu	76
8.2.2.	Edit Menu	77
8.2.3.	Preferences Menu	78
8.2.3.1.	Fields/Fractions Preferences	78
8.2.3.2.	Reference Dose Preferences	79
8.2.3.3.	Dose-volume Edit Options Preferences	79
8.2.3.4.	Additional Dose-volume Edits Preferences	80
8.2.4.	Help Menu	80
8.3.	Combination Utility	81
8.3.1.	Input data	81
8.3.1.1.	Identification Parameters	81
8.3.1.1.1.	Patient ID	81
8.3.1.1.2.	Treatment Date	81
8.3.1.2.	Field/Fraction Specifications	81
8.3.1.2.1.	Number of Fractions	81
8.3.1.2.2.	Number of Fields per Fraction	81
8.3.1.2.3.	Set Button	82
8.3.1.3.	Field Data	82
8.3.1.3.1.	Dose File Name	82
8.3.1.3.2.	Boron Concentration Data	82
8.3.1.3.3.	Exposure Data	82
8.3.1.3.4.	Gamma Repair Data	82
8.3.1.3.5.	Active Field	83
8.3.1.3.6.	Same Fields Button	83
8.3.2.	Summation Data	83
8.3.2.1.	Fraction Sum	83
8.3.2.2.	Plan Sum	83
8.3.3.	Output Files	83
8.3.3.1.	Plan File	83
8.3.3.2.	Combined Dose Files	84
8.4.	Edit Module	84
8.4.1.	Basic Input Data	84
8.4.1.1.	Dose File	84
8.4.1.2.	Plan Name	84

8.4.1.3.	Patient ID.....	85
8.4.1.4.	Blood B10.....	85
8.4.1.5.	N_avg.....	85
8.4.1.6.	Upper DV bin (%).....	85
8.4.1.7.	Nbin_DV.....	85
8.4.1.8.	Reference Dose Search Component.....	85
8.4.1.9.	Reference Dose Search Type.....	86
8.4.1.10.	Reference Dose RBE Values.....	87
8.4.1.11.	Perform Standard Edits Button.....	87
8.4.2.	User-defined Edits Input Data.....	87
8.4.2.1.	Point.....	87
8.4.2.2.	Line.....	88
8.4.2.3.	Dose-volume Histogram.....	88
8.4.2.4.	Contour.....	89
8.4.3.	Standard Edits.....	89
8.4.4.	Output Files.....	90
8.4.4.1.	Reference Dose.....	90
8.4.4.2.	Point Edits.....	90
8.4.4.3.	Line Edits.....	90
8.4.4.4.	Dose-volume Histogram Edits.....	90
8.4.4.5.	Contour Edits.....	91
8.4.4.5.1.	Standard Edits.....	91
8.4.4.5.2.	User-defined Edits.....	91
9.	DOSE CONTOURING.....	92
9.1.	File Menu.....	93
9.2.	Color Menu.....	94
9.3.	Options Menu.....	95
9.4.	Help Menu.....	96
9.5.	Loading Images.....	96
9.6.	Loading Contours.....	101
9.7.	Working with Contours.....	103
9.8.	Loading Masks.....	104
9.9.	Working with Masks.....	106
9.10.	Working with Colors.....	107
9.11.	Preferences.....	110
9.12.	Entering Dose Factors.....	113
9.13.	Selecting Dose Components.....	114
9.14.	Changing Field of View.....	115
9.15.	The Image Contouring Window.....	115
10.	2D DOSE PLOTTING TOOLS.....	117
10.1.	Introduction.....	117
10.2.	Simplot.....	117
10.3.	DVall.....	118
10.4.	Patient Positioning.....	120
11.	SYSTEM REQUIREMENTS.....	122

12. TROUBLESHOOTING.....	123
13. REFERENCES	124
APPENDIX A. TRANSPORT CALCULATION MODULE (seraMC).....	125
A.1. Introduction.....	125
A.2. General Problem Input.....	126
A.2.1. Line 1 - Title.....	126
A.2.2. Line 2 - Nbatch, nhist, wncut	126
A.2.3. Line 3 - Target Point.....	126
A.2.4. Line 4 - Zb, phi, theta.....	126
A.2.5. Line 5 - Combinatorial Geometry (CG) File.....	127
A.2.6. Line 6 - Univel File.....	128
A.2.7. Line 7 - Nedit2 and nnegp.....	128
A.2.8. Line 8 - Neutron energy group boundaries	128
A.2.9. Line 9 - X0, Y0, Z0	129
A.2.10. Line 10 - Delw.....	129
A.2.11. Line 11 - Old restart file.....	129
A.2.12. Line 12 - New restart file	130
A.2.13. Line 13 - Source file	130
A.2.14. Line 14 - Irand_num and ied.....	130
A.2.15. Line 15 - Run_dir and run_date.....	130
A.2.16. Line 16 - Material file.....	131
A.2.17. Line 17 - Cross section file	133
A.2.18. Line 18 - Edit isotopes	133
A.2.19. Line 19 - Run script for seraMC.....	134
A.2.20. Line 20 - Proton range file	135
A.2.21. Line 21 - Ultrafast directory file.....	135
A.2.22. Line 22 - Ultrafast cross section file.....	135
A.3. Coordinate Transformations	135
A.4. Automatic Beam Locator.....	136
A.4.1. Positioning when iop=1	137
A.4.2. Positioning when iop=2	137
A.4.3. Positioning when iop=3	138
A.4.4. Positioning when iop=4	138
A.4.5. Positioning when iop=5	138
A.5. Edit Directives.....	138
APPENDIX B. seraMC COMBINATORIAL GEOMETRY	140
B.1. Introduction.....	140
B.2. Body Input Specifications.....	140
B.2.1. Rectangular Parallelepiped (RPP)	141
B.2.2. Box (BOX)	141
B.2.3. Sphere (SPH).....	142
B.2.4. Right Circular Cylinder (RCC).....	142
B.2.5. Right Elliptical Cylinder (REC)	143
B.2.6. Truncated Right Angle Cone (TRC).....	143
B.2.7. Ellipsoid (ELL).....	144
B.2.8. Right Angle Wedge (WED)	144

B.2.9.	Arbitrary Polyhedron of 4, 5, or 6 sides (ARB)	145
B.2.10.	Right Vertical Wedge (RVW)	146
B.2.11.	Restricted General Ellipsoid (RGE)	146
B.3.	Combinatorial Geometry Descriptions	147
B.4.	Examples of CG Descriptions	147
B.4.1.	Example 1	147
B.4.2.	Example 2	148
B.4.3.	Example 3	149
B.4.4.	Example 4	149
B.4.5.	Example 5	150
B.5.	Format of Combinatorial Geometry File	151
APPENDIX C.	seraMC SOURCE DESCRIPTION	152
C.1.	Introduction	152
C.2.	Formats	152
C.3.	Variable Descriptions	152
C.3.1.	Nsorks	152
C.3.2.	Nmu	152
C.3.3.	Xtnsn and Xtnsg	152
C.3.4.	Dim	154
C.3.5.	Fspec	154
C.3.6.	Lreg, Mat, and Reg	154
APPENDIX D.	seraMC EDIT DIRECTIVES	155
D.1.	Introduction	155
D.2.	Passive Directives	155
D.2.1.	patient	155
D.2.2.	date	155
D.2.3.	beam	156
D.3.	Reference Dose Parameters	156
D.3.1.	refvol	156
D.3.2.	refpoint	156
D.3.3.	refdepth	156
D.3.4.	ref_reg and more_ref	156
D.3.5.	ref_b10	157
D.3.6.	ref_rbe	157
D.4.	General Parameters	157
D.4.1.	mw_min	157
D.4.2.	mw	157
D.4.3.	b10_blood	158
D.4.4.	b10_ratio	158
D.4.5.	delta	158
D.4.6.	eps	158
D.4.7.	in_reg and more_reg	158
D.4.8.	nbin_DV	159
D.4.9.	rbe	159
D.4.10.	images	159
D.4.11.	ap, bp, and cp	159
D.4.12.	skinentry	160

D.5.	Active Edits.....	160
D.5.1.	point	160
D.5.2.	line	160
D.5.3.	px, py, and pz	160
D.5.4.	contour	161
D.5.5.	ottocon.....	161
D.5.6.	box	161
D.5.7.	DVbs	162
D.5.8.	fiducial	162
D.5.9.	ras.....	162
D.5.10.	beamplt.....	162
D.5.11.	N_avg.....	163
D.5.12.	crosshair	163
APPENDIX E.	seraConvert	164
E.1.	Invoke seraConvert.....	164
E.2.	Load QSH and Spline Files	164
E.3.	Making Sure all Bodies Are Known.....	164
E.4.	Build the Univel Files	165
E.5.	Varifying Results	165
E.6.	Viewing Univel Files in SeraModel	165
APPENDIX F.	dicom2qsh	166
F.1.	Introduction.....	166
F.2.	Installing dicom2qsh	
F.3.	Converting DICOM to QSH.....	166

SERA

Simulation Environment for Radiotherapy Applications

User's Manual Version 1C0

1. Introduction

SERA is a suite of command line or interactively launched software modules, including graphical, geometric reconstruction, and execution interface modules, for developing the Boron Neutron Capture Therapy (BNCT) radiation treatment plans. In typical use, SERA consists of three main components:

- Manual and semi-automated geometric modeling of treatment objects (brain, tumor, eyes, sinus, etc.) derived from MRI, CT, and other medical imaging modalities
- Dose computation for these geometric models with seraMC, the INEEL radiation transport computer code
- Dose contouring that overlays dose contours derived from the dose computation onto the original image planes to determine healthy tissue limits.

Digitized data produced from medical imaging devices are used as a template to draw, fill, or paint regions of interest in each image plane, forming a solid model description of the region of interest. These solid model descriptions are composed of volumetric automatons, which will be referred to as uniform volume elements (univels) [1]. Because of the uniformity of the univels, geometric interrogation can be performed using a very efficient integer arithmetic algorithm. Also, the resolution of the univels is only limited by the imaging modality, since each univel can be as small as an individual image pixel.

Using the solid model descriptions of the regions of interest, a three-dimensional Monte Carlo radiation transport model, seraMC [2], developed at INEEL, is used to calculate the complex radiation fields present in BNCT treatment. These radiation fields are then related to radiation doses delivered to the treatment volume and surrounding areas.

When the planning calculations are complete, the radiation fields are displayed in the original image space as dose contours. This gives the clinician or researcher an accurate and convenient way to view the radiation fields, so that the optimal beam location and treatment time can be quickly determined, balancing tumor cell kill against healthy tissue sparing.

The remainder of this manual provides a guide to the practical use of SERA, but is not an exhaustive treatment of each feature of the code.

2. Concepts and Notations

The basic concepts used by SERA are slices, bodies, and body-slices, as illustrated in Figure 2.1. A slice is data corresponding to a single medical image. Typically, several anatomical features of interest will intersect a slice. For example, a slice can contain a cranial object, tumor object, and objects defining eyes, sinuses, and other relevant features.

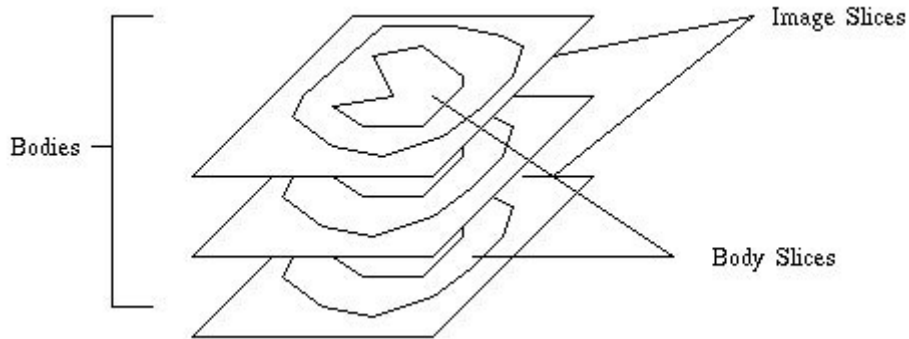


Figure 2.1: Basic SERA Concepts

Bodies correspond to anatomical features from the image (such as the skull, brain, eyes). Body-slices are intersections of bodies (physical features) with slices (a medical image). They are defined by drawing, filling, or painting the body on the slice. Once defined, body definitions in different slices are combined to form three-dimensional volumes.

Typically, the user begins by displaying an image (slice), and then defining the objects on that image that represent body-slices (see Section 5). Body-slices can then be copied to new slices, and modified to correspond with the shapes in each new slice.

After several body-slices have been defined, they may be reconstructed (or synthesized) into a body representing an anatomical feature. These reconstructions can be displayed in a three-dimensional viewing window, where viewing controls allow arbitrary views of the reconstructed data.

2.1. Window Objects

When the program is running, there are a number of different types of window objects that appear on the screen. The graphics interface is based on the X11 window environment [3], the Motif interface toolkit [4], and the OpenGL three-dimensional graphics system [5].

As shown in Figure 2.2, there are several components in a windows environment.

- Buttons - the beginning of all operations in SERA. The main SERA menu consists of a single column or row of buttons. Buttons may also be the terminus of a menu selection as well as the means of canceling a selection. To "press" a button, move the mouse cursor into the objects area and press a mouse button. In some cases, any mouse button will do; in others, different mouse buttons have different effects.
- Menus - sets of buttons representing the choice available to the user. Menus may be arranged either horizontally or vertically.
- Pulldown Menus - submenus that appear when a menu button is selected. These menus exist only until a selection is made or a mouse button is pressed outside of the menu.

- Popup Menus - transient menus that appear unattached to any other menu and disappear after a specific action is taken.
- Input/Output Windows - for error messages, feedback, and any typed entries that are required from the user. The user may enter strings in the input regions of these windows by positioning the cursor in the region and depressing the left mouse button that activates the region. After entering the appropriate text with the keyboard, the user will either press the ENTER key, or select the appropriate button, such as "OK".
- Graphics Windows - windows containing graphical representations of the data being studied. There are two kinds: image windows, which contain a medical image slice and in which objects identifying important features are drawn and filled or painted, and viewing windows, in which 2D and 3D representations of the reconstructed geometries are displayed.

Throughout this manual, the terms window and widget will be used interchangeably, to the chagrin of the purists. A window is a framed view of the virtual computer display, while a widget is a combined set of window objects, such as a viewing window and several buttons.

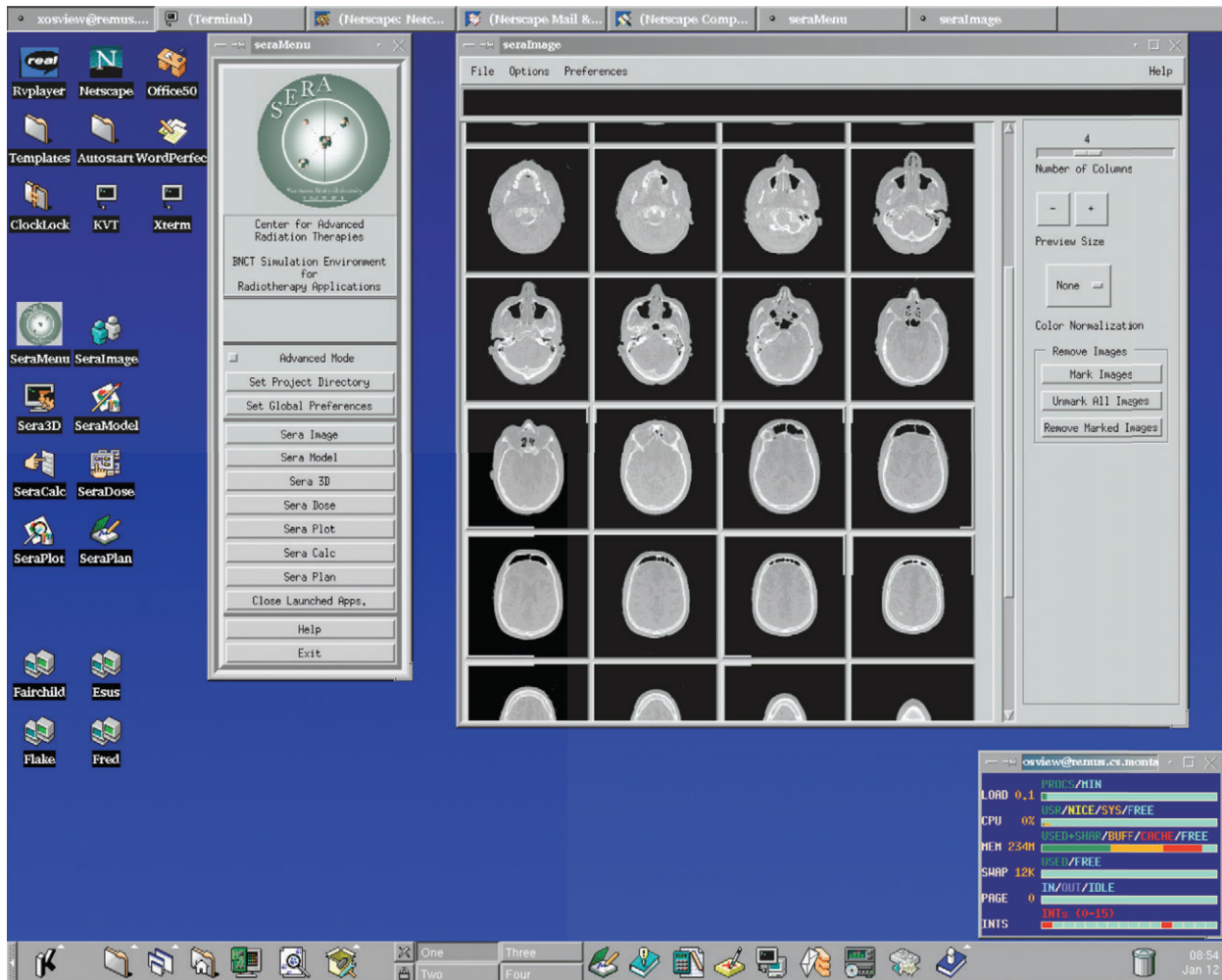


Figure 2.2: Typical X11 Environment.

2.2. Input

SERA input devices are the keyboard, which is used for textual input, and the mouse, which is used for both pointing and selection by maneuvering the mouse pointer about the screen and clicking the mouse button to indicate desired actions. The meaning of the mouse clicks and key strokes depends upon their context.

In a menu, the left mouse button selects an item from the menu to indicate what operation is to be performed. At other times, the keyboard can be used to generate strings in the active window. The program prompts for the type of input expected.

3. Getting Started

To operate properly, SERA relies on the existence of several ancillary files in specified locations within the user's UNIX file system. These files control the appearance of the various user interfaces within the SERA system, provide physical property information, colormaps, and contextual help, and store user-specific preferences. There is one file, usually in the user's home directory, called `serarc`, that provides the information the operating system needs in order to determine the location of these ancillary files. In UNIX parlance, this file is "sourced" when the user login shell is started. If this procedure is unfamiliar to you, check with your local UNIX or X-Windows expert. An example of this file follows:

```
# .serarc: sets up the environment variables for SERA. There are
# several ways to make this file take effect. See your system
# administrator for the best way.
setenv SERA_HOME $HOME/sera1
# SERA_RESOURCES locates resource information
setenv SERA_RESOURCES $SERA_HOME/Resources
# set location for X11 resource files
setenv XUSERFILESEARCHPATH $SERA_RESOURCES/app-defaults/%N
# set documentation paths
setenv SERA_HELP $SERA_HOME/Docs/dialog_help
setenv FILES_MANUAL_PATH $SERA_HOME/Docs/Manuals/sera1/
setenv NET_MANUAL_PATH http://www.cs.montana.edu/~bnct/manual/
#seraMC resources
setenv SERAMC SeraMC
setenv SERAMC_PATH $SERA_HOME/$SERAMC
```

By default, all of the SERA executable binaries are installed in `$SERA_HOME/Target/bin`, and this directory path should be in the user's "path" environment variable. The default installation directory structure under the user's home directory should appear similar to what is shown in Figure 3.1. Most of the SERA distributions will reflect this directory structure and for most cases the user can place the gzipped tar file included in the distribution compact disk (CD) (e.g., `sera1C0linux.tar.gz`) in the appropriate directory, usually the user's home directory, and unzip and extract the tar archive (e.g., `tar xvzf sera1C0linux.tar.gz`.)

There should be a README file included with your software distribution that will detail this information, but if you are not familiar with UNIX and X11, try to get help from your local UNIX system administrator.

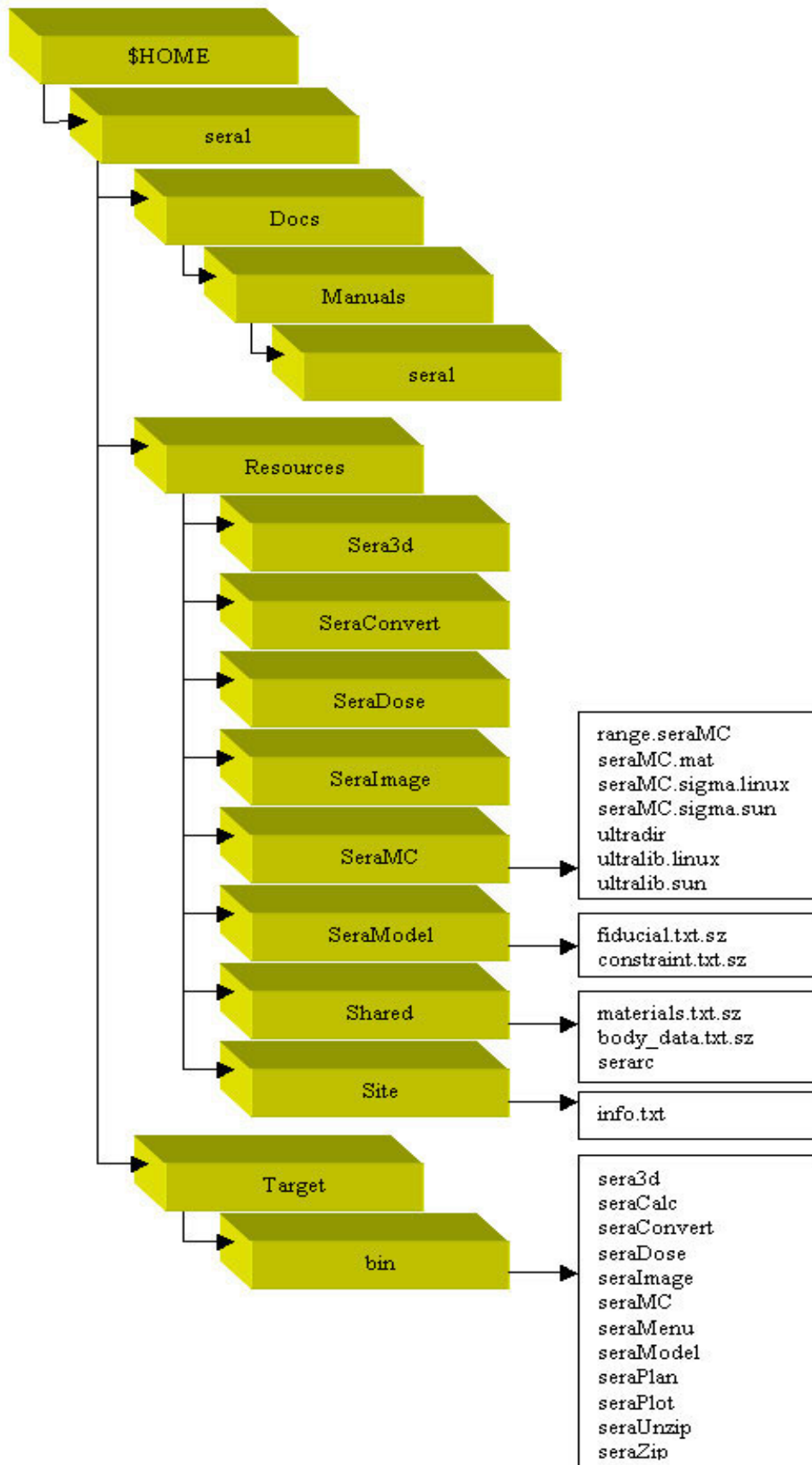


Figure 3.1. Default UNIX directory structure for the SERA system.

3.1. Main Menu (seraMenu)

Once SERA is properly installed, the user issues the command `seraMenu` from the command line, and the main menu, as shown in Figure 3.2, should appear. If the response is `Command not found`, the UNIX path variable may not be properly set. If the program runs, but the menu appearance is strange (i.e. poor color choices, odd fonts, or inappropriate font sizes), it is likely your X11 server has not found the SERA resource file.



Figure 3.2. SERA Main Menu (seraMenu).

Buttons are provided to execute the five main functions within SERA, which include the image formatter, image editor, 3D viewer, dose display utilities, and dose calculation utilities. These functions/utilities may also be executed from the operating system prompt (command line), but the main menu accommodates certain advanced features, such as changing displays, monitoring

the processes initiated from the main menu, setting global preferences, and other features. Each function can execute the other functions (cross-launch capability), so the ability to monitor the various processes can be a necessity. The seven main modules of SERA will be briefly discussed in turn in the remainder of this section.

3.2. Image Formatter (seraImage)

Most of the treatment plans developed with SERA will begin with seraImage. Its basic function is to convert the original image format into the QSH format, which is the internal format used within the SERA modules. The image formatter is discussed at length in Section 4. Shown in Figure 3.3 is the image formatter's user interface with the File pulldown selected.



Figure 3.3. Image Formatter user interface (seraImage).

The image formatting function will accept unformatted (raw) and dicom images. It will also allow the user to edit images already in the QSH format. Images may be deleted, image header information may be changed, and the images may be translated or rotated.

3.3. Image Editor (seraModel)

After the image information has been prepared, image editing with the seraModel module is usually the next step in the treatment planning process. This is discussed at length in Section 5. The user interface for seraModel is shown in Figure 3.4, with the File pulldown selected.



Figure 3.4. Image Editor user interface (seraModel).

The purpose of the seraModel module is to rapidly break an image set into regions of interest. The image matrix used for display in this program has been generalized to work on systems with different color depths, and to view an arbitrary number of columns of images at an arbitrary zoom level in an arbitrary window size.

The image editor provides many useful image operations, including manual and automatic definition and generation of univel-based regions of interest, which form the geometry used by the Monte Carlo radiation transport simulation. Various image processing tools, including region copying, scaling, overwriting, and painting by fill or borders, are provided to aid in the manual definition of these regions. Threshold-based segmenting, 3D region growing, body copying, and automatic target definition operations are also provided.

3.4. Three-Dimensional Viewer (sera3d)

After all of the bodies are created with seraModel, the three dimensional viewer, sera3d, which is detailed in Section 6, provides flexible three-dimensional displays of the univel-based solid

models (see Figure 3.5) and isodose contour data. Points/lines, solid volume elements, or hollow volume elements can be used to view the geometry.

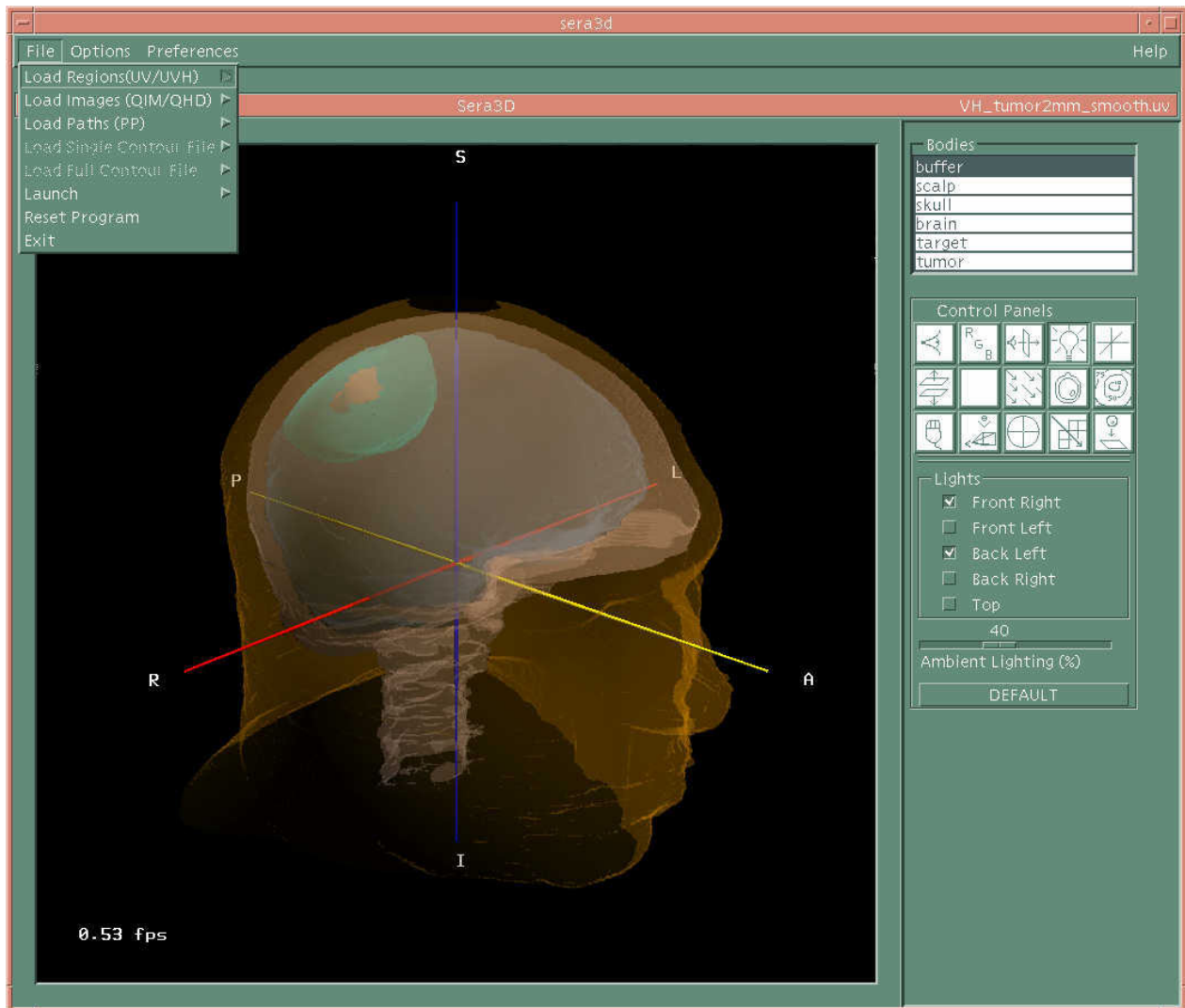


Figure 3.5. Three-dimensional viewer user interface (sera3d).

Two clipping planes are provided in each of the three orthogonal directions. The images used to define the regions in the image editing step may be inlaid into the viewing window via the texture map. The beam line and selected particles may also be displayed in the viewing window. A unique feature for viewing two- and three-dimensional isodose contours is also provided.

3.5. Dose Contours (seraDose)

The SERA dose contouring utility, seraDose, can be invoked either from the Sera Dose button in the main menu, or directly from the command line by entering seraDose. The two-dimensional dose display module contours, registers, and displays two-dimensional isodose curves, superimposed over the original image set. It will also display user-selectable dose components, and it permits dose component editing.

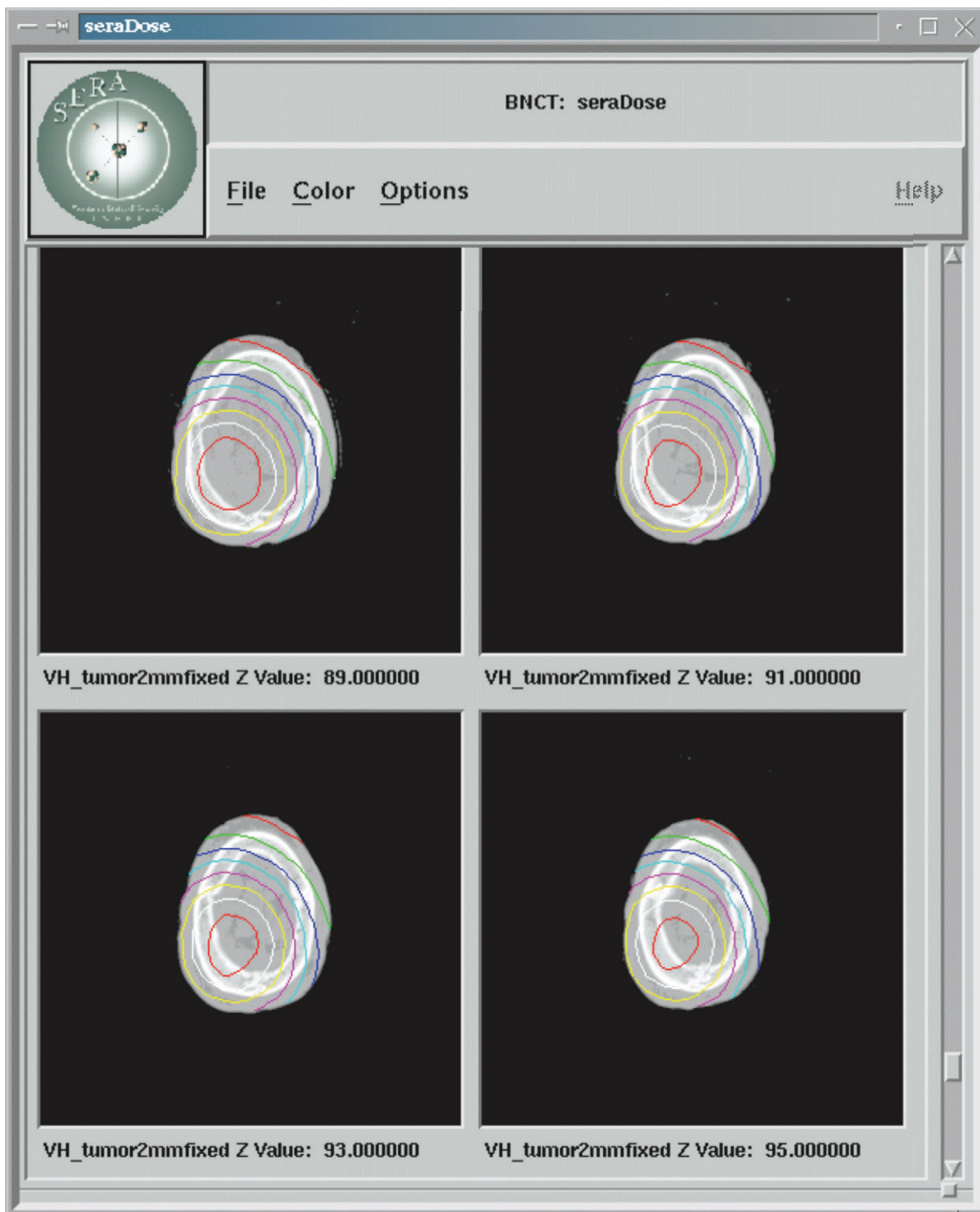


Figure 3.6. Two-dimensional dose contour user interface (seraDose).

The seraDose module also allows user-specified contour levels, and it will report contour values at the current cursor location. Isodose contours can be represented either as lines or color-washed regions. A unique dose masking feature is provided to remove unwanted dose curves in specific regions. Moreover, seraDose will accept either QSH formatted or raw 256x256 pixel images. This is further described in Section 9.

3.6. Dose Plots (seraPlot)

The Sera Plot selection from the main menu provides integrated control of dose-depth and dose-volume histogram plotting utilities (Figure 3.7), which post-process the results of the treatment simulation and display the results as dose-volume histograms or depth-dose plots. This module is further described in Section 10.

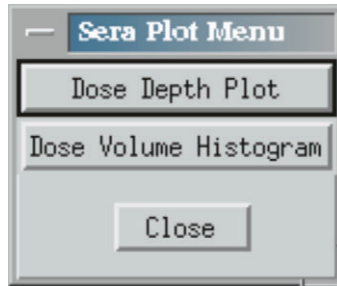


Figure 3.7. Sera Plot pop-up widget (seraPlot).

3.6.1. Dose-Depth Plotting

The dose-depth plotting utility reads the seraMC standard output and invokes the xmgr generalized two-dimensional plotting code for each encounter of a seraMC line edit. The utility can be run from the Dose Depth Plot button in the seraPlot pop-up widget (see Figure 3.8), which is triggered from the main menu's Sera Plot button. The seraPlot pop-up widget can also be brought up from the command line, by entering seraPlot at the operating system prompt.

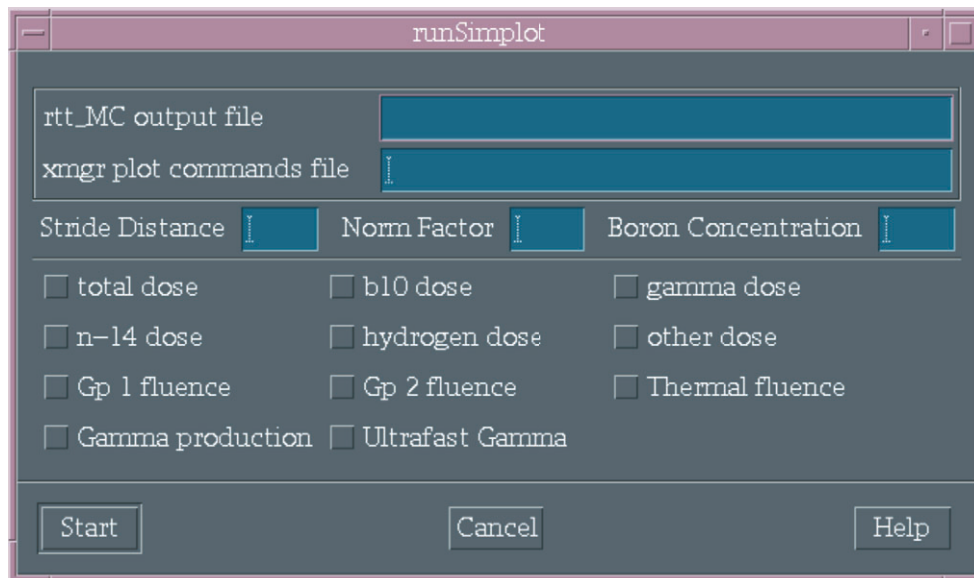


Figure 3.8. Dose Depth plotting widget in seraPlot.

Dose depth plots can be shown for either or all of the following dose components:

- total dose
- b-10 dose
- gamma dose
- n-14 dose
- hydrogen (proton recoil) dose

- other dose
- Group 1 fluence
- Group 2 fluence
- Thermal fluence
- Gamma production
- Ultrafast Gamma.

More information can be found in Section 10.

3.6.2. Dose-Volume Histogram Plotting

The dose-volume histogram plotting utility reads the seraMC standard output and invokes the xmgr plot code for each encounter of a seraMC dose-volume edit. The utility can be run from the Dose Volume Histogram button in the seraPlot pop-up widget (see Figure 3.9), which is triggered from the main menu's Sera Plot button. The seraPlot pop-up widget can also be brought up from the command line, by entering seraPlot at the operating system prompt.

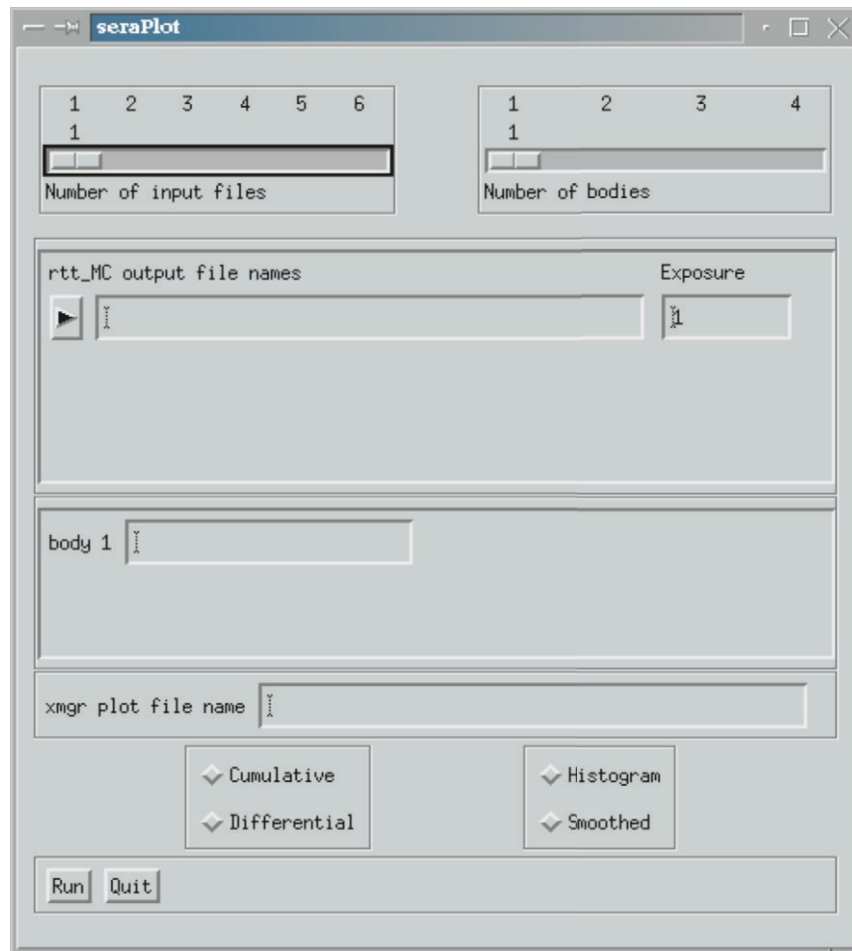


Figure 3.9. Dose-Volume Histogram plotting widget in seraPlot.

3.7. seraMC Interface (seraCalc)

The seraMC interface for transport modeling can be invoked either from the Sera Calc button of the main menu, or directly from the command line by entering seraCalc. The seraMC interface is shown in Figure 3.10. The user may build a new seraMC input file from scratch or open an existing input file, which can be modified as desired to produce an updated input file. The input

can then be run from this interface. The components of this interface are explained in Section 7. The detailed input description of the components appears in the seraMC user's manual.

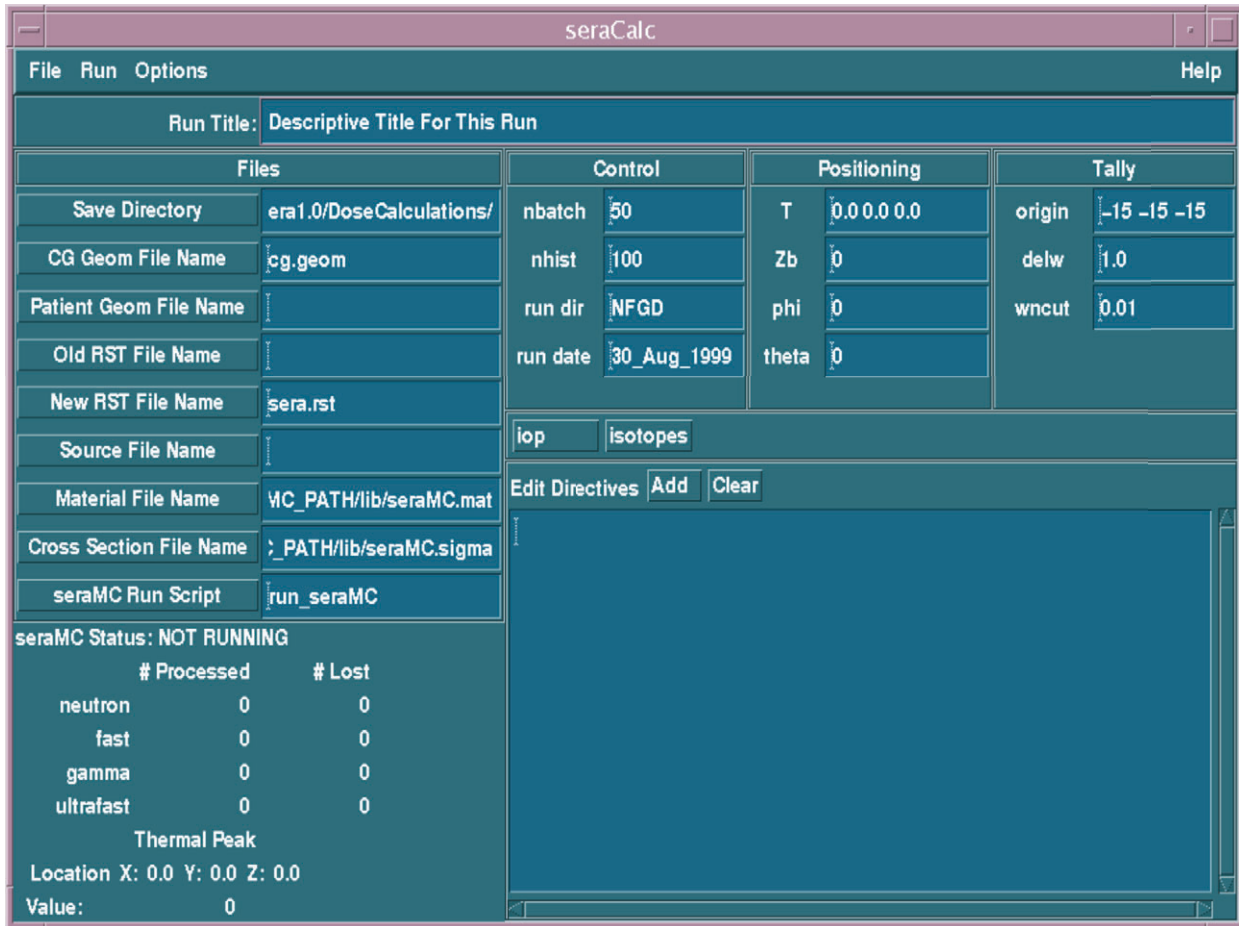


Figure 3.10. The seraMC input interface (seraCalc).

3.8. Dose Combination (seraPlan)

The seraPlan module allows the user to statistically combine fields and/or fractions from several independent seraMC calculations, so that single effective doses can be presented. SeraPlan can be invoked either from the seraPlan button in the main menu, or directly from the command line by entering seraPlan. When seraPlan starts, the window in Figure 3.11 will be displayed. The user may select between 1 and 4 fractions and between 1 and 6 fields per fraction. This is discussed further in Section 8.

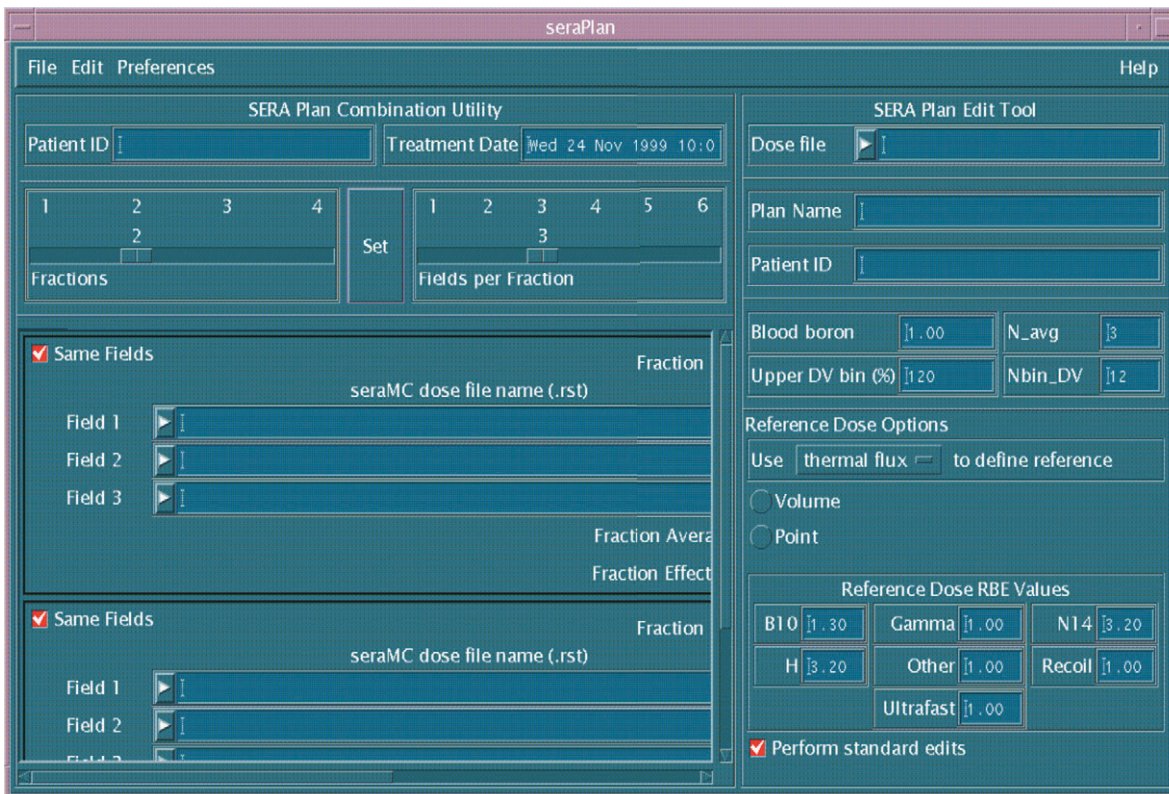


Figure 3.11. The dose combination widget (seraPlan).

3.9. Help

Selecting the Help button changes the user's cursor to a question mark (?) character. The user can then select any button on the main menu. After the user selects a button on the main menu, a help prompt widget will appear, similar to Figure 3.12. The help prompt widget gives the user a choice of selecting Dialog Help, a context sensitive help text, Internet Manual, which displays an appropriate section of the manual in the user's web browser, or the Files Manual, which is similar to the Internet Manual selection, except that the source files for the manual are taken from the user's local file system. In most cases, the Internet Manual selection should be the most recent source of information.

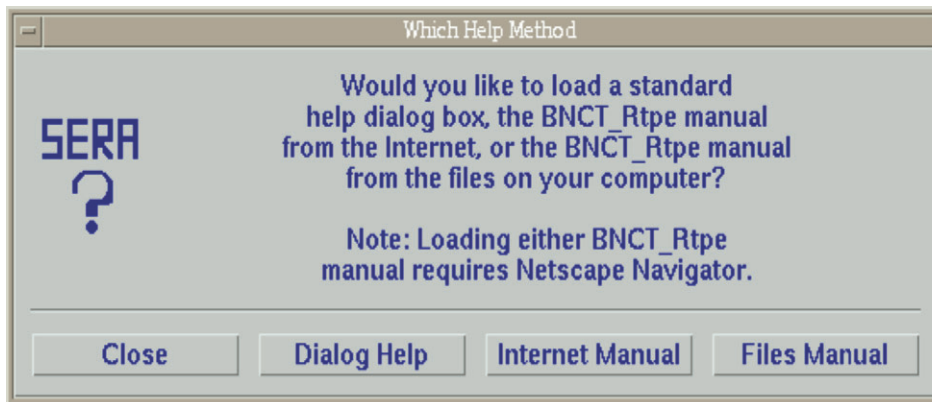


Figure 3.12. Help Prompt widget.

4. Image Formats and Preparation

In order to process images, SERA must know the image format. At this time, SERA accepts either 8/16-bit 256x256 row-major image data or QSH formatted images [6,7,8].

The QSH file format was developed by Maguire and Noz to provide a common image format and incorporate data generated by image-processing into a picture archiving and communication system (PACS). The image file format is an adaptation of the AAPM standard image format and consists of an ASCII header file in the form of standard key-value pairs for its descriptive data and a second file composed of the N-dimensional array of pixel values. The header file might include the following information:

- date of image creation.
- imaging modality
- field of view
- bytes per pixel
- All of the parameters necessary to convert the N-dimensional array of pixel values to the visual representation for interpretation.

Since the image series consists of two separate files, a naming convention was adopted such that both files have the same root filename with different extensions or suffixes. The extension .qhd denotes the ASCII header file, and the extension .qim is used for the actual image or pixel data. Both of these files must be present to access any QSH-formatted image.

SERA has utilities to convert a variety of image formats, including the DICOM format, into QSH. Most of these utilities are incorporated into the `seraImage` module, while the DICOM format conversion is available in the `dicom2qsh` module. The `seraImage` module also provides a convenient mechanism for editing existing QSH header files. The capability also exists to read multiple vendor imaging modality data and present it in the QSH format.

4.1. Image Formatting (`seraImage`)

Most of the treatment plans developed with SERA will begin with `seraImage`. Its basic function is to convert the original image format into the QSH format, which is the internal format used among the SERA modules. The image formatter can be brought up by selecting the Sera Image button from the main menu (see Section 3.1). Alternatively, entering `seraImage` at the command line prompt will also bring up the image formatter's user interface, shown in Figure 4.1, with the File pulldown selected.

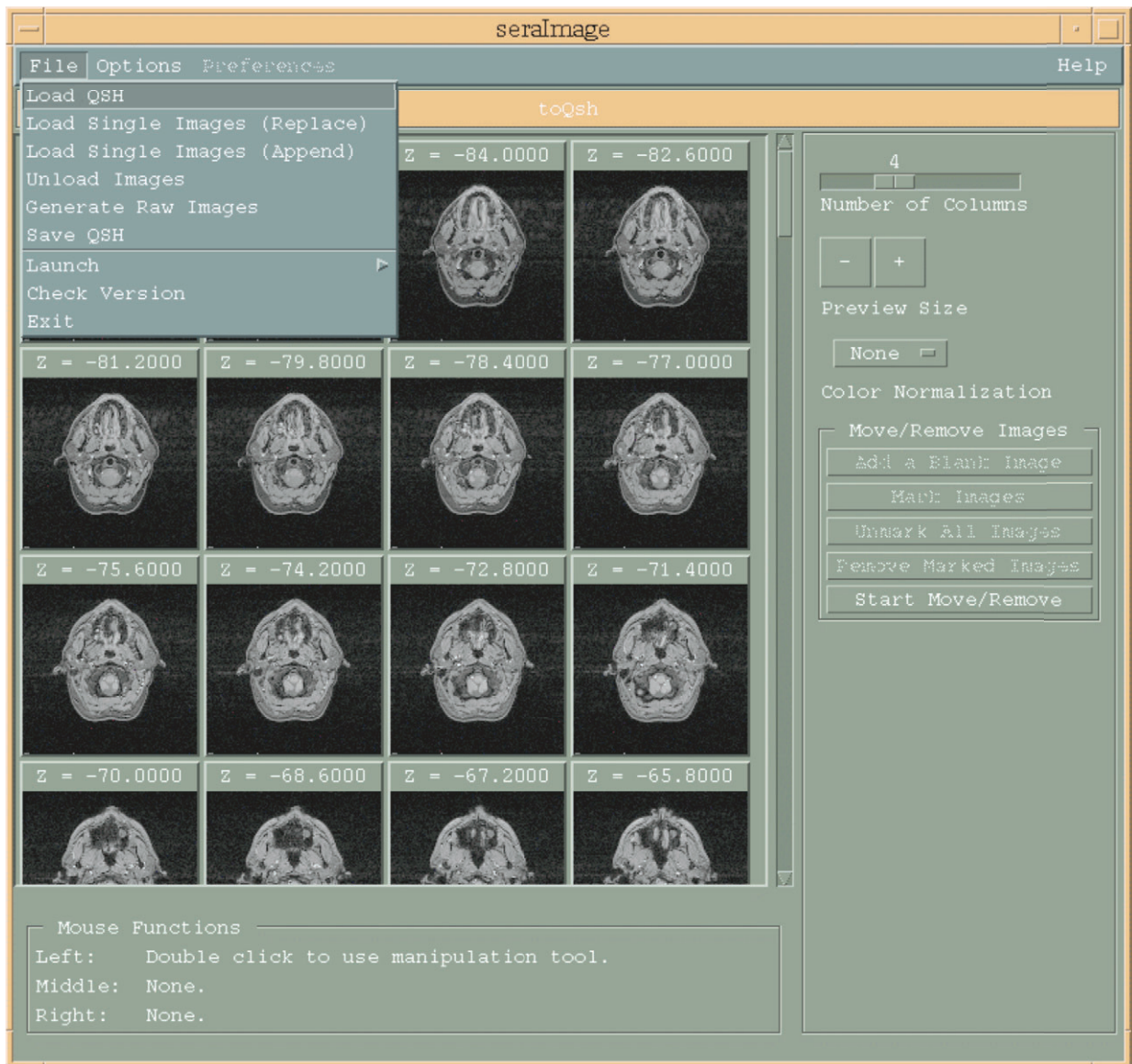


Figure 4.1: seralImage with File Button Selected.

4.2. File Menu

The File menu has the following parts, which will be discussed in turn in the following subsections:

- Load QSH - Loads an existing QSH-formatted file for editing purposes.
- Load Single Images - Reads individual raw images. Allows the user to guess at image format.
- Unload Images - Clears the widget of current image set, clearing the way for a new set.
- Generate Raw Images - Converts current QSH-formatted to a set of individual raw images.
- Save QSH - Saves current set of images as QSH files.
- Launch - Allows the user to run other sera modules.
- Check Version - Checks the version of the current module against the most recent version.
- Exit - Closes current session.

4.2.1. Load QSH Selection

Selecting Load QSH brings up the widget found in Figure 4.2. The widget allows viewing and editing of the required QSH keys.

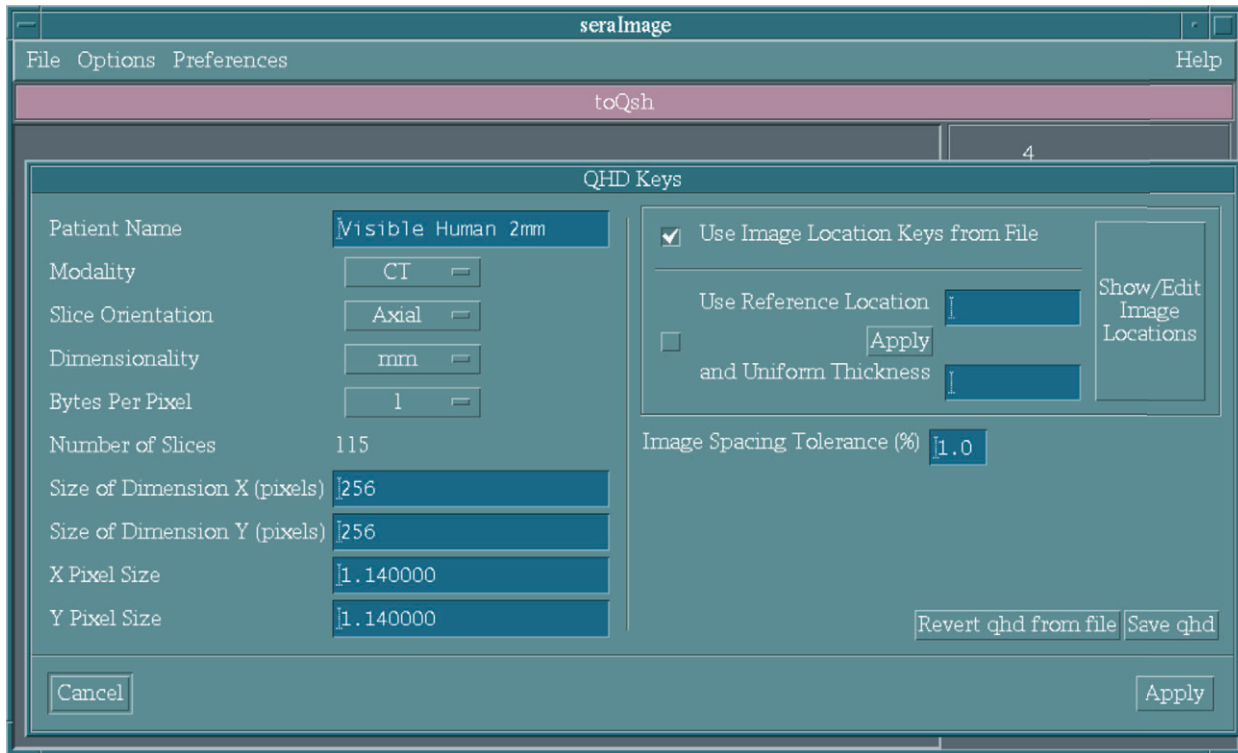


Figure 4.2. Load QSH selection widget.

In addition, the user can specify any of the required QSH file header information which seraImage was unable to recognize from the original image file, so that if required, a new QSH file containing this corrected information can be created. It is very important, especially in the case of univel-based geometries, that the image locations be properly set, and that the image-to-image distances be uniform. This can be checked with the Show/Edit Image Locations button in the right hand side of the widget. If necessary, the image location values can be edited, either one image at a time or by specifying a reference location for the first image and a uniform spacing between slices. As with any editing feature in the SERA system, this must be done with extreme care, as it could destroy the fidelity of the model geometry.

4.2.2. Load Single Images Selection

Raw image files, each containing a single one- or two-byte image, can be viewed, edited, and converted to the QSH format with this option. In most cases, the user will have a group of raw image files in a single directory in the file system, such as depicted in the Select File widget shown Figure 4.3. The user can select the image files in the directory one at a time, all at once, grouped by a wildcard character, or as a grouping created by dragging the cursor through files listed in the second column of the widget under the Files subheading.

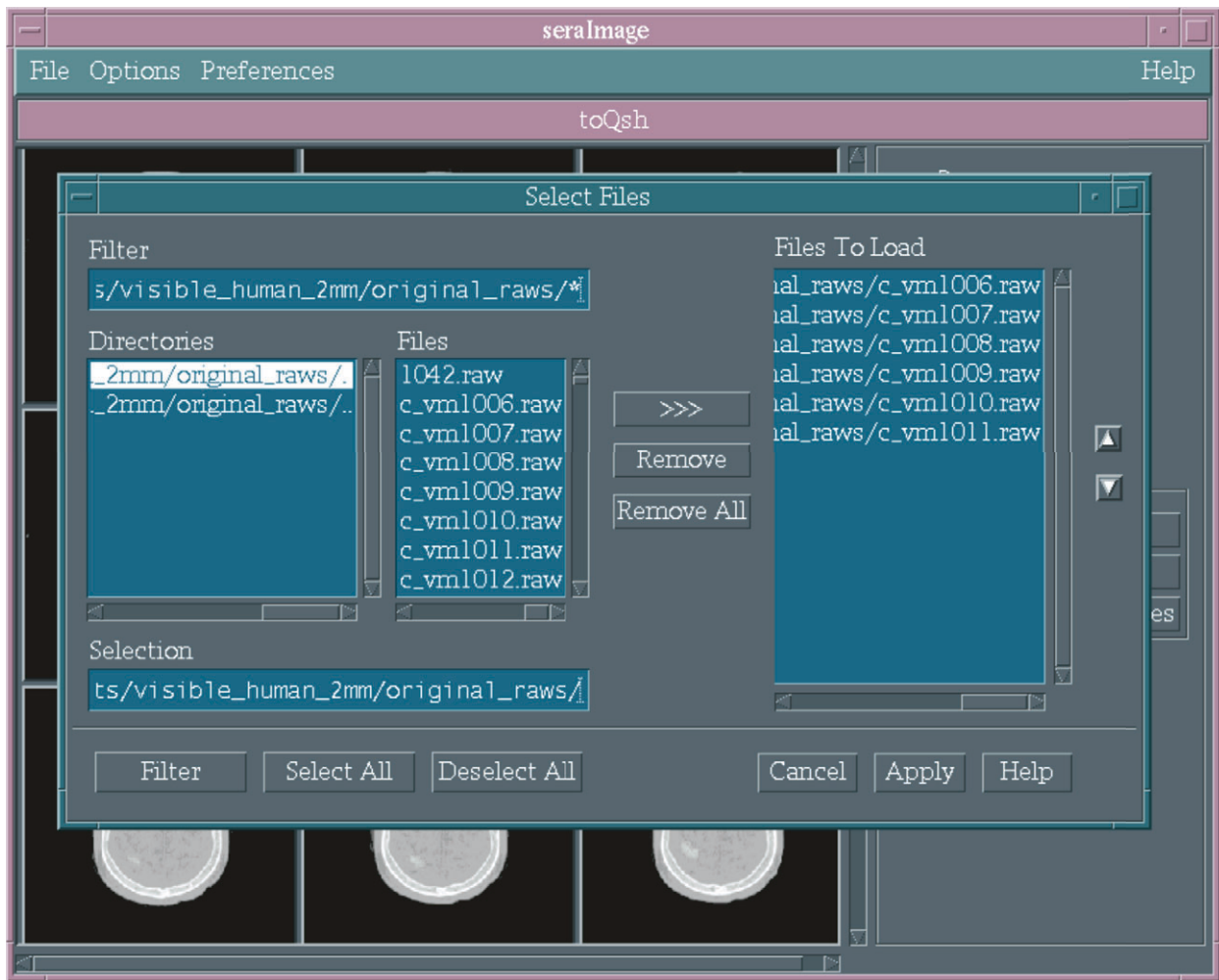


Figure 4.3. Load Single Images selection.

Once grouped, the selected files are loaded by pressing the button labeled with the >>> symbol and choosing the Apply button. At this time, seraImage will only deal with square image sets. A common image size is 256 pixels x 256 pixels x 1 byte. This will yield a file size of 65536 bytes for an image of one byte per pixel. Based on the image's file size, seraImage will make a guess at the image format, while providing the user the opportunity to change the image parameters.

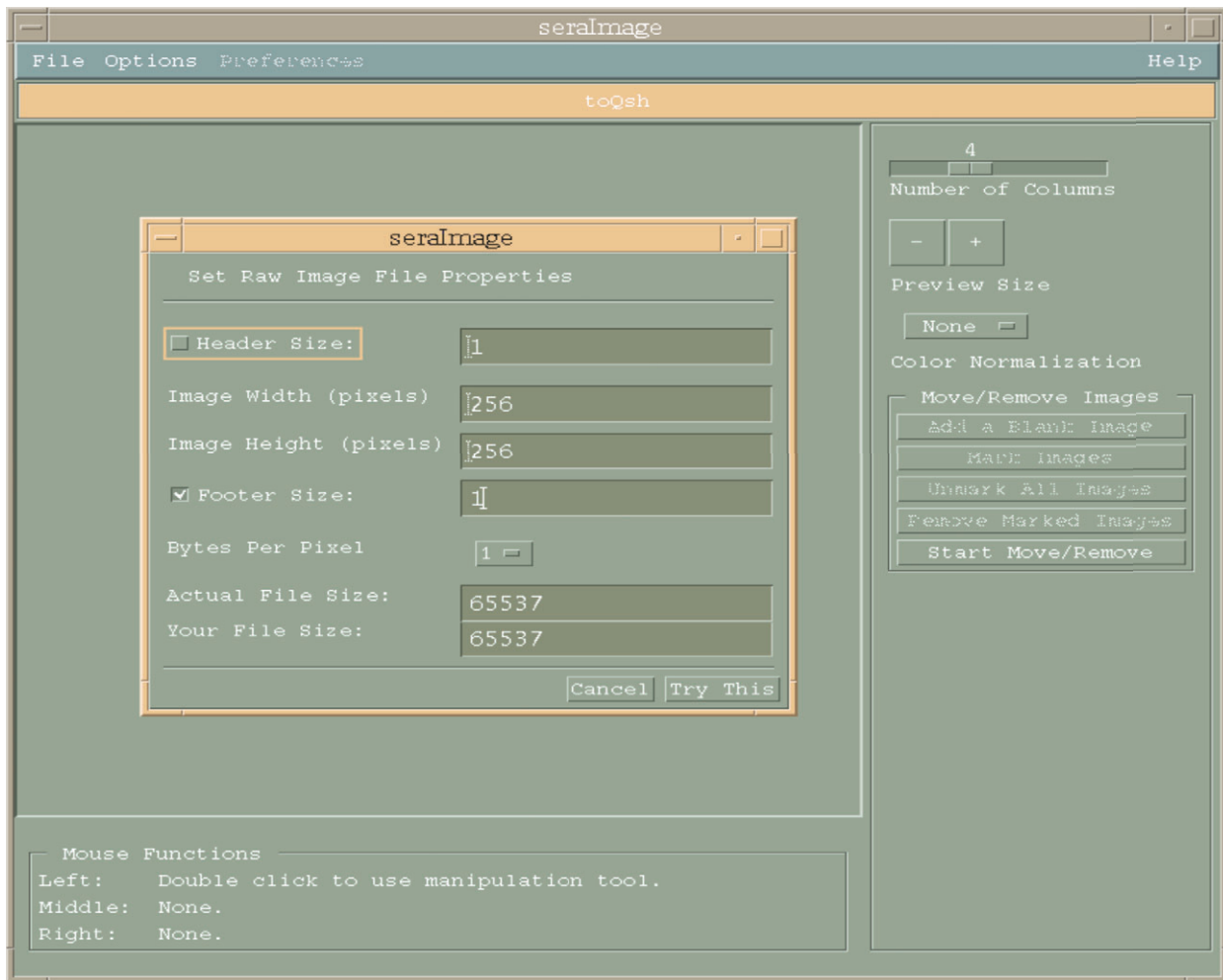


Figure 4.4. Guess at Raw Image Format selection.

If the image contains more than just pixel information, the user can guess at either or both the header and footer sizes (see Figure 4.4). While this is far from a perfect solution, the user still can make several guesses at the image properties, until an acceptable display of the images is achieved.

4.2.3. Unload Images Selection

This is a simple operation that allows the user to remove a current set of images in order to work with a new set of images.

4.2.4. Generate Raw Images Selection

This option is provided so that the user can extract individual raw image files from the QSH image (.qim) file.

4.2.5. Save QSH Selection

This option saves the current seraImage images as a QSH formatted image set.

4.2.6. Launch Selection

From this selection, the user can run the other modules of the SERA system.

4.2.7. Check Version Selection

The most recent version number is maintained at the central BNCT system site at Montana State University. As a convenience to the user, this option compares the version of the currently operating module with the most recent version available. **This feature is no longer functional.**

4.3. Options Menu

The Options menu has the following parts, which will be discussed in turn in following subsections:

- Edit QHD
- Tool Bar
- Reverse Images
- Manipulate Images.

Figure 4.5 shows the Options popup menu.



Figure 4.5. Options Menu selection.

4.3.1. Edit QHD Selection

This option brings up the QSH key editing widget, shown in Figure 4.2, where the user can change the values of the required keys.

4.3.2. Tool Bar Selection

This option displays/undisplays the seraImage buttons on the right hand side of the main widget, in the case where screen space is limiting.

4.3.3. Messages Selection

When this option is enabled, an informational message describing the buttons function appears in the space below the menu bar in the main widget as the cursor passes over the widget buttons.

4.3.4. Reverse Images Selection

This option simply reverses the display and file order of the current set of images.

4.3.5. Manipulate Images Selection

This option allows the user to rotate, translate, or scale a single image or the entire image set, as shown in Figure 4.6.



Figure 4.6. Manipulate Images Menu selection.

4.4. Preferences Menu

This feature is not operational in version 1C0.

5. Image Editor (seraModel)

After the image information has been prepared, image editing is usually the next step in the treatment planning process. Figure 5.1 shows seraModel's user interface in the Edit mode, along with annotated widget components. The purpose of the seraModel module is to break an image set easily and rapidly into regions of interest, and to create the model geometry. This process is described in Section 5.3.2. The image pixel matrix used for display in this program has been generalized to work on systems with different color depths and to view images at an arbitrary zoom level, in an arbitrary window size, and with an arbitrary number of columns.



Figure 5.1. User interface for seraModel in Edit Mode.

SeraModel provides many useful image operations, including manual and semi-automatic (threshold) definition and generation of univel-based regions of interest that form the geometry used by the radiation transport simulation. Various image processing tools are provided to aid in the manual definition of these regions, including region copying, scaling, overwriting, and painting by fill or borders. Threshold-based segmenting, 3D region growing, body copying, and automatic target definition operations are also provided.

Finally, the program will save user-defined regions in a uniform volume element format (.uv/.uvh) to describe both the geometry of the model and the assignment of the physical properties to the univels.

5.1. File Menu

The File menu has the following parts, which will be discussed in turn in following subsections:

- Open
- Save
- Launch
- Remove Bodies
- Restart Program
- Exit Program.

5.1.1. Open Selection

With this selection, the user can open pre-existing image sets, and either compressed (gzip) or uncompressed (.uv/.uvh) body files. Image files can be opened so that they either replace or overlay an existing set of images. Also, the user can open the fiducial and constraint marker files, as well as the files describing the body and material properties. When opening a body file, the user is prompted to enter the fiducial and constraint marker files.



Figure 5.2. Sera file selection widget.

The user has a choice of using the standard Motif file selection widget, or the Sera file selection widget, which maintains a list of recently used files for convenient and quick access. The sera file selection widget is shown in Figure 5.2. The last file accessed is highlighted and set as the default file to load.

5.1.2. Save Selection

This option saves either the modified image files, or the new or modified univel (.uv/.uvh) files. The user has a choice of using the standard Motif file selection widget or the sera file selection widget described in section 5.1.1.

5.1.3. Launch Selection

From this selection, the user can run the other modules of the SERA system. This selection is discussed in Section 3.9.1.

5.1.4. Remove Bodies Selection

This selection removes all of the defined bodies while retaining the current image set. The user must confirm the removal action.

5.1.5. Restart Program Selection

This option resets the program to its initial state. All images and bodies are removed without a need to exit the program. The user must confirm this action.

5.1.6. Exit Program

This selection closes the module with confirmation notice. All other currently running modules are unaffected by this action.

5.2. Edit Menu

The Edit menu has the following parts that will be discussed in following subsections:

- Undo
- Relabel Regions
- Remove Bodies from Image Range
- Colormap
- Constraint Markers
- Fiducial Markers
- Sort Images by Z-Value
- Reverse Image Set
- Slice Orientation.

5.2.1. Undo

This selection allows the user to undo recent image editing actions.

5.2.2. Relabel Regions Selection

It is very important that every pixel be assigned a body (label) in the model to assure a reliable transport simulation. The relabel selection allows the user to reassign bodies using the widget displayed in Figure 5.4. Matching a highlighted selection in the left hand column with one in the right hand column changes the body assignment. For instance, highlighting unlabeled in the left hand column, then highlighting buffer in the right hand column, will assign all unassigned pixels as buffer.

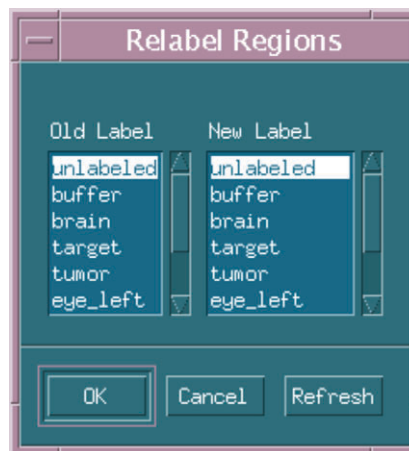


Figure 5.4. Relabel widget selection.

The user needs to exercise a great deal of caution when performing this operation. The buffer is a special body type that should only be assigned in the outermost areas of the image space. If

buffer is assigned to any body inside the outermost body (in the case of the head, this would be the scalp), an error in particle tracking could occur.

5.2.3. Remove Bodies from Image Range

This selection allows the user to remove selected bodies from the image range.

5.2.4. Colormap Selection

With this selection, the user can adjust the contrast, gamma correction, and colormap used to display the images. This is useful during manual editing, when trying to discern certain objects in the image. Unfortunately, there are usually no set rules for setting the values in this widget (see Figure 5.5).

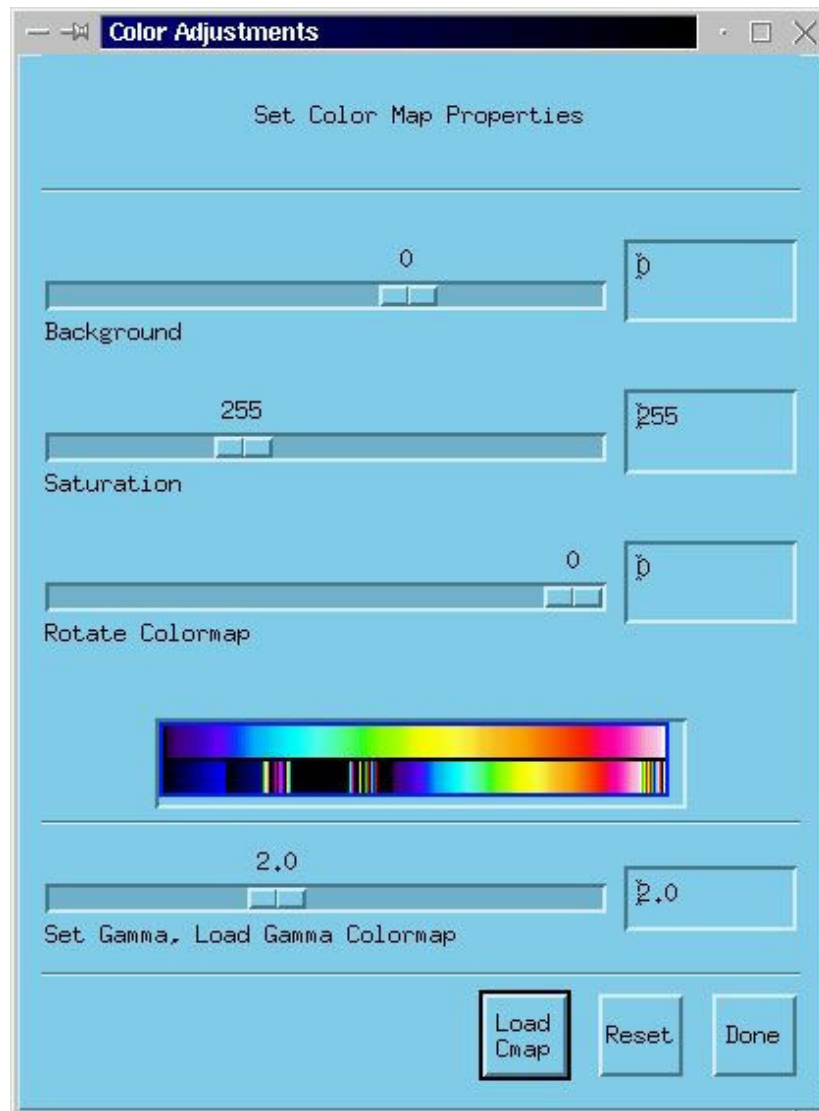


Figure 5.5. The colormap selection widget.

The user will have to experiment in order to obtain a good result. We have noticed that when using CT images from the bone window, the soft tissue is enhanced when the background is set to 65 and the saturation is set to about 150.

5.2.5. Constraint Markers Selection

Constraint markers will be used to control dose to healthy tissues. Their location can be specified in seraModel and read by seraMC. After the transport calculation is complete, seraMC will compute the flux at the marker locations and, given the boron concentration and RBEs specified in the edits, will compute the dose. Constraint markers will have an associated tolerance dose, and seraMC will warn the planner when this tolerance dose is exceeded, so that the plan specifications can be altered to keep the particular structure within desired limits. Default constraint markers are stored in the file Constraint.txt, which must be read by seraModel. Figure 5.6 shows an example for the left eye and the right eye. In this example, the left eye has been specified by a mouse click at the proper location and the right eye is being specified. The right eye marker is still active and can be moved with the cursor before establishing its final location. The body is specified to allow the marker to associate the marker with material properties, including boron ratio relative to blood, and appropriate RBE factors.

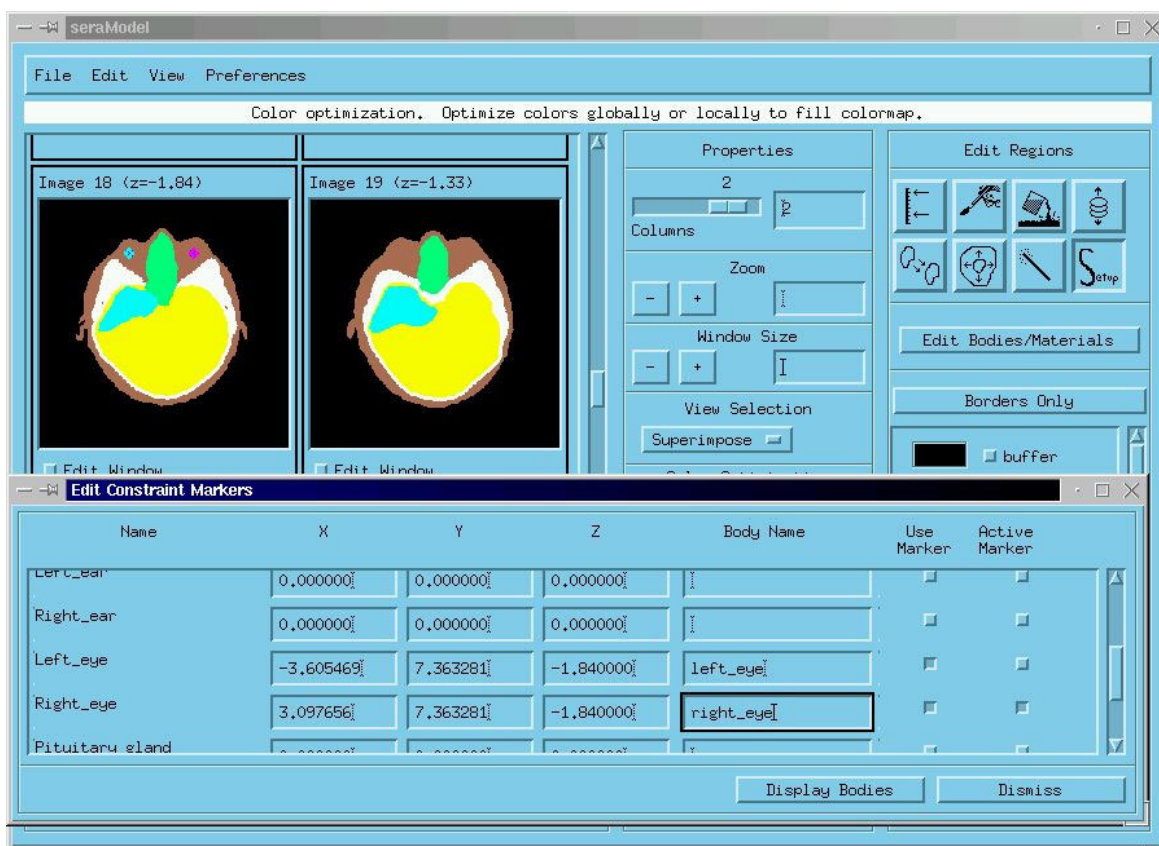


Figure 5.6. Constraint Marker selection.

5.2.6. Fiducial Markers Selection

The patient may be positioned with respect to the beam with the aid of fiducial markers. Fiducial markers are stored in the file Fiducial.txt, which must be read by seraModel. In the case of MRI, for example, the fiducial marker may be a vitamin E capsule located at a tattoo mark. The location of the fiducial marker will be read by seraMC, and the coordinates relative to the beam entry point will be calculated by seraMC and used for the positioning process. Otherwise, positioning requires a three-dimensional translation from patient geometry to beam geometry, which is tedious and quite error-prone. Figure 5.7 shows fiducial markers set for a sample patient geometry.

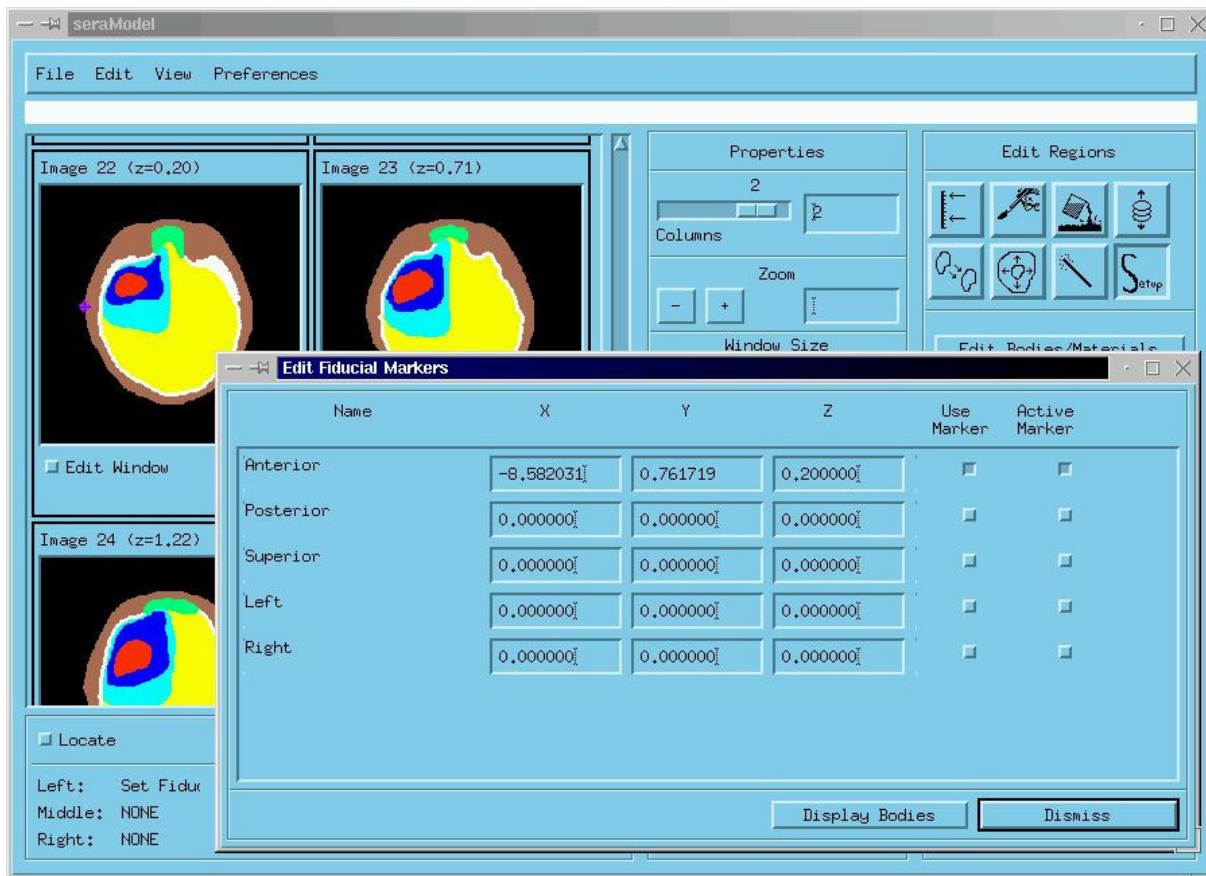


Figure 5.7. Fiducial Marker selection.

5.2.7. Sort Images by Z-Value Selection

In the unlikely event that the user has an image file set that is in random order with respect to the axis of the image set, this selection may prove to be useful. Usually, the user will correct this problem with the seraImage module.

5.2.8. Reverse Image Set Selection

If the images are ordered along the axis of the image set in the opposite direction from that which the user requires, then this selection should be used. For an axial image series of the head, the radiation transport model expects the images to be in increasing order from the bottom to top (inferior to superior). If the images are ordered in the opposite direction, this selection will correct the problem.

5.2.9. Slice Orientation

This allows the user to set the orientation of the row axis, the column axis, and the slice axis.

5.3. View Menu

The View menu has the following parts, which will be discussed in turn in subsequent subsections:

- Properties Toolbar
- Edit Regions
- Context Help
- Bottom Information

- Cross-Section Tool
- Intensity Histogram
- Region Volumes.

5.3.1. Properties Toolbar Selection

The properties for the seraModel widget (see Figure 5.1) are displayed in the column to the immediate right of the image display area. This selection is toggled on/off. Normally, the square button to the left of this selection should be in the on (depressed) position. If the user requires more screen space to display the images, then the button should be set in the off (outward) position. The setting for this selection can be saved in the preferences menu.

The properties column contains buttons and sliders that affect how the images and regions are displayed. The user can adjust widget and image sizes, the number of columns used to display the images, whether to display only images, regions, or regions superimposed on images, pixel optimization, window movement synchronization, and image removal. These adjustments can be quite time consuming when operating on a large set of images.

5.3.2. Edit Regions Selection

The main purpose of this selection is the creation of the univel geometry for the transport model, which is at the heart of the modeling process. It therefore requires more explanation than any other seraModel selection. This selection is toggled on/off, like the properties toolbar selection. When the selection is in the on position, the Edit Regions column will appear on the far right side of the widget, as shown in Figure 5.1. This selection has seven region editing modes and one setup mode. When defining bodies for a model, the user should remember two very important considerations:

1. All pixels must be assigned a body (region).
2. The buffer region must surround all other regions and cannot be assigned to any interior pixels.

The code will generally warn the user when there are unassigned pixels. Beneath each image there is the Edit Window button, which will allow the user to enlarge the image to reveal more detailed pixel information. The user can scroll through the entire image set and determine where the unlabeled pixels are located. This feature is also very useful for defining regions, either when thresholding or when manual editing.

SeraModel will not guard against improper buffer definition, but if the user follows the prescribed editing procedures, this should not be a problem. There is no substitute for careful editing.

In order to make body definition complete and convenient, seraModel provides several editing modes, which are described in the following subsections. The user need not, and should not, use all of these modes, but it is a good idea to be aware of these capabilities without being overwhelmed by them. As the user becomes experienced, certain editing patterns should develop that best suit the user and the image quality available.

The user will soon find that one of the most important buttons in seraModel is the undo button, located on the bottom right hand side of the edit section. Undo can also be accessed by using

Edit->Undo. seraModel stores in its memory queue each edit operation, in order from first to last. Each time the user presses this button, the last edit operation is undone and removed from the memory queue. The utility of this will become obvious to the user almost immediately.

5.3.2.1. Setup Mode

Setup is the default mode of the edit regions selection. Setup is a good place to include all of the bodies that will be used in describing the model, and to assign a material type to each of the bodies through the Edit Bodies/Materials button in the Edit Regions column. Below the Edit Bodies/Materials button, there is a button labeled either Borders Only or Filled Regions. This button is available to all of the edit modes; it is used to set the region display mode. The Borders Only mode displays only the outer boundary of the region, so that the image underlying the region fill values can be revealed.

The color area next to the body name in the Edit Regions column is the color used to display that body on the images. The user can change the color by pointing and clicking in this color area. A popup widget will appear with a color palette, adjustable color mixing sliders, and fixed color buttons. All of these settings can be saved for subsequent modeling sessions by using the Preferences selection described in Section 5.4.

5.3.2.2. Threshold Mode

This mode works best when defining regions that have pixel values distinct from adjacent regions, as when defining scalp and skull in some CT series. Even in these cases, pixel value ambiguities occur between low density regions, such as the sinuses and the area surrounding the scalp (buffer area). This mode is less useful when working with MRI images. Even so, we have provided this mode because when it works, it will save modeling time, and this is especially true when you have an image set with high contrast, uniformity, and low noise. As SERA matures, we plan to develop and implement automatic (unsupervised) segmenting algorithms, but the methods we have examined thus far are less efficient than manual editing, chiefly because of the computation time resources required by these methods.

Figure 5.9 shows seraModel in threshold mode, with some sinus and skull regions defined. Thresholding works by defining a region as those pixels in an image with intensity values bounded by minimum and maximum pixel values. For 8-bit images, these values range between 0 and 255. In Figure 5.8, we are defining the skull as those pixels ranging between 187 and 238. This can be accomplished by either setting the sliders located on the right hand side of the figure at the desired range, or by using the mouse buttons to determine the range. This can be done by observing the mouse button functions, as indicated in the bottom info section of the seraModel.

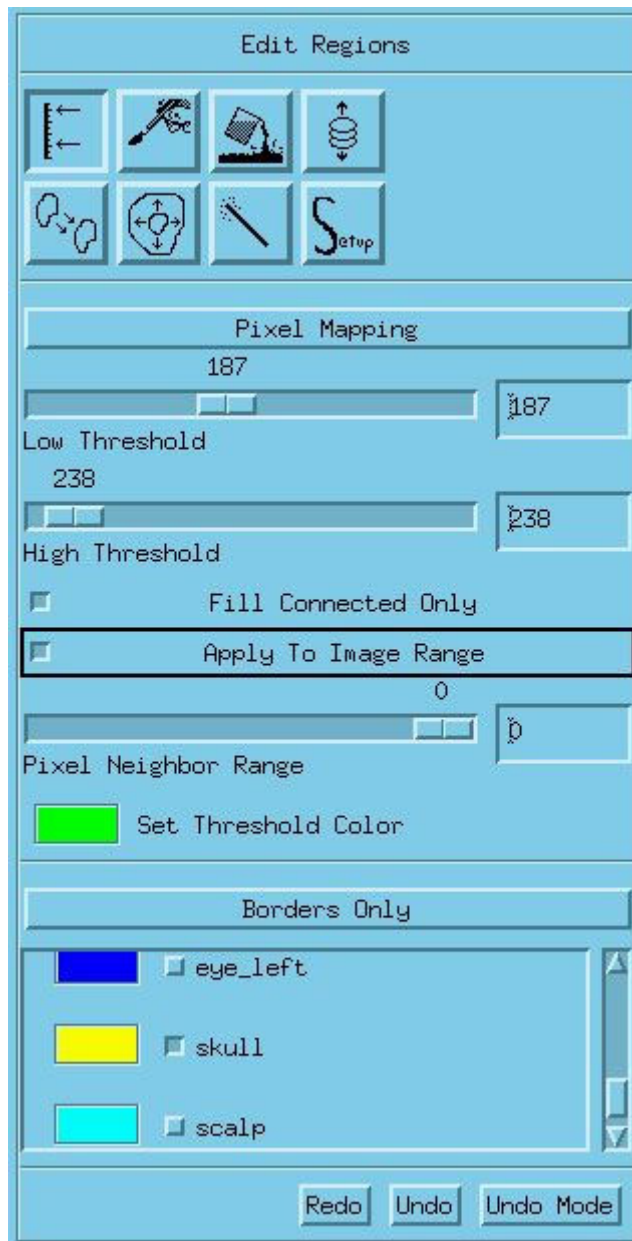


Figure 5.8. Thresholding mode selection.

Passing the cursor over the region in the image, while holding down the middle mouse button, will set the threshold range. Then, to fill in the region definition, while the cursor is in the region to be thresholded, press the left mouse button. This will take some experience to use effectively, and, as stated earlier, thresholding has limited applicability, working best with certain CT image sets.

Despite all of this, the thresholding mode is probably the best place to start defining some of the regions, like the buffer. Because the buffer, which must be distinct from all other regions, has pixel values similar to some of the low density regions, such as sinuses, a simple mapping of all low density pixels to the buffer will not work. Figure 5.9 shows the Edit Window while in threshold mode.

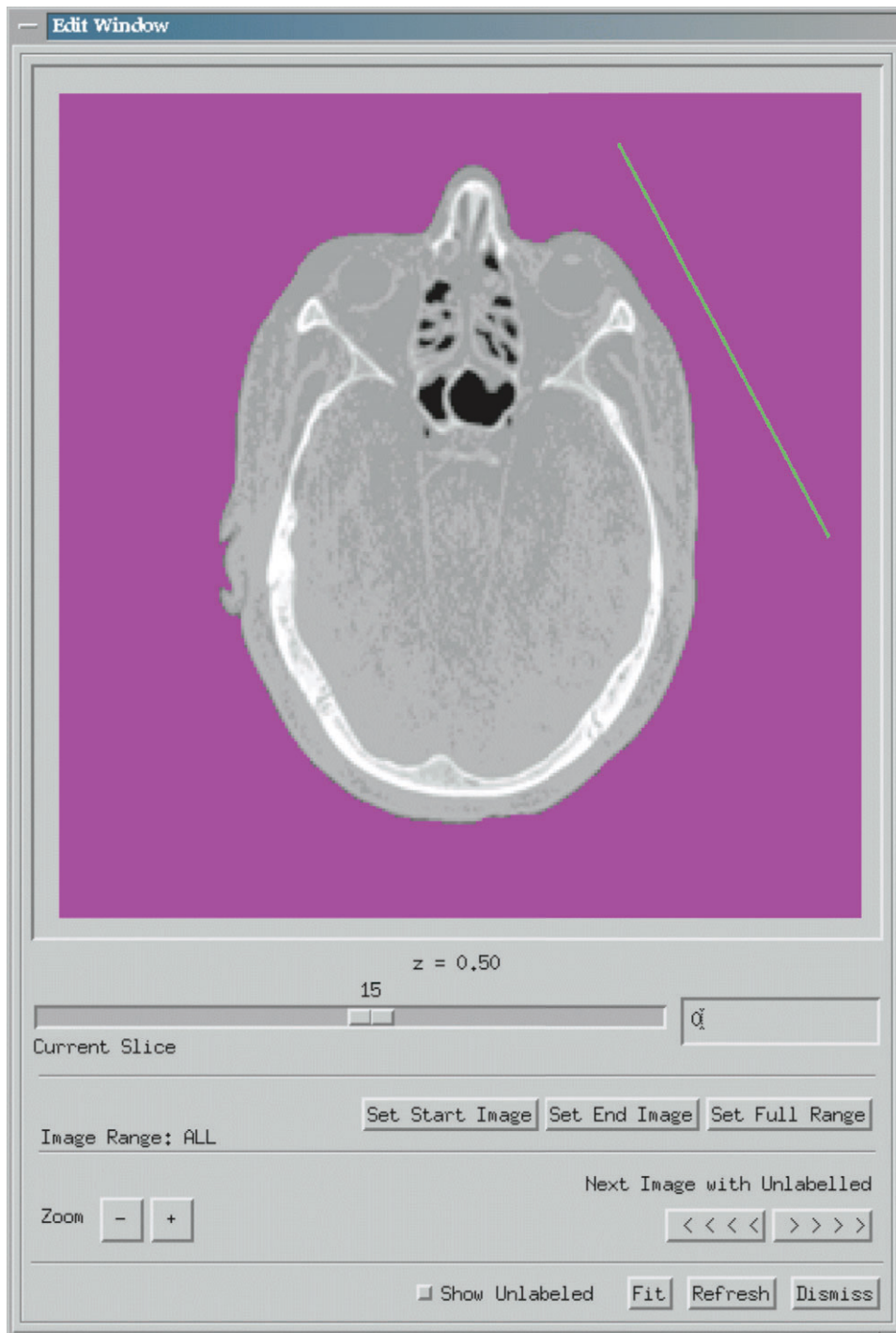


Figure 5.9. Edit Window while thresholding.

The mouse button assignments are shown in the bottom information section of the main seraModel window. Dragging the cursor through the buffer area of the image, while holding down the middle mouse button, will set lower and upper threshold values along the line shown in the buffer area of Figure 5.10. The threshold sliders will be automatically set following the release of the middle button. Alternatively, the user can set the values by moving the sliders. Once these values are set, the user can threshold the currently displayed image, or all of the images in a pre-specified range, by setting the appropriate radio button underneath the sliders.

There is a second radio button labeled Fill Connected Only. This button will only attempt to threshold contiguous regions of the specified pixel range, and it should be set to the on position when defining the buffer. When applying the fill threshold operations to the entire image range with Fill Connected Only set, the user should be careful to note that the filling is based on the cursor position in the reference image, and this may not produce the desired effect in the other images in the range. If this is the case, the user should try unsetting Fill Connected Only or Apply to Image Range. In the threshold mode, with the right mouse button depressed, the Fill Neighbor Range slider is operative and will fill the image with current body based on the pixel value (+/- the fill neighbor range value) at the cursor location.

After setting the low and high threshold sliders, position the cursor in the buffer area and press the left mouse button to fill the buffer area of the image(s) with the buffer region. The Fill Connected Only mode should prevent buffer material from being assigned to the voids inside the image area, if there are no penetrations into the image area caused by surface aberrations. Voids inside the image area that have the same threshold range as the buffer can now be assigned. If the user wants to get all voids within the image area assigned at one time, he should make sure the Fill Connected Only mode is not set.

At this point, it is worth noting several of the Edit Window functions that are especially useful during thresholding and manual draw operations. From any of the Edit Windows the user can:

- scroll through all of the slices of the image set using the current slice slider
- set the image range for various thresholding and editing operations
- zoom image to reveal further detail for the purposes of editing and inspection
- display unlabeled and mislabeled regions.

Through practice, the user will acquire sufficient familiarity with these and other related features to fully exploit them for model development.

Selecting the "Pixel Mapping" button will display the pixel mapping widget shown in Figure 5.10. The widget is used to map a pixel range to a body. Pixel mapping can be used to define bodies with well-defined pixel ranges, or it can be used in conjunction with thresholding to fill in the remaining pixels. To map a pixel range from the map to a body, first select the body in the right side of the widget, move the cursor into the pixel mapping histogram display, and, while holding down the left mouse button, drag the cursor through the desired pixel range. Once the pixel ranges are assigned, selecting the Apply Pixel Mapping button will apply this mapping to the current image range.

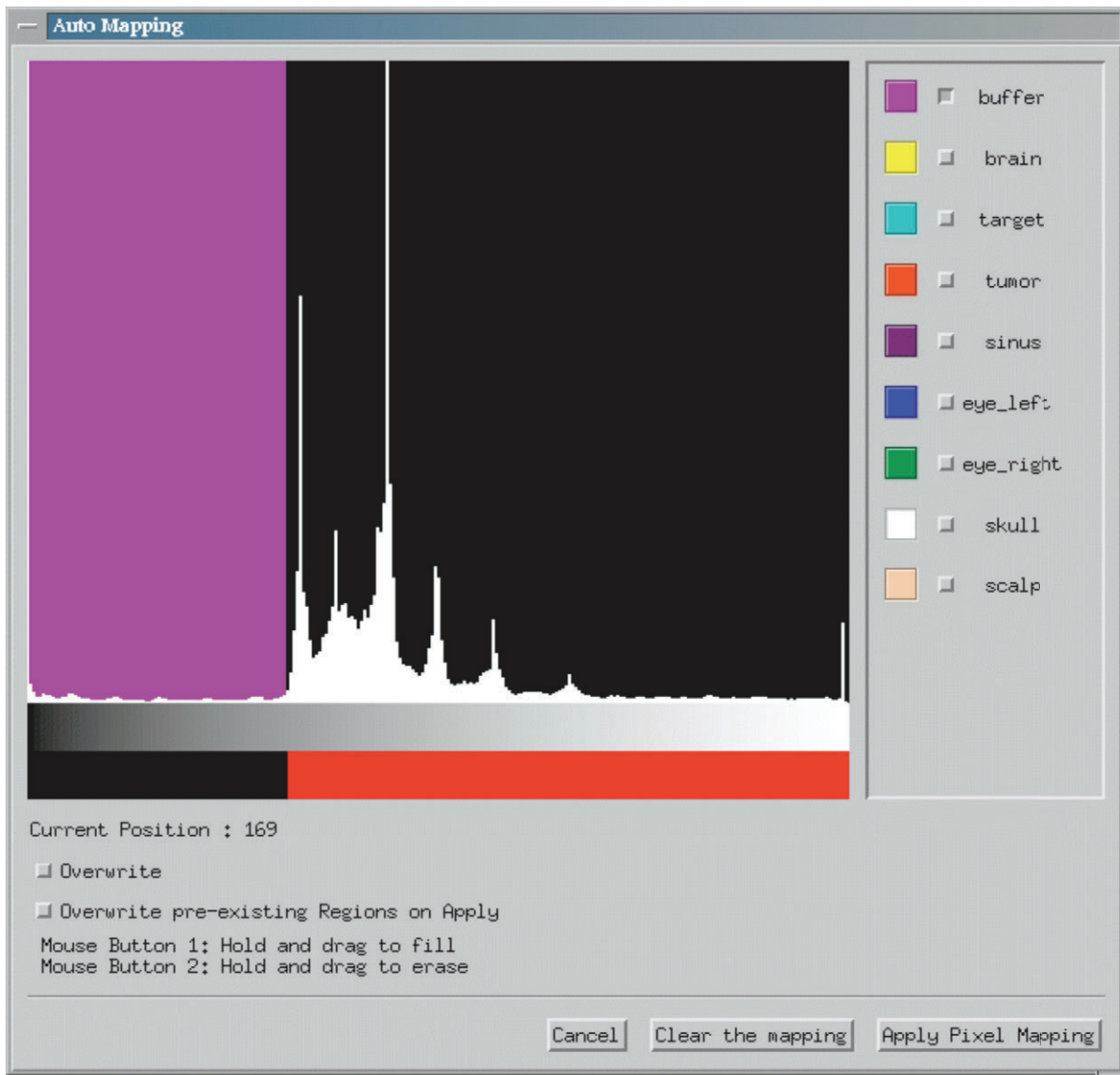


Figure 5.10. Pixel Mapping widget.

5.3.2.3. Manual Draw Mode

This is the simplest and most robust method for defining bodies. It is also the most tedious. While in the manual draw mode, the user can define any body with one of five paintbrush sizes. The brush sizes are selected by placing the cursor on the appropriate brush size icon (see Figure 5.11) and pressing the left mouse button. Once the brush size icon is highlighted and a body is selected, the user can move the cursor to an image and draw the body outline using the left mouse button, as with many popular paint programs. The middle mouse button is used to erase, and, once the border is defined, the right mouse button is conveniently used to fill in the region. The overwrite option box allows switching between the options:

Don't Overwrite

Only allows the user to draw on unlabeled regions. Only allows the current region to be erased.

Overwrite All

Allows drawing on any region. Allows any region to be erased.

Overwrite Selected

Pressing the Select Overwrite Bodies allows the user to select which bodies can be overwritten and erased. Overwrite Selected allows you to only overwrite the selected bodies (note: The current body is always selected).



Figure 5.11. Manual Draw mode selection.

While the user can draw into any of the image windows, use of the Edit Window is highly recommended, because of the features outlined in section 5.3.2.2. The authors recommend that the user become skilled with this mode by experimentation and liberal use of the undo button. In the long run, this skill will save considerable modeling time.

5.3.2.4. Fill Mode

This feature is used to fill regions bounded by previously defined boundaries. In the Fill mode, as in all other seraModel modes, the bottom information area will show how the mouse buttons are mapped to the mode operations. The left mouse button will make an educated guess at the filling body. For example, the thresholding operation can miss pixel labeling when defining bodies, such as the skull. The user can conveniently fill in these missed pixels with this operation. Once some bodies are defined, the middle button can be used to fill inside or between these bodies with a selected body. In the Fill mode, the right mouse button will display a filled region with only its inner and outer borders.

5.3.2.5. 3D Grow Mode

As with most other semi-automatic body definition strategies, this one works best with images containing narrowly defined pixel intensities. Once a body is defined in one slice (the Base Growth Index), this operation will try to define that body in the range of slices set by the sliders shown in Figure 5.12. At this time, the Base Growth Index must be within the selected range.



Figure 5.12. 3D Grow mode selection.

If the Overwrite button is selected in the grow operation, it will replace previously defined bodies within the grow volume. Normally, the overwrite button will not be selected.

3D Grow has four methods that can be used:

Edge Threshold

Works best for regions that have an edge that has a pixel intensity greater or less than the surrounding area but the interior area is very similar to the exterior (i.e., if the region has a bright white or dark black boundary)

Picky Edge Threshold

Similar to Edge Threshold but more selective about pixel intensity.

Whole Threshold

Uses all the pixel intensities in the region for the growth operation.

Old Method

Uses pixel intensities in the region and the physical boundaries of the region on the base image for the growth operation.

5.3.2.6. Copy Body Mode

When there are small slice-to-slice variations in the size and shape of a body, this feature may be time saving. Once outlined and filled, a body may be copied from any slice to any other slice, using sliders as shown in Figure 5.13. Like manual draw, this mode has the overwrite option.

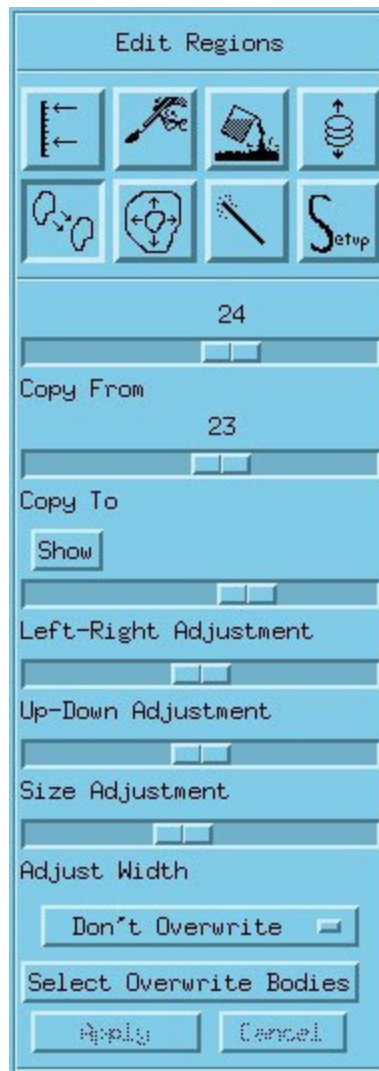


Figure 5.13. Copy Body mode selection.

The user can show how the copy operation appears and can adjust the position, size, and width of the copied body before applying the operation. This feature seems to work best with tumor and brain bodies. Once applied, this operation can be undone with the undo button.

5.3.2.7. Make Margin Mode

For many tumor types, tumor cells can be found outside the main tumor mass. The main tumor mass is enhanced in the image, but the tumor cells outside the main tumor mass are not enhanced. This robust feature is used to define the target volume containing the non-enhanced part of the tumor surrounding the defined tumor object. Because the target volume usually only occurs in the brain region, it is important that the Make Margin operation is the last edit step, so that brain can be the only region overwritten.

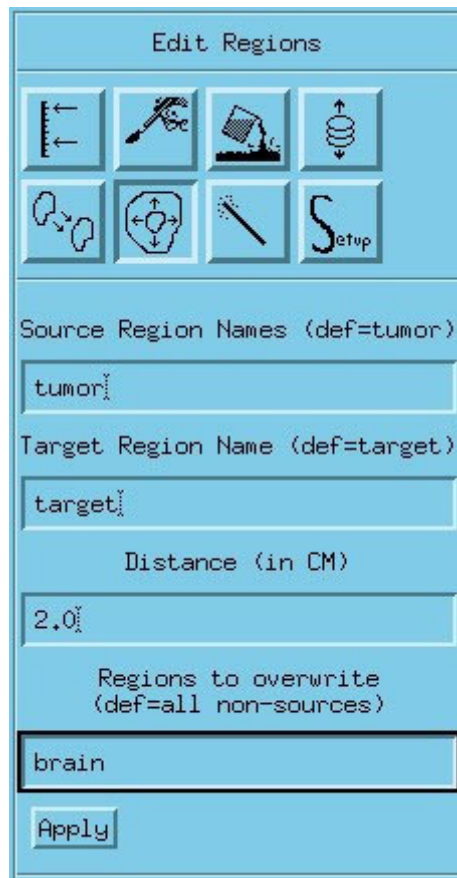


Figure 5.14. Make Margin mode selection.

The Make Margin mode, shown in Figure 5.14, will create a region of a specified thickness surrounding the Source Region, with the thickness applied in three dimensions relative to the surface normal vector. It is important to note that the Source Region may be more than one body, e.g., tumor and edema.

5.3.2.8. Wand Mode

The wand tool is a thresholding tool like the Thresholding Mode; however, the wand tool uses the pixel value directly under the cursor hotspot to threshold. It has three different uses depending on which button is pressed:

Guess Threshold

This is used when the left button is pressed. It looks at the area surrounding the mouse pointer and uses that sample of the image to try to find a threshold for that region. Then that threshold is used to flood-fill the region.

Grow Region

Expands the region over which the mouse pointer is currently located by one pixel up, down, left and right.

Shrink Region

Shrinks by one pixel the region over which the mouse pointer is currently located. This can be used with grow region to do closings and openings. A closing is a Grow followed by a Shrink. This closes any small holes in the region created with Guess Threshold. An opening is a Shrink followed by a Grow. This removes any small protrusions from the

image. In order to do a closing that closes larger areas, use multiple grows followed by the same number of shrinks.

The methods to choose from are Careful Narrow, Narrow, Normal, and Wide. Each of these determines the patch size that Guess Threshold uses. If you want the wand to be less discriminating, use Wide. If you want it to use a smaller threshold, then use Narrow. Region Glue determines whether or not a shrunk region will shrink away from another region.

5.3.3. Context Help Selection

When the cursor passes over some of the buttons in the main window, a brief explanation of those buttons will appear in the message area, if this selection is set.

5.3.4. Bottom Information Selection

If the Bottom Information item is selected, the bottom portion of the main widget will contain information regarding the mouse button mappings, the location button, and the image range applying to certain operations, such as thresholding.

5.3.5. Cross-Section Tool Selection

The cross-section tool shown below in Figure 5.15 provides an "at a glance" view of the image and body information within the model. The user can choose views along any of the three principal axes.



Figure 5.15. Cross-Section tool widget.

5.3.6. Intensity Histogram Selection

This selection, illustrated in Figure 5.16, shows either the pixel distribution of an individual image, or the entire image set. It may be of value in showing the image definition and the suitability of the various edit modes for the image set.

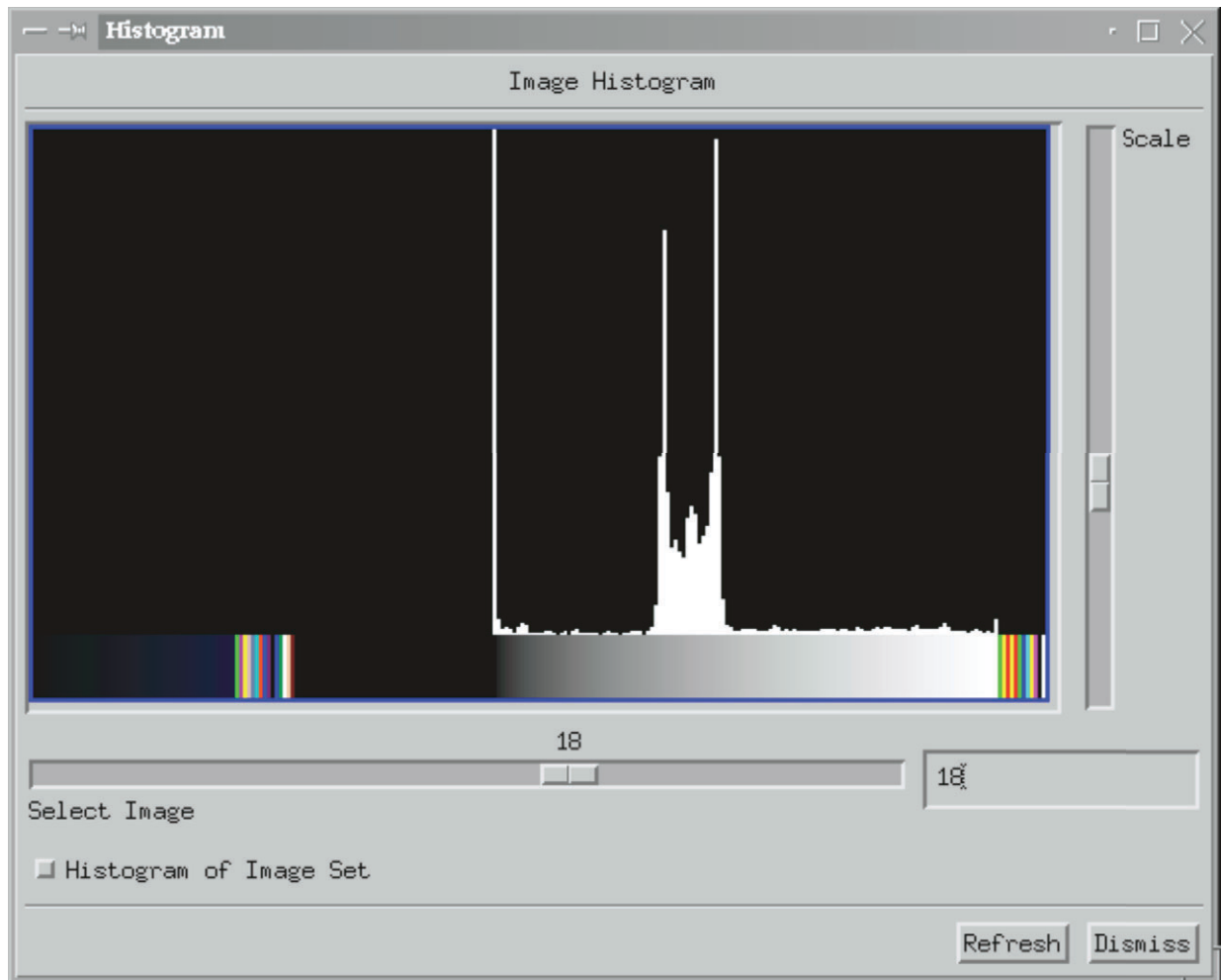


Figure 5.16. Intensity Histogram widget.

5.3.7. Region Volumes Selection

The Region Volumes widget, illustrated in Figure 5.17, shows the volume distribution for the current model. It can reveal gross inconsistencies in the model and show if there are any unlabeled regions remaining in the model.

NAME	NUMBER	CUBIC CM	PERCENT
unlabeled	0	0	0.00%
buffer	1305901	13577	68.71%
brain	151120	1571	7.95%
target	8898	92	0.47%
tumor	427	4	0.02%
sinus	9022	93	0.47%
eye_left	977	10	0.05%
eye_right	1025	10	0.05%
skull	51179	532	2.69%
scalp	371995	3867	19.57%

OK

Figure 5.17. Region Volumes widget.

5.4. Preferences

5.4.1. Preferences Menu

The user can customize any of the features shown in Figure 5.18 for subsequent use in seraModel by selecting the button adjacent to the preference.

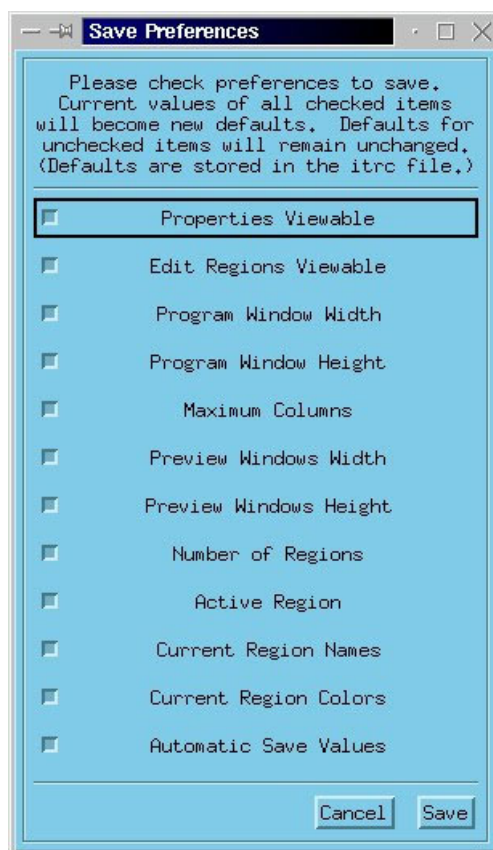


Figure 5.18. Preferences menu.

5.4.2. Autosave Options

Autosave Options change how autosave works. If autosave is on, seraModel will automatically save the current regions periodically to disk.

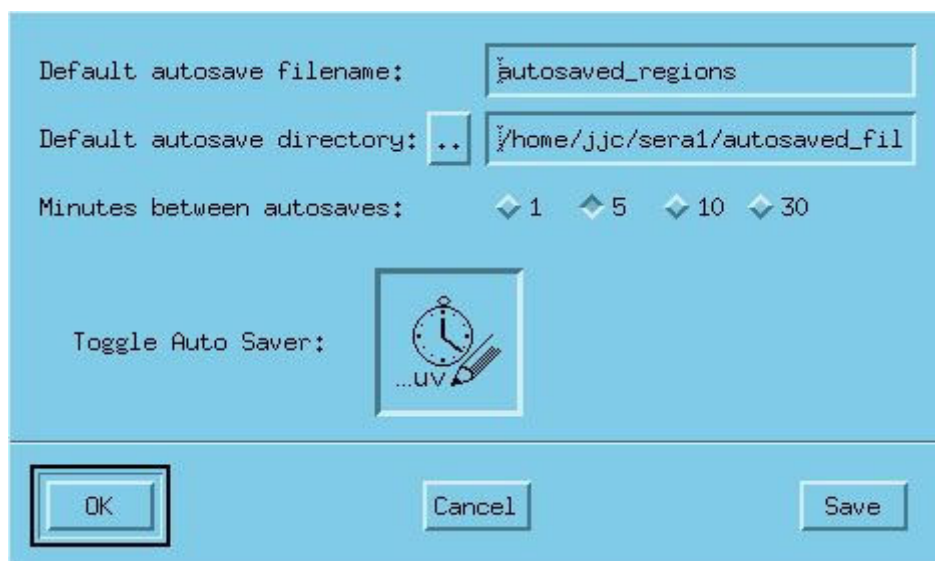


Figure 5.19. Autosave options.

If the file has not been named yet the autosaved files will be saved at:

\$HOME/sera1/autosaved_files/autosaved_regions.uv and
\$HOME/sera1/autosaved_files/autosaved_regions.uvh (i.e., the default autosave directory plus the default autosave filename plus .uv and .uvh).

The first autosave will take place five minutes after the program starts, and subsequent autosaves will occur every five minutes thereafter. Autosave is on since the button to the right of Toggle Auto Saver is pressed.

5.5. Suggested Modeling Steps for BNCT Treatment Planning

There are many ways to develop the patient model using seraModel. This section discusses steps we have found useful for developing a model. An accurate model can be constructed with an image separation of 4mm along the transverse direction. A finer resolution will increase the workload and require more computer memory with a diminishing return on accuracy. When modeling lesions or objects that are less than 4mm along the transverse direction, it will be necessary to use a finer resolution image set. It is important to note again that the basic property of the univel representation is that image transverse spacing must be uniform in order to achieve high performance.

Because CT and MRI image sets are fundamentally different with respect to how the pixel intensity values are derived, it is necessary to consider separate modeling steps for each. CT images have high geometric fidelity, low image noise, and good bone definition, but poor soft tissue contrast. MRI images provide excellent soft tissue resolution, but bone information must be inferred, and the images are usually noisy. Segmenting MRI images using thresholding is of limited value.

As a general rule, the modeler should try to assign contrasting colors to each body. The buffer body should stand out among all others, so that it can be distinguished when found in the image area. The Edit Window allows body definition at the resolution of an individual pixel, and it can be used to detect unlabeled and improperly assigned pixels. This is a good way to detect if a buffer pixel crept into the image area while thresholding. When first learning seraModel, the modeler will, and should, make generous use of the undo button.

5.5.1. Suggested Modeling Steps Using CT Image Sets

If there are no pixel ambiguities, and the pixel value ranges of each body are known, then the modeler should use the Pixel Mapping button of the thresholding option, make the body assignments to the pixel ranges, and apply pixel mapping; then the model is complete. Otherwise, remembering to frequently save the .uv/.uvh files during the modeling steps, the modeler should try the following:

- Using the thresholding option, define buffer with the Fill Connected Only and Apply Threshold To Range buttons set to the on state.
- Define low-density bodies (i.e., sinus) with Fill Connected Only set to the off state.
- Manually define tumor, eyes, and other relatively small objects that do not occur in the majority of slices.
- Attempt to threshold the brain. If that doesn't work out, use the Manual Edit mode for brain.
- Threshold skull, bone, and cartilage material. Paint in what thresholding missed.
- Use full threshold range with Overwrite turned off to set the last body, in this case, the scalp.
- Touch up improperly assigned pixels using the Manual Edit mode.
- Use Make Margin to define the target. Only allow the brain to be overwritten with the target body.

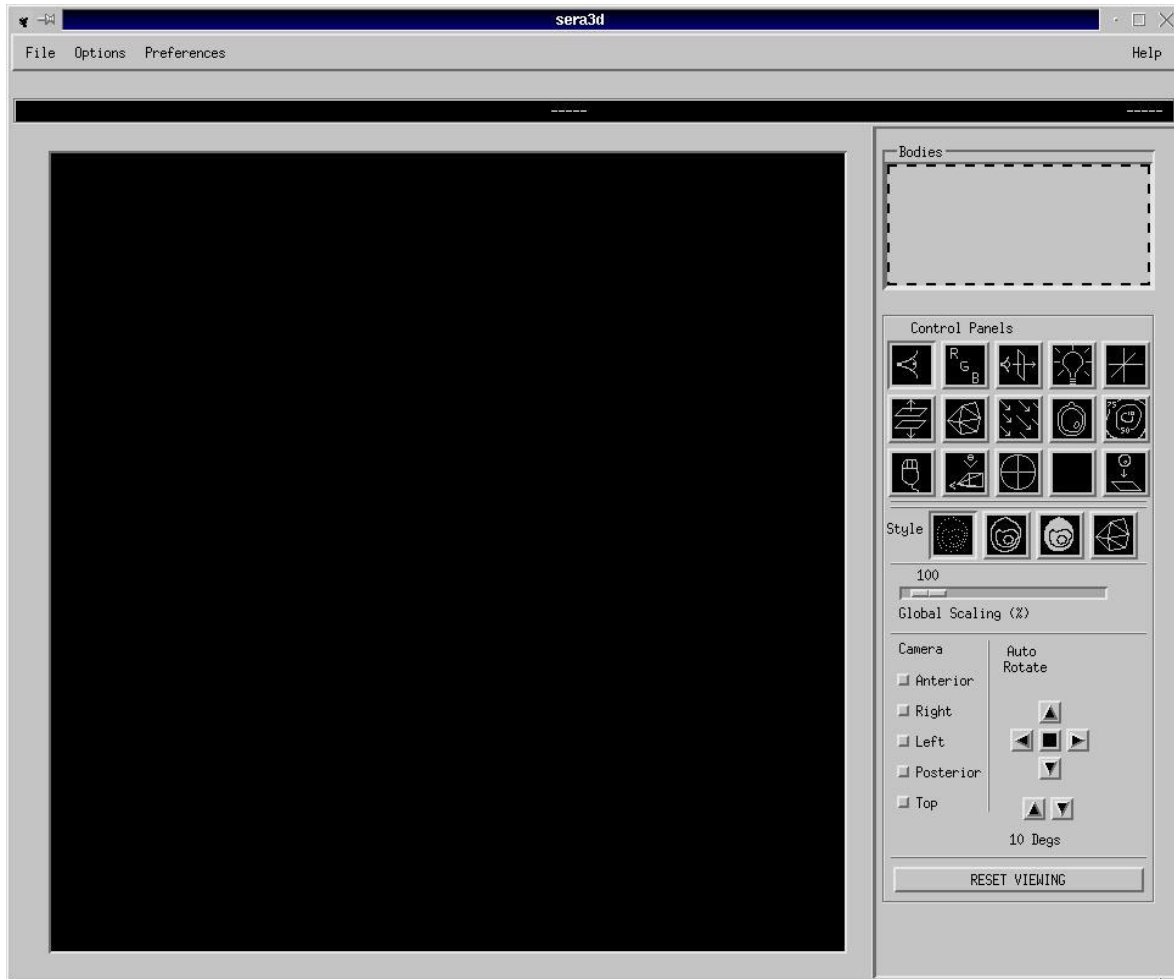
5.5.2. Suggested Modeling Steps Using MRI Image Sets

Defining bodies in MRI images will be dominated by Manual Editing, Wand and Filling. The Wand tool is fairly effective on brain, tumor, and sometimes skull. If the image set has high signal to noise ratios, or can be effectively made so, the user may be able to use thresholding to define the very low (buffer, sinus) and very high (edema, tumor) signal regions making sure the Fill Connected Only state is set on. The modeler should try thresholding, followed by Manual Edit touchup, using experience as a guide, for unfiltered MRI image sets.

6. Three-Dimensional Reconstruction (sera3d)

Note: The Sera3d module will not properly render data sets with slice dimensions larger than 256x256.

The main interface for sera3d is composed of four main parts: the menubar, the message bar, the main rendering window, and the tools.



6.1. The Menubar

6.1.1. The File Menu

Figure 6.1 shows the sera3d File menu.

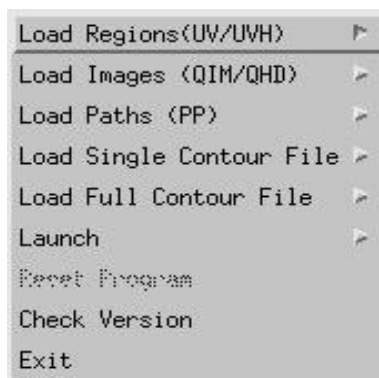


Figure 6.1. File menu selections.

To begin a reconstruction, the user must first load a univel file pair (.uv/.uvh), which is created by seraModel. This is done with the Load Regions menu item. The cascading menu from Load Regions will remember recently used files. A file select box is also available under the cascading menu. Other menu selections will remain grayed out until regions have been loaded.

Once the regions are loaded, other file types may also be loaded. These include the medical images, which can be loaded with Load Images (QIM/QHD), particle track information generated by seraMC with the Load Paths (PP) option, and either single or 3D isodose contour files, respectively, with the Load Single Contour File and Load Full Contour File selections. The Launch menu item, which is present in all modules, allows launching of the other modules of the SERA package. Choosing the Exit button closes sera3d.

6.1.2. The Options Menu

The selections available from the Options menu are shown in Figure 6.2. The Options Menu gives various options for controlling the interface.

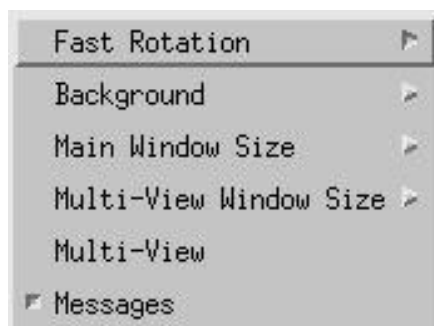


Figure 6.2. Options menu selections.

The Fast Rotation option applies to the reconstruction. When using smooth surfaces or solid rendering, the rendering speed is not always interactive (especially without hardware accelerated graphics). To dramatically increase the rendering speed of rotations, Fast Rotation is used. When rotating an object, the object may be rendered as a wireframe, or not at all, to keep the rendering interactive. The choices for Fast Rotation are:

- Wireframe – while the object is being rotated, it will be drawn as a wireframe

- Axes - the object being rotated will not be drawn, but only the coordinate axes will be shown during rotation
- Off - the object will be drawn in its current rendering style.

The Background option will set the background of the main rendering window to one of the following choices: Black, 25% gray, 50% gray, 75% gray, or white. Main Window Size sets the size of the main rendering window.

Sera3d has the ability to show multiple rendering windows. This feature is called multi-view. When this is enabled, there will be three small rendering windows to the left of the main window, each through a different camera. Two options control the multi-view renderings. Multi-View Window Size sets the size of the three multi-view rendering windows, and Multi-View simply toggles the multi-view feature on and off. The Messages option toggles the message bar, just below the menubar, on and off.

6.1.3. The Preferences Menu

The Preferences menu will simply bring up the preferences dialog window. The preferences dialog is covered later in this section of the manual.

6.1.4. The Help Menu

The Help menu at the top right of the menu bar provides context-sensitive help. To use it, simply click the Help item, and the cursor will turn into a question mark. Then, click the question mark on the item about which you have a question. This will pop open a help window on that item.

6.2. The Message Bar

The sera3d message bar is shown in Figure 6.3 below.



Figure 6.3. Message bar for sera3d.

The message bar in sera3d provides interactive help on the various controls, as well as status information during rendering and reconstruction. As the mouse pointer moves across the various tools in sera3d, the message bar reports the name and use of the tool. The center of the message bar displays the messages, while the right side of the message bar displays the name of the univel file in use.

6.3. The Main Rendering Window

The main rendering window is the large window in which sera3d renders the reconstruction. Clicking the right mouse button in the window brings up a popup menu that switches between using the mouse and using sliders to rotate the reconstruction.

6.4. The Tools

The sera3d Tools panel contains the various mechanisms for manipulation of data displays. The Tools panel is shown in its default state in Figure 6.4.

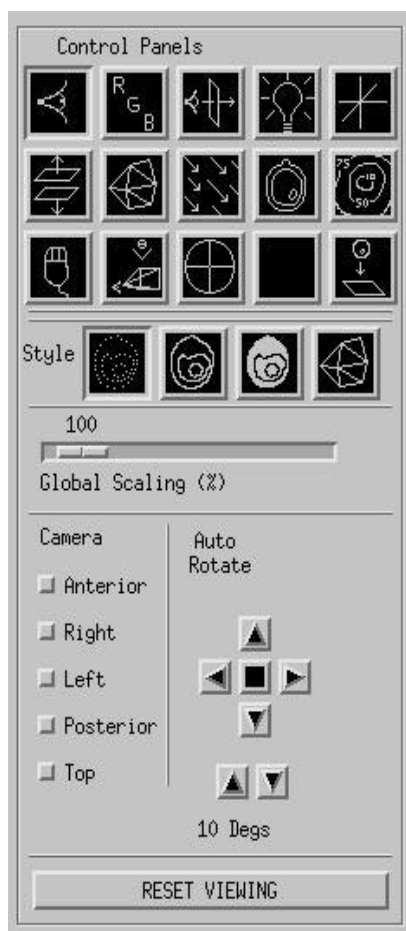


Figure 6.4. The sera3d Tools panel in its default configuration.

The tools for sera3d are all laid out to the right of the main rendering window. The top of the tool bar has the Bodies window. The Bodies window lists all of the segmented regions in the current univel file. All of the listed bodies can be toggled on/off simply by clicking on their name in the Bodies window. In the current image all bodies are toggled on (default). When a body is toggled off, it is removed from the rendering window. Toggling bodies on and off can be done at any time. Below the Bodies window are the Control Panel buttons. The Control Panel buttons switch between the available control panels. The Control Panel is located below the double divider, beneath the Control Panel buttons.

In the example shown in Figure 6.4, the Control Panel begins with the Style panel and continues to the bottom of the Tools panel. The button corresponding to the active panel is depressed - in this case, the first panel (viewing panel). Each panel provides various tools in sera3d. The Control Panel buttons are all labeled with icons. As the mouse moves over an icon, the message bar displays the meaning of the icon. The buttons, in the order of their appearance on the Tools panel, are as follows:

Viewing	Color	Transparency	Lighting	Axes
Clipping	Polygon	Particle Tracking	Slice	Contour
Mouse	View Parameters	Beam	Unused	Texture Mapping

All of the control panels and their tools are explained in the subsequent subsections.

6.4.1. Viewing Panel

Figure 6.5 shows the panel configuration for the Viewing panel.

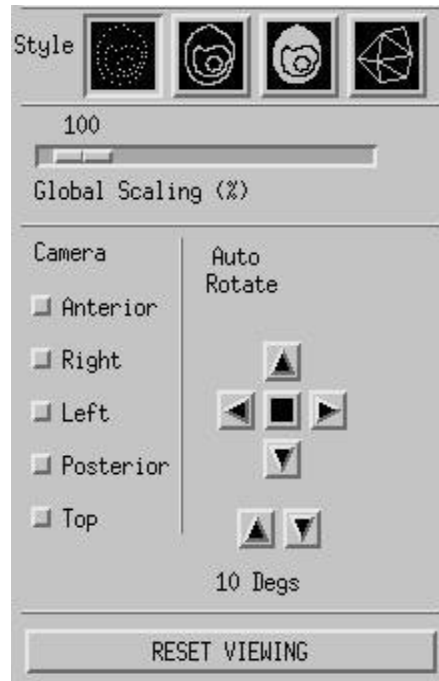


Figure 6.5. Viewing panel layout.

The top of the Viewing panel has four view style buttons. These control the rendering style for the segmented bodies. The first is the Wireframe rendering style (default). Next is the Solid Outline style. The Solid rendering style is the third, and the Polygonal Surfaces style is the last button. The greatest image fidelity is provided by the polygons, but the Wireframe provides the fastest rendering performance. The other styles offer degrees of compromise between the two.

Below the view style buttons is the Global Scaling slider. The slider allows the rendering to be scaled to a larger or smaller size. Below and to the left is the Camera panel. This selects the camera angle for the rendering in the main window. The default camera angle is not in this list. The Reset Viewing button resets the scaling and camera to their default values, and sets the rotation of the rendering back to its original position. The Auto Rotate feature will start and stop the rotation of the rendering. The four arrows control the direction of the auto rotation. The center button stops the auto rotation, while the two arrows below control the speed of the rotation.

6.4.2. Color Panel

The Color panel layout is illustrated in Figure 6.6 below.

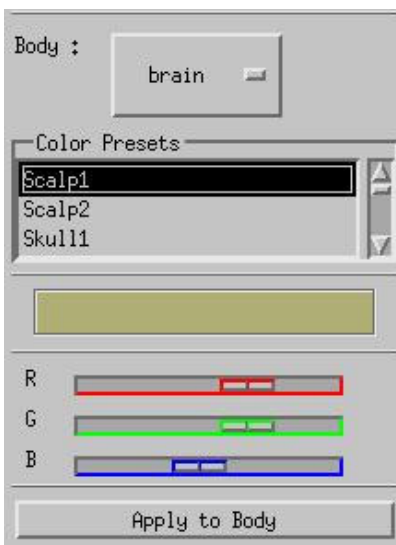


Figure 6.6. Layout of the Color panel.

The Color panel sets the colors of the rendered regions. The Body menu at the top provides the list of bodies available in the univel file; the selected body is the body that is changing color. Below the body menu is a Color Presets list. This list is simply a few preset colors for various regions. By clicking on a preset color, the sliders and color swatch will be updated to the preset values. The color swatch displays the current color set by the red, green, and blue sliders. Finally, to apply the selected color to the selected body, the Apply to Body button is used.

6.4.3. Transparency Panel

The layout of the Transparency panel is illustrated in Figure 6.7 below.

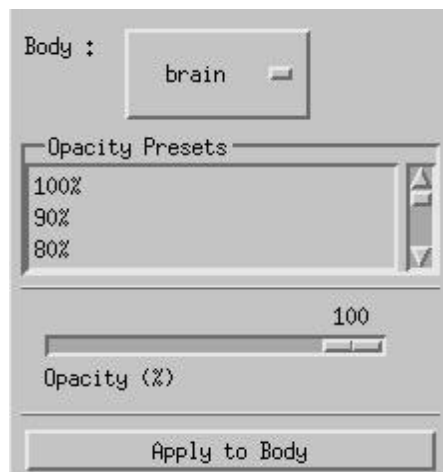


Figure 6.7. Layout of the Transparency panel.

The Transparency panel is very similar to the Color panel. The current body for which to set the transparency is selected in the Body menu. The opacity wanted for the current region is then selected by using either the Opacity Presets list or the Opacity slider. Then the Apply to Body button applies the opacity setting to the selected body.

6.4.4. Lighting Panel

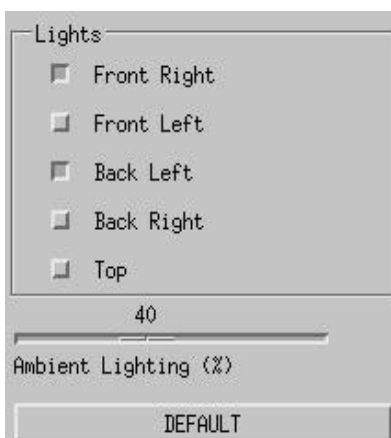


Figure 6.8. Layout of the Lighting panel.

The Lighting panel, shown in Figure 6.8, controls the lighting of the rendering. The Lighting panel will only activate when the polygon view style is used. Five lights are available in sera3d, and multiple lights may be illuminated simultaneously. These are listed in the Lights frame. Below the Lights frame is the Ambient slider. This sets the fraction of ambient light in the rendering. The Default button returns the lighting controls to their default positions.

6.4.5. Axes Panel

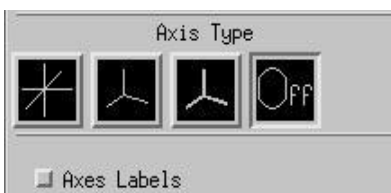


Figure 6.9. Layout of the Axes panel.

The Axes panel, shown in Figure 6.9, allows the user to inlay the axes into the rendering. The icons at the top represent the type of axes to inlay. The first axes are the full set: Inferior-Superior, Posterior-Anterior, and Right-Left. The second and third axes represent the positive quadrant, with the third icon providing a bold representation of the axes. Below the icons is the Axes Labels toggle. This toggles the labels of the axes in the rendering.

6.4.6. Clipping Panel

Sera3d provides two clipping planes in each orthogonal direction. A clipping plane is used to "cut" away portions of the rendered regions. The Clipping panel, illustrated in Figure 6.10, contains a Clipped Bodies list. The Clipped Bodies list shows all of the segmented regions for the current univel file. The buttons on the list toggle the clipping of a specific body by the current clipping planes. If a body is checked, it will be clipped; if it is not checked, no clipping planes will cut it. To set the clipping planes, bring up the clipping popup window, as seen in Figure 6.11, using the Set Clipping button.

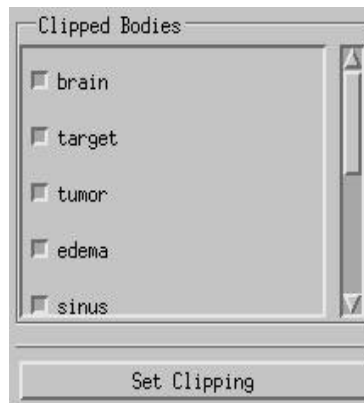


Figure 6.10. The Clipping panel layout.

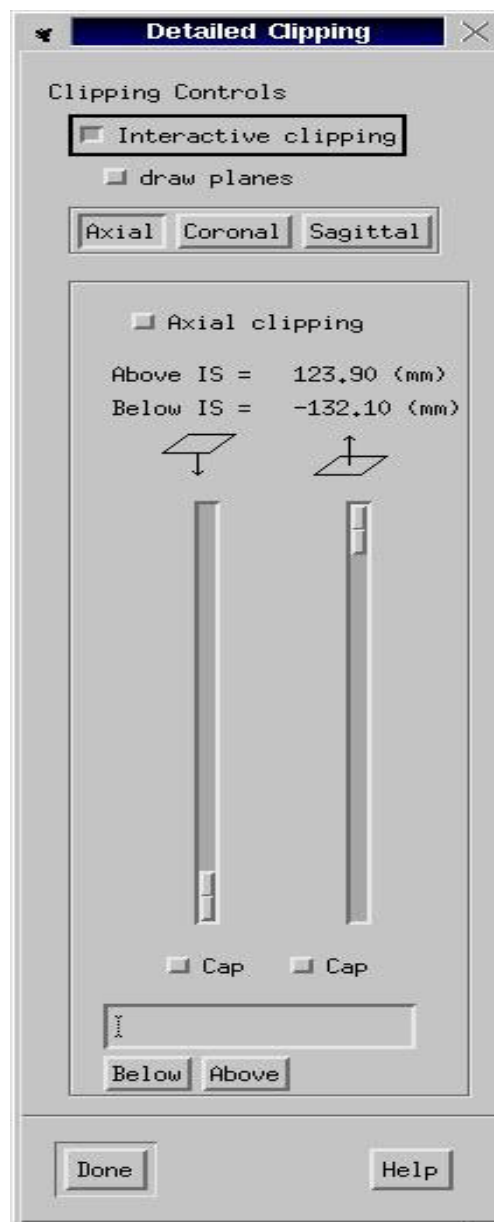


Figure 6.11. The Clipping popup widget.

The Clipping Popup provides the tools necessary for controlling the six clipping planes. At the top is the option to clip interactively. When clipping is performed interactively, as the sliders are moved, the clipping is updated in the main rendering window. At the same time, the draw planes toggle will display the clipping planes in the main rendering window.

The three clipping directions allowed are Axial, Coronal, and Sagittal. To clip in a particular direction, first select that direction from the three push buttons. The panel below will switch to the appropriate direction. At the top of the clipping panel, there is a toggle button, which is labeled with the selected clipping direction. In the example shown in Figure 6.11, this is labeled Axial Clipping. This button will toggle the clipping planes - i.e., it will activate the clipping process. Below the toggle are labels showing the current position of the two clipping planes. Each clipping plane has a corresponding slider. By moving the slider up or down, the label representing the same clipping plane is updated to show the position of the clipping plane. If the toggle button is in the depressed position, clipping will proceed continuously as the sliders are moved. There is also a text box available for typing in a specific clipping location. For example, to clip everything above 98.35 in the axial direction, type 98.35 into the text box, and press the Above button. This will set the slider to the appropriate position.

Below each slider is a toggle called Cap. If this toggle is selected, the corresponding clipping plane will be capped with the medical image data (if the images have been loaded).

Clipping in multiple directions is also supported. To clip in the sagittal direction as well as the axial direction, turn on and set up the axial direction clipping information, then press the Sagittal push button, enable the sagittal direction, and set the sliders to the desired position.

When clipping is performed interactively, the Done button simply closes the window. If clipping is performed non-interactively, the Done button will apply all the clipping settings to the rendering.

6.4.7. Polygon Panel



Figure 6.12. The Polygon panel layout.

The polygon panel, as shown in Figure 6.12, has only one menu. This menu controls which algorithm will be used when creating the polygonal surfaces for the polygonal view style. The options for the algorithm are:

- 8-Cell algorithm using surface normals - provides flat shaded polygons
- 8-Cell algorithm using vertex normals - provides smooth shaded surfaces
- Marching Cubes - uses the Marching Cubes algorithm to determine the polygonal surfaces, and produces more polygons than the 8-Cell algorithm, thus slowing the rendering, but increasing the fidelity.

6.4.8. Particle Tracking Panel

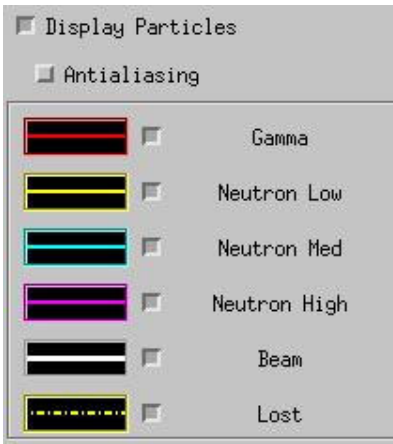


Figure 6.13. The Particle Tracking panel.

The particle tracking panel controls the rendering of the loaded particle tracks. The particle tracks are output from the seraMC calculations, and stored in the `lost.pp` or `tracks.pp` file. The Display Particles button toggles the display status for all particles in the rendering. The antialiasing button toggles the antialiasing mode. Antialiasing is used to smooth the particles tracks in the rendering.

Below the two toggles is the panel of particle track types. To the left of each type name is a toggle and a color box. The toggle controls the rendering of that particular type. The color box shows the current color and line style of the particle track type. Clicking on the color box will bring up a small dialog, allowing the user to change the color and line style.

6.4.9. Slice Panel

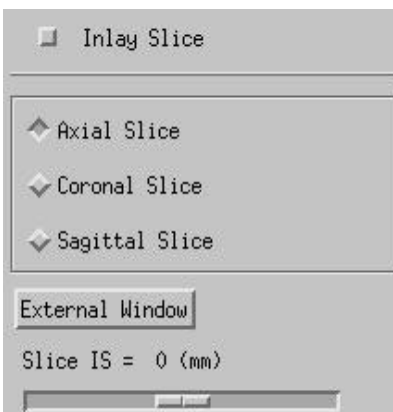


Figure 6.14. The Slice panel layout.

The slice panel, displayed in Figure 6.14, provides the tools for inlaying a single medical image slice into the rendering. To inlay medical data, the images must first be loaded from the file menu. Once the images are loaded, they are built into a texture map. From the texture map, the medical slices can be inlaid into the rendering.

The Inlay Slice toggle controls the display of the slice(s) in the rendering. The direction of the slice can be set by selecting it from the direction panel. The choices are Axial, Coronal, or Sagittal. The External Window button brings up an external window containing the 2-D slice that is currently inlaid. This option is currently unavailable for the Sun systems.

At the bottom of the Slice panel is the slice slider. This slider controls the position of the inlaid slice. Above the slider is the label reporting the slices' position. The slider is interactive. As the slider is moved, the slice is moved in the rendering, and the label displays the position of the slice.

6.4.10. Contour Panel

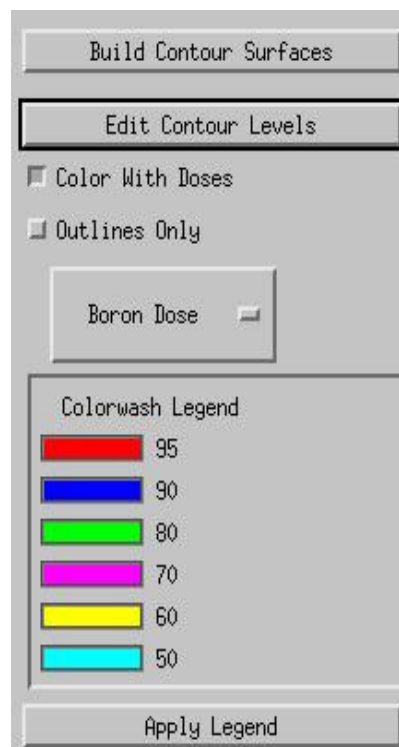


Figure 6.15. The Contour panel layout.

The contour panel, shown in Figure 6.15, is used to display the dose contours output by seraMC. The displaying of the dose contours is done in two ways. The first method is to colorwash the medical image data. With this method, the inlaid slices will be displayed with the contour rendered as colorwashed regions on the image slices. The Outlines Only button reduces the colorwash to outlines of the isodose contours, but this is only active when the Color With Doses button is also activated. The second method of displaying the dose contours is to build contour surfaces.

The dose contours must first be loaded before they can be displayed. They are loaded from the file menu; each contour data file may be loaded singly or the 3D contour data file may be loaded in one step. To set which contour levels to display, click on the Edit Contour Levels button to obtain a small popup window. Simply typing in the contour levels, separated by spaces, will update the legend for the new levels.

Below the Outlines Only toggle is the Dose Component menu. This determines which dose component of the contours will be displayed. These are the same components produced by the seraMC dose calculation. The Colorwash Legend shows the current levels to view, as well as their corresponding colors. The color for a particular level can be changed by clicking on the corresponding color box. Finally, the Apply Legend button will colorwash the slices with the current settings.

To build the contour surfaces, press the Build Contour Surfaces button at the top of the Contour panel. The surfaces are built according to the levels and colors specified in the legend. Once the surfaces are built, they are treated as additional bodies. They will be listed in the Bodies List at the top of the Tool bar. They will also appear in the bodies menu of the color and transparency panels.

6.4.11. Mouse Panel

The layout of the Mouse panel is shown below in Figure 6.16.



Figure 6.16. The Mouse panel layout.

The two icons at the top of the panel switch the rotation control method between the mouse buttons and sliders. The left icon is for the sliders, and the right is for the mouse. When the slider rotation is activated, the sliders will appear below the main rendering window.

The rotational locks are used to prevent the bodies or clip planes from being rotated. For example, the bodies can be locked, which means that they cannot be rotated. If a clipping plane is inlaid, the clipping will rotate around the locked bodies, cutting the bodies as it rotates.

6.4.12. View Parameters Panel

The View Parameters panel is shown below in Figure 6.17.

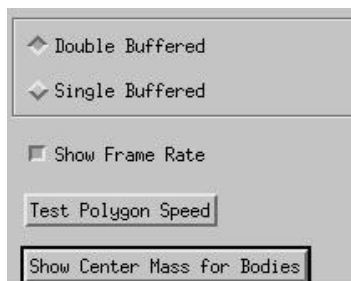


Figure 6.17. The View Parameters panel.

The frame at the top switches between double buffering and single buffering. With double buffering, the rendering is done offscreen, and then the rendering frames are swapped. This provides smooth rotations without any flashing. With single buffering, the rendering is done directly to the screen. When rotating in single buffer mode, the screen flashes as it re-renders. Single buffering is handy for watching the rendering. For example, when volume rendering, it is beneficial to be in single buffered mode. This allows the reconstruction process of stacking slices to be viewed interactively.

The second item is the toggle for the Frame Rate. When toggled on, the frame rate is displayed in the bottom left corner of the main rendering window. The Test Polygon Speed button runs a small test to give an approximation of the rendering speed of your system. The Show Center Mass for Bodies will display the center of mass for each of the segmented regions in the univel file.

6.4.13. Beam Panel

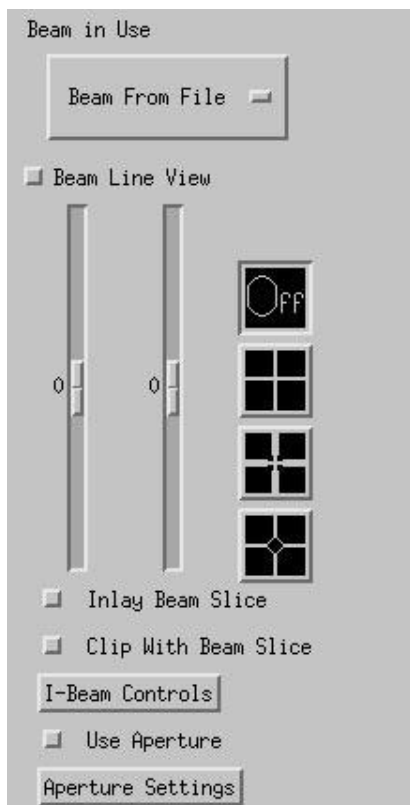


Figure 6.18a. The Beam panel.

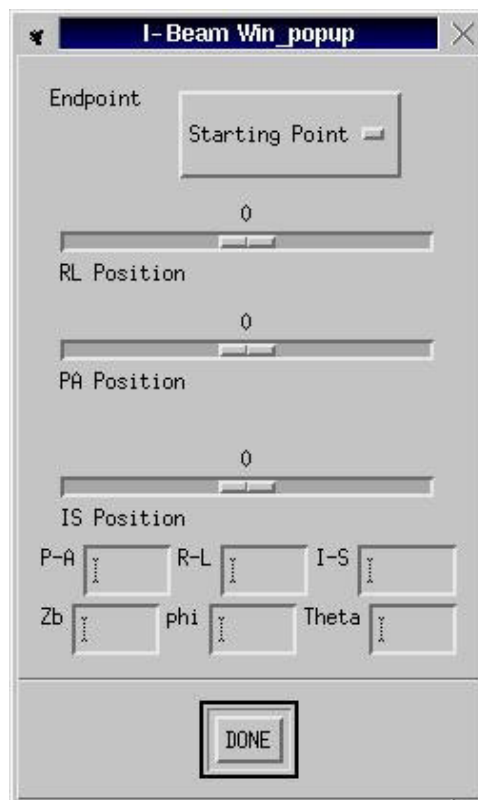


Figure 6.18b. The Interactive Beam Control Dialog.

The Beam panel, shown in Figure 6.18a, is used as an aid to position the beam/patient, or simply to view along the beam specified in the particle track file. The Beam in Use menu at the top switches between the beam from the particle track file and the interactive beam. To switch the view in the main rendering window to a beam line view, toggle on the Beam Line View button. The icons on the right side of the panel provide several different crosshairs for use in viewing along the beam line.

The two sliders below the Beam Line View panel control the position along the beam line. The left slider will slide the camera view forward or backwards along the beam line. The right slider is used for the beam slice. A beam slice is an inlaid medical image slice normal to the beam. To inlay a beam slice, toggle on the Inlay Beam Slice toggle button. The right slider then moves the position of the beam slice along the beam line. The Clip With Beam Slice will clip the regions at the position of the beam slice.

To control the position of the interactive beam, the I-Beam Controls button brings up a dialog like the one shown in Figure 6.18b. The position of the interactive beam can be set using the three sliders, or the values can be entered into the text boxes labeled P-A, R-L, and I-S. As the position of the beam changes, the values in the text boxes labeled Zb, phi, and Theta are also updated. These values correspond to the parameters in the seraCalc interface with the same names. The interactive beam can also be set by supplying these values directly.

When Beam Line View is toggled on, the option Use Aperture can be selected. This provides a small ring in the center of the main rendering window, which gives a measurable ring for examining the regions through a given beam diameter. When Use Aperture is enabled, the rendering switches to a parallel projection, allowing for accurate measuring with the ring aperture. Because of the parallel projection, the position of the slider along the beam will not affect the rendering. The sizes of the aperture and rendering window can be set with the Aperture Settings button. When Use Aperture is enabled, the aperture and window sizes are displayed in the top left corner of the main rendering window.

6.4.14. Texture Mapping Panel

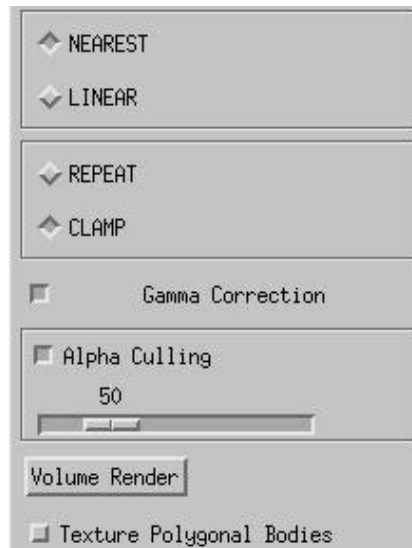


Figure 6.19. The Texture Mapping panel layout.

The Texture Mapping Panel provides the tools for controlling the texture mapping. The first panel determines how the texture will be applied to the polygons. Nearest will do a direct mapping from the nearest value in the texture map. This is the faster method, but it will produce a blocky appearance between the slices. Linear will linearly interpolate between the eight nearest neighbors in the texture map, providing a smoother slice.

The second panel determines what to do with the polygons that extend out of the texture map. With the Repeat option, the texture will repeat for polygons outside of the texture map. The Clamp option will clamp at the border value for the polygons outside of the texture map.

The Gamma Correction toggle will apply a gamma correction function to the gray values of the images, brightening them. The Alpha Culling button toggles the alpha culling. Alpha culling is used to remove gray levels from the polygons. For example, if a slice is inlaid without alpha culling, the whole slice, including the buffer, will be inlaid into the rendering. If Alpha Culling is enabled, the buffer (i.e., dark values) will be removed, leaving just the image itself. The slider under the Alpha Culling toggle sets the gray values to be culled. The values culled are less than the current slider value.

Volume Rendering gives the ability to reconstruct 3D structure directly from the medical images. To volume render, simply press the Volume Rendering button. At the bottom of the panel is the Texture Polygonal Bodies toggle. When the viewing style is polygonal, this button will toggle the texture mapping of the surfaces. This is useful when the colorwashed dose contours are loaded; texturing the surfaces will colorwash the regional surfaces with the dose contours. An example would be viewing the surfaces of the tumor and being able to see the 95% dose level directly on the surface.

6.5. The Preferences

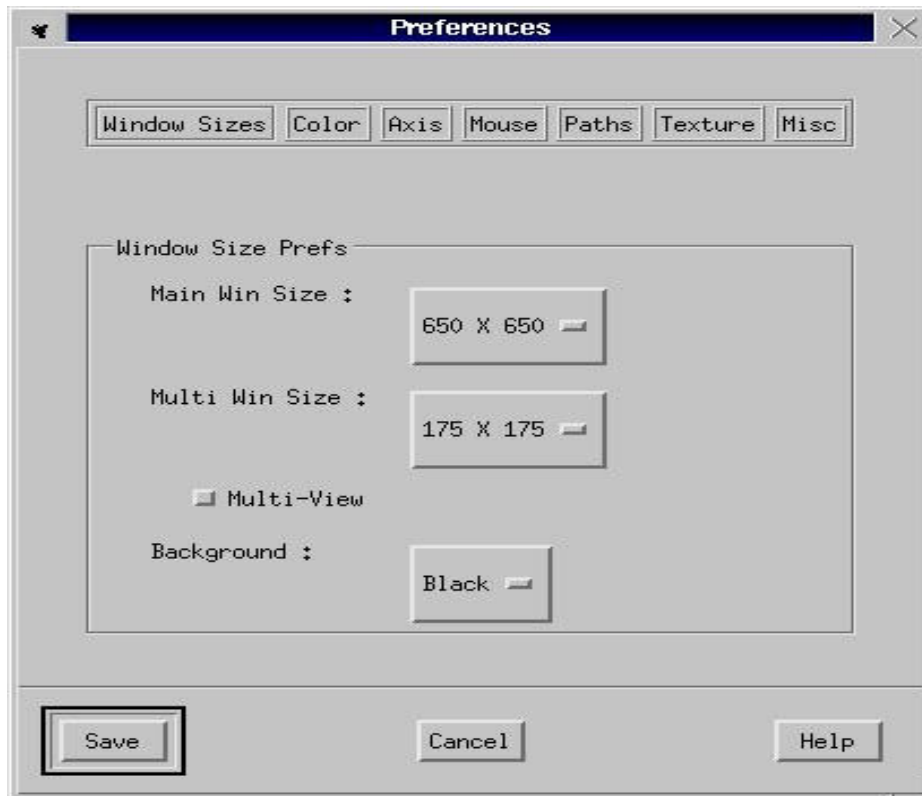


Figure 6.20. The Preferences popup window.

The Preferences window, shown in Figure 6.20, is selected under Preferences in the Menubar. This window is used to set the commonly used items in the program to suit the user's

preferences. When the preference window is opened, all of the items in the window are set to their current settings in the program. Once the preferences are saved (using the Save button), these become the defaults for the program. When the program starts, it sets all its parameters to the user specified preferences. Note that changing the values will only apply to the next invocation of sera3d, not to the present session.

The preferences are broken into separate panels.

6.5.1. Window Sizes

The window size preference panel can be seen above (Figure 6.20). It has the settings for the window sizes and background color, which also are found in the Options menu.

6.5.2. Color

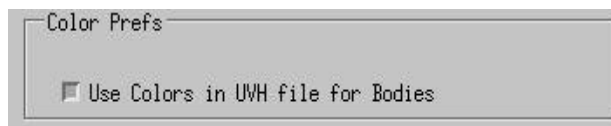


Figure 6.21. The Color Preference panel.

The color preference panel, shown in Figure 6.21, has a toggle for the body coloring information. The colors for the regions are stored in the .uvh file. To use the stored colors, simply check this toggle. If preferred, the user can specify default colors for the regions. Uncheck the toggle and the color presets for each body will be used.

6.5.3. Axes

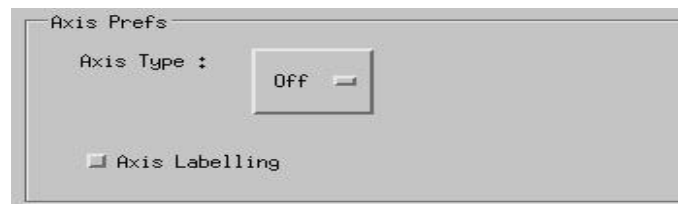


Figure 6.22. The Axis Preference panel.

The axis preference panel, seen in Figure 6.22, is the same as the Axes Control panel in the main interface.

6.5.4. Mouse



Figure 6.23. The Mouse Preference panel.

The Mouse preference panel (see Figure 6.23) sets the preferred rotation method (sliders or mouse). This has the same function as the Mouse panel in the main interface.

6.5.5. Particle Paths

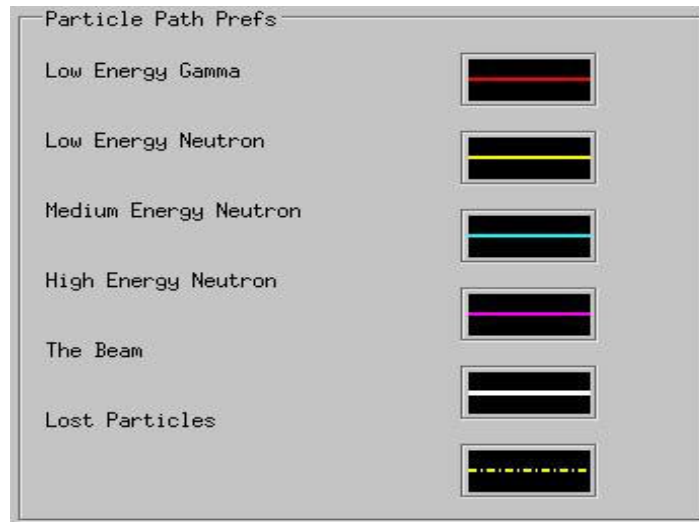


Figure 6.24. The Particle Paths Preference panel.

The Particle Paths preference panel (see Figure 6.24) sets the preferred line types and colors for the particle tracks.

6.5.6. Texture

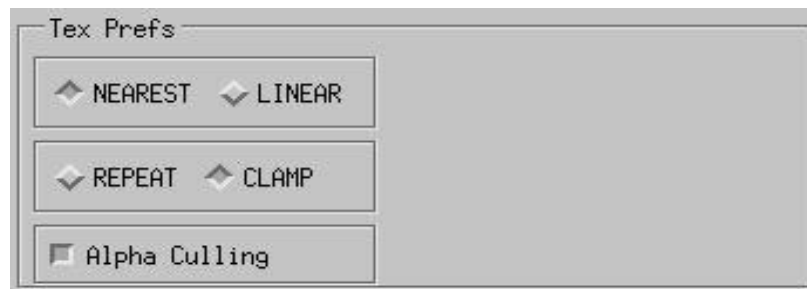


Figure 6.25. The Texture Preference panel.

The Texture preference panel, shown in Figure 6.25, allows the preferred texturing parameters to be set.

6.5.7. Miscellaneous

The Miscellaneous preference panel, shown in Figure 6.26, has a variety of items. The Rendering Quality panel is used to set the quality of the first three view styles (wireframe, solid outline, and solid rendering). In general, as the quality of the rendering is lowered, the speed of the rendering improves. When running on machines without hardware acceleration, it is better to use a lower quality.

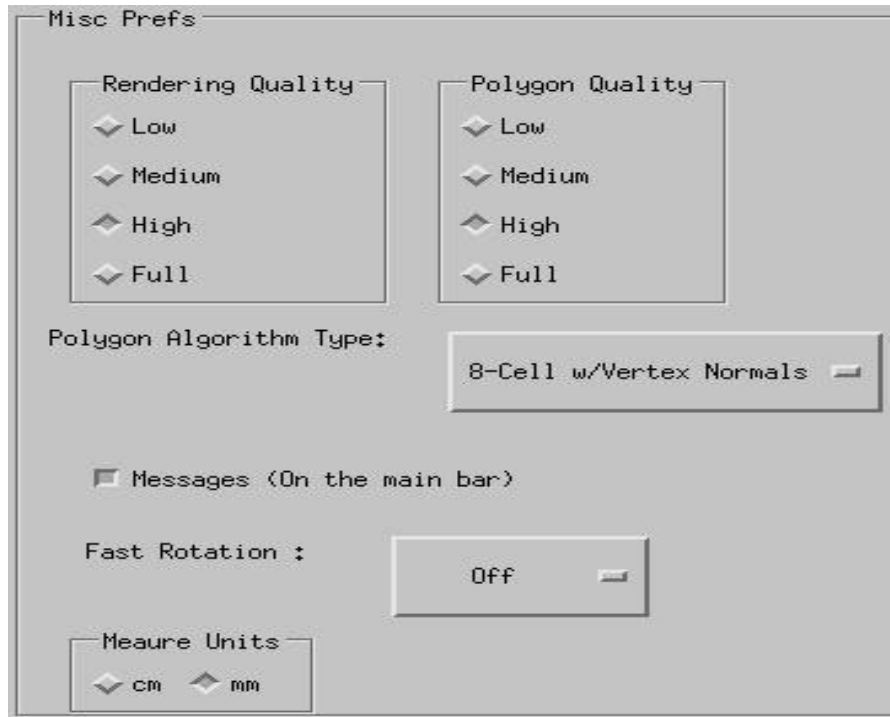


Figure 6.26. The Miscellaneous Preference panel.

The Polygon Quality is very similar to the Rendering Quality, but this applies to the polygon surface view style. Again, without hardware acceleration, a lower polygon quality should be used. This lowers the number of polygons created for the surface representation. The Polygon Algorithm type is the same menu as on the Polygon Control Panel.

The Messages toggle controls the display of the message bar at startup. The Fast Rotation menu is the same as in the Options menu. The Measure Units box sets the units, either cm or mm, for all measurements in sera3d.

7. Transport Modeling

7.1. Introduction

The seraMC interface for transport modeling, seraCalc, provides an interactive input interface for the seraMC radiation transport module. This interface allows the user to manipulate the various input parameters and edit directives for seraMC, to create an input file for a specific calculation. Existing input files may be read in and modified, or the input may be created from an essentially blank template. A sample configuration for the seraCalc interface is shown in Figure 7.1. The following sections provide a brief explanation of the various input fields in the interface, with links to more detailed descriptions of the input parameters in the appendices.

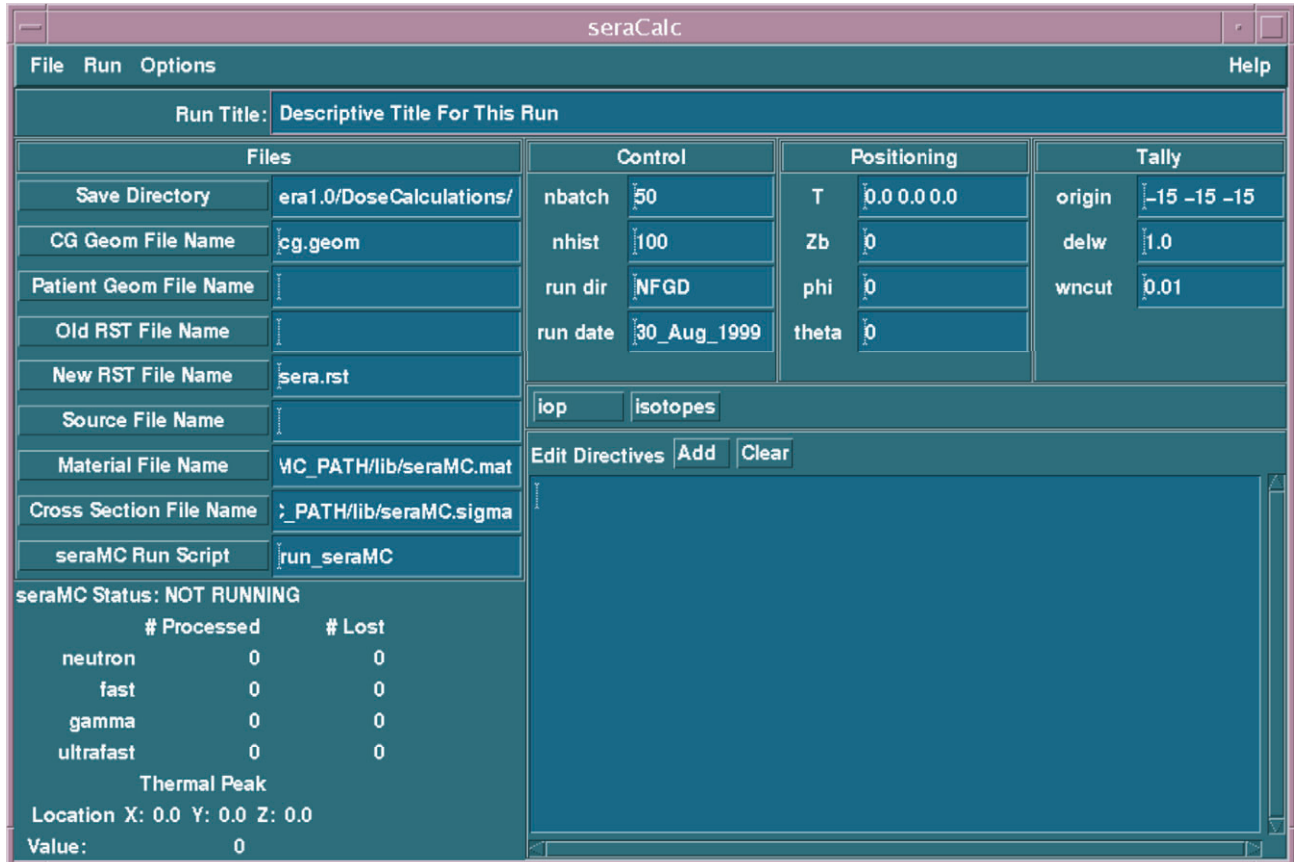


Figure 7.1. Sample seraCalc interface display. This is the default configuration at startup.

7.2 Menus

7.2.1. File Menu

To start the creation of a seraMC input file, the user should select the name of the input file. By default, a set of standard values is displayed in the interface window, as in Figure 7.1, with Save Directory assigned the directory from which seraCalc was invoked. Selecting Open from the File menu, as shown in Figure 7.2, allows the user to start with values from a previous session. The Save and Save As menu buttons allow the modified input data to be saved either to the loaded file, or to a new file as chosen in a file selection widget.

The Launch button allows other sera modules to be executed from within seraCalc. The Check Version entry is no longer functional.

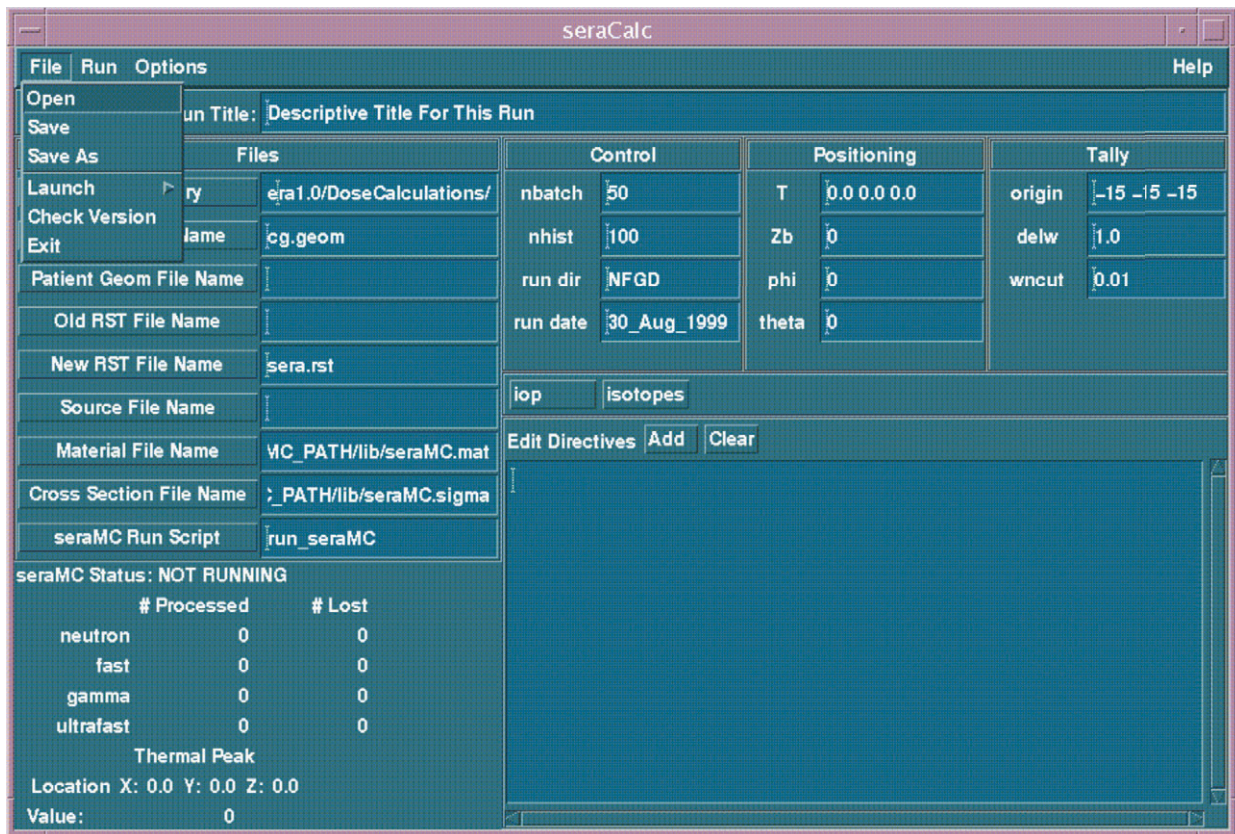


Figure 7.2. File pulldown menu.

7.2.2. Run Menu

The Run menu in the top left corner of the seraCalc window has three selections, as shown in Figure 7.3. The Run button is used to initiate a run of the seraMC program using the newly generated input file. This option invokes the script file given in the seraMC Run Script field (see Section 7.4.9), with the input file specified by either the Save or Save As options in the File menu (see Section 7.2.1). The Test button is used to run a short trial of the seraMC program. This causes a short seraMC run to be made for checkout. For this run, particle tracks are written to a file named input.pp, where input is the name of the input file used for the run. These tracks can be viewed in sera3d along with the three-dimensional model for further model verification. The Kill button is used to stop the currently running seraMC program.

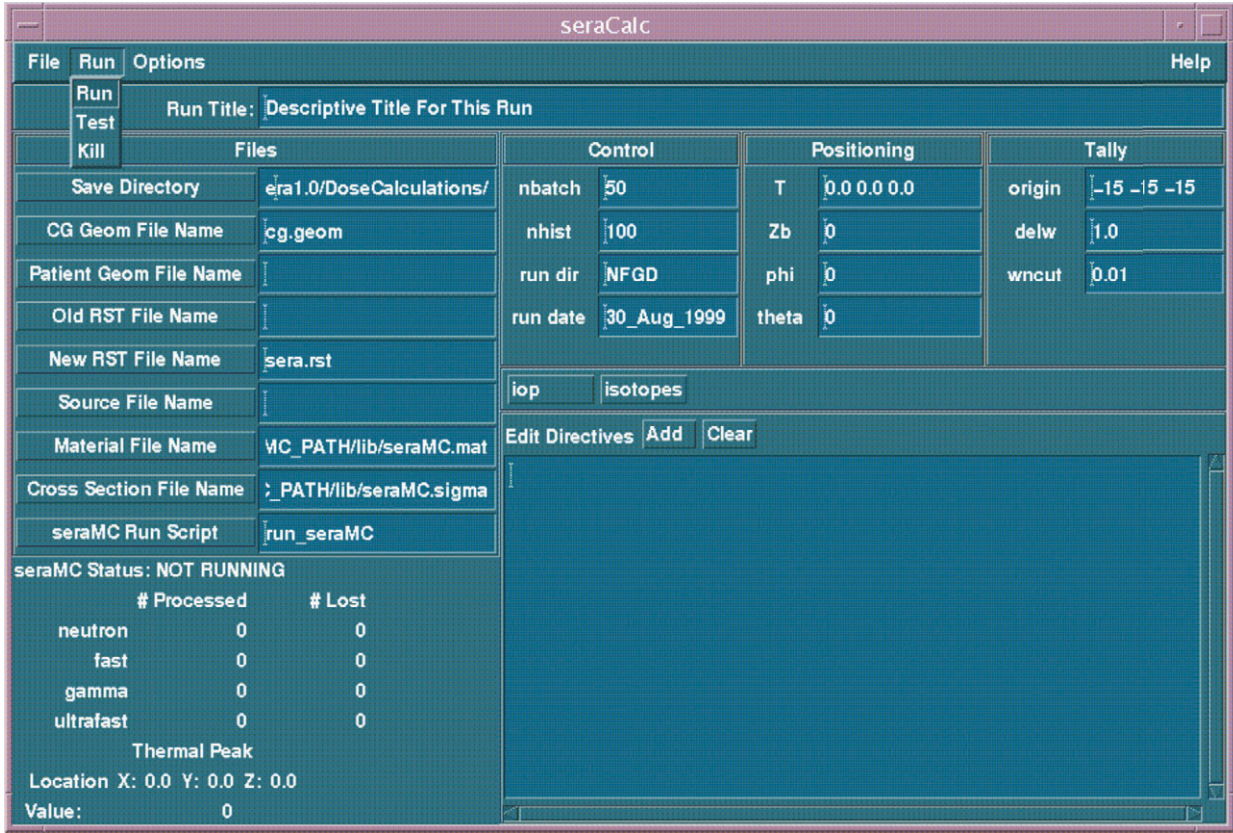


Figure 7.3. Run pull-down menu.

7.2.3. Options Menu

The Options menu has only one selection, as shown in Figure 7.4. This option toggles the automatic display of the three axial views of the source-patient configuration generated by seraMC. These views are saved in the three raster files view.1, view.2, and view.3.

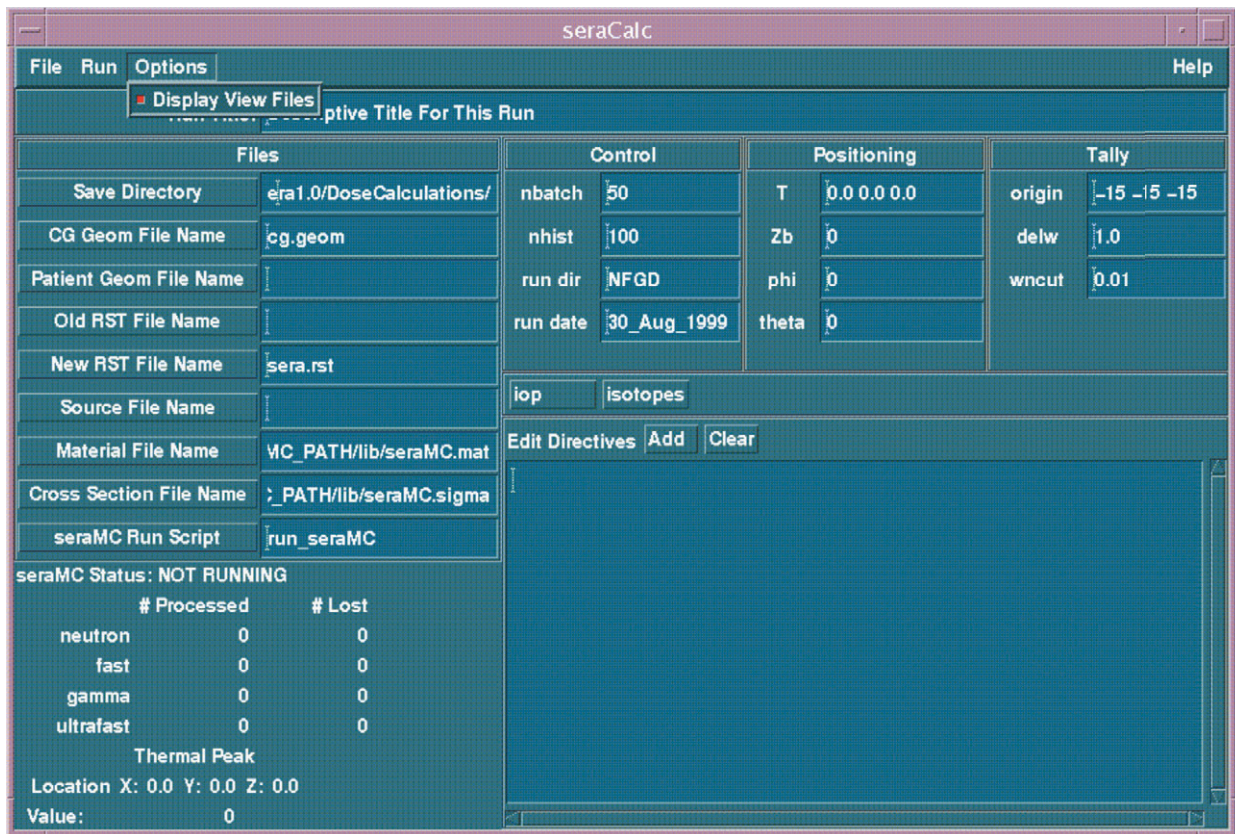


Figure 7.4. Options pulldown menu.

7.3. Run Title Panel

The title for the seraMC run is entered in the field at the top of the panel labeled Run Title: (see Figure 7.5). This identifies the particular data set. As for other data entry fields, the user can edit this field by positioning the text cursor anywhere in the input area and entering new characters. The entire line also can be highlighted and deleted or replaced.



Figure 7.5. Run Title.

7.4. Files Panel

All of the fields on the panel work in a fashion similar to Run Title, with the exception of the fields in the subpanel labeled Files in Figure 7.6. This corresponds to the upper left quadrant of the seraMC interface.

Each of these selections consists of a push button with a label and a text entry field to the right of the button. Entering a value can be performed using one of two methods. The first method consists of the user selecting the text entry field by moving the cursor to the area and clicking the left mouse button. The value for the field is entered using the keyboard. In this instance, environmental variables (e.g., \$SERAMC), relative paths (e.g., ./filename), and home directory shortcuts (e.g., ~user_id) may be employed, as these are expanded in the seraMC code. The second method consists of left clicking on the button with the label. A file selection popup

appears and the user can select the name of an existing file. When the OK button of the file selection popup is pressed, the name chosen is entered into the text entry field. Right clicking on this button will display the full file path, with environment variables fully translated, for long file names.

Files	
Save Directory	era1.0/DoseCalculations/
CG Geom File Name	cg.geom
Patient Geom File Name	
Old RST File Name	
New RST File Name	sera.rst
Source File Name	
Material File Name	MC_PATH/lib/seraMC.mat
Cross Section File Name	MC_PATH/lib/seraMC.sigma
seraMC Run Script	run_seraMC

Figure 7.6. Files Panel.

7.4.1. Save Directory

The Save Directory field specifies the name of the working directory to be used for the seraMC calculation. It is to this directory that all the files created by the seraMC calculation will be written. A copy of the input file, with the same name chosen for the Save and/or Save As buttons, will also be written to this directory.

7.4.2. CG Geom File Name

This file defines the combinatorial geometry (CG) space for the problem. For cases where a patient geometry (image space) is specified, the CG space is used for the source region and the world outside the image space. The source geometry is defined in the source file (see Section 7.4.6 and Appendix C). However, it is permitted to have cases where only a CG space is specified. This is generally done for phantom irradiations, or where the geometry may be simply defined in terms of intersections and/or unions of solid geometric shapes. A CG file is always required. Additional details on CG file construction and the interaction of CG and univel geometry files are available in the appendices.

7.4.3. Patient Geom File Name

The patient geometry file is used to define the geometry of the patient in image space. This will be a univel file constructed by seraModel from the MRI, CT, or PET images. For cases where the entire geometry is described by the CG file, a patient geometry file is not required.

7.4.4. Old RST File Name

The old restart file provides a restart capability for seraMC. Specification of a file name here causes seraMC to read in the dose information from this file, and continue the calculation from this starting point. Note, however, that this restart ability only allows the user to run additional run directives, not to run additional particles for a previously calculation. For example, if a seraMC case was run for 1,000,000 neutron histories, and this was deemed insufficient for convergence, the restart capability will not allow the user to run an additional 1,000,000 histories for a total of 2,000,000 histories. It will allow the user to run only the neutron histories, then later run the biased fast neutron and gamma components, or to run the neutron and gamma components, and edit later. For calculations that do not utilize the restart capability, the old restart file should be specified as none. For further details, please see the run_dir section below and in the appendices.

7.4.5. New RST File Name

The new restart file is the restart file to be created by the present seraMC calculation. One restart file is created for each run directive specified, except the edit directive. The first file is named filename, the second is filenameea, the third is filenameeb, and so on. Since each of these later versions contains the information from the previous file, the older files are deleted as the newer ones are created. Edit only calculations should specify none for the new restart file. For additional information on the contents of the restart files, see the appendices.

7.4.6. Source File Name

The source file provides the coupled energy-angle description of the neutron and gamma source from a particular facility, as well as the CG description of the source and collimator geometry. A complete description of the source file contents and formats is available in the appendices.

7.4.7. Material File Name

The material file provides nuclide density composition specifications for a variety of materials useful for BNCT and other forms of clinical radiation therapy. Cross sections to access from the seraMC cross section file are given for each nuclide.

7.4.8. Cross Section File Name

The cross section file contains the neutron and gamma cross section data for the set of nuclides required by the material file. Neutron and gamma production data are available at energies up to 16.9 MeV and gamma interaction data are available to 100 MeV. Details on this file are available in the appendices, with a list of available nuclides given in the edit isotopes section.

7.4.9. seraMC Run Script

The seraMC run script is a simple Unix shell script that invokes the seraMC transport module and performs some file management activities. The supplied default script file (in \$SERA_HOME/Target/bin) is usually sufficient for most users' purposes.

7.4.10. Ultrafast Mode Files

There are three "hidden" file inputs not shown above. These files concern the high-energy neutron calculations available for analysis of fast neutron therapy facilities. These text fields only become visible when the U and/or P run directives are entered (see Section 7.5.3). The three files specify the proton range data file, the ultrafast directory file, and the ultrafast cross section file.

The proton range data file gives tabulated range-energy data for protons in a variety of materials. This is used to compute the proton recoil dose from scattering of high-energy (>16.9 MeV) neutrons on hydrogen. A general description of the proton dose computation is provided.

The ultrafast directory file provides information that seraMC requires to read the ultrafast cross section file. The cross section file contains a limited subset of the materials in the seraMC cross section file, for energies between 16 and 100 MeV. Further details are provided in the appendices.

7.5. Control Panel

The control panel specifies the number of particle histories to simulate and the run mode(s). It is shown below in Figure 7.7. This panel has four fields, which are text entry fields. The user chooses a field for which a change is to take place and enters new values in the field.

Control	
nbatch	50
nhist	100
run dir	NFGD
run date	23_Apr_1999

Figure 7.7. Control Panel.

7.5.1. Nbatch

The nbatch input parameter specifies the number of statistically independent batches of neutron histories to run in seraMC.

7.5.2. Nhist

The nhist input parameter specifies the number of neutron histories to run in each batch. There is presently a limit of 2000 histories per batch.

7.5.3. Run Directives

The run directives determine which particular calculation modes are to be invoked in seraMC. The valid directives are N, F, G, D, T, U, and P. The valid combinations of these directives are NFGD, NFD, NGD, FGD, UPGD, UPD, UGD, PGD, and GD.

7.5.4. Run Date

The run date allows the user to specify a date for identification purposes. This date is not used in any computation, merely to allow the user to retain information regarding the construction of the treatment plan, or to identify the treatment date. The date is limited to 15 characters, and it must be provided in the form of a character string, with no embedded blanks or other white spaces. The default is the date of invocation of seraCalc, given in the form DD_MMM_YYYY, where the month designation is the abbreviation using three alphabetic characters.

7.6. Positioning Panel

The positioning panel specifies the beam orientation for this case. It is shown below in Figure 7.8. This panel has four fields that are text entry fields. Any changes may be entered as described above. These four fields provide a complete description of the beam orientation for a particular irradiation setup.

Positioning	
T	0.0 0.0 0.0
Zb	0
phi	0
theta	0

Figure 7.8. Positioning Panel.

7.6.1. Target Point Location

The target point location T is given in terms of the model coordinate system, which is the coordinate system used in seraMC. This provides the (x,y,z) coordinates of the point where the center of the source beam is aimed.

7.6.2. Distance to Target Point

The distance to the target point, Zb, is the straight-line distance from the source plane to the target point. This parameter should be set sufficiently large to prevent overlap between the patient and the source/collimator assembly. This distance is given in cm.

7.6.3. Polar Angle

The polar angle, phi, is the angle in degrees between the superior axis and the desired beam location. Positive values are associated with rotation towards the right side of the patient.

7.6.4. Azimuthal Angle

The azimuthal angle, theta, is the angle in degrees between the right axis and the desired beam location. Positive angles are associated with rotation towards the front of the patient.

7.7. Tally Panel

The tally panel specifies the overlying 3D edit mesh. It is shown in Figure 7.9. This panel has three fields, which are text entry fields. Any changes may be entered as previously described.

7.7.1. Origin

The origin parameter specifies the (x,y,z) coordinate location of the center of the edit mesh system. This superimposes a 30x30x30 cubic mesh over the 3D image system, centered on the designated origin point. This edit mesh is used for all dose and flux tallies in the seraMC calculation.



Figure 7.9. Tally Panel.

7.7.2. Delw

The delw parameter specifies the size, in cm, of each mesh element in the 3D edit mesh. All mesh elements are regular cubes, so only one dimension need be provided.

7.7.3. Wncut

The wncut parameter designates the minimum weight allowed for neutrons in the particle history simulation. When a particle weight falls below this value, a binary decision is made. The particle weight may be doubled and tracking continued, or the particle history may be terminated. The initial particle weight is specified in the source routines.

7.8. Auxilliary Panels

There are two buttons below the three panels just described (see Figure 7.10). These activate popups for special operations affecting the transport run. Pressing a button will cause its popup to appear. Each of the popups has text entry fields with default data (Figures 7.11 and 7.12). The user can change these data and then press the Save button on the popup to put it away. The Reset button on the popups causes the default values to be reinstated for that popup. The names on these buttons are iop and isotopes.



Figure 7.10. Auxiliary Panel.

7.8.1. Beam Optimization and Automatic Positioning

The beam optimization and automatic positioning popup is displayed by pressing the iop button, as shown in Figure 7.11.

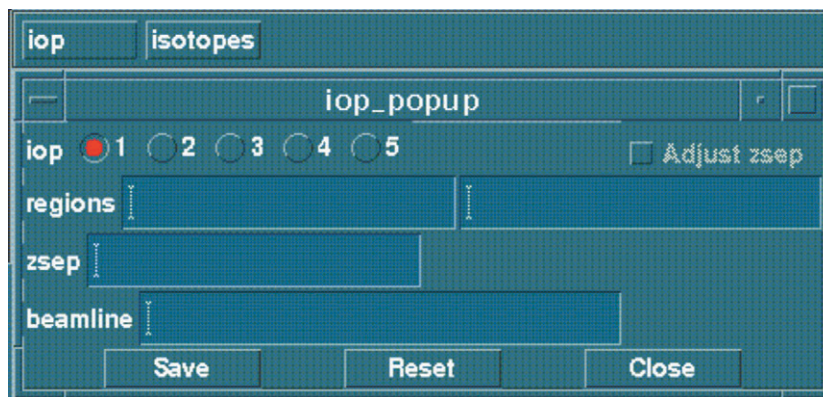


Figure 7.11. Beam optimization and positioning input panel.

There are five options for the beam positioning search, selected by the four buttons at the top of the popup. The default positioning is $iop=1$, which uses the values for T , Z_b , ϕ , and θ to position the beam.

For $iop=2$, all the parameters in the popup (regions, zsep, and beamline) must be specified. The Regions option designates the regions encountered between the patient and the source plane. Zsep gives the minimum distance that must be maintained between the skin surface and the source aperture. Beamline is a point that the beam centerline is forced to intersect between the center of the source plane and the target point. These values will allow seraMC to determine the beam positioning by calculating new values for Z_b , ϕ , and θ .

For $iop=3$, only the regions and zsep parameters need be provided. The positioning search determines ϕ and θ such that the distance from the skin surface to the target point is a minimum.

For $iop=4$, the input is similar to that for $iop=3$. The search fixes the beam angles at the specified values, and sets the separation distance, zsep, between the skin surface and the source aperture.

For $iop=5$, the input is similar to that for $iop=3$. The positioning search uses the θ angle as specified in the input and finds ϕ such that the distance from the skin surface to the target point is a minimum.

The Adjust zsep button activates an option to adjust the distance zsep so that it is a minimum but causes no geometric interferences between the collimator structures and the patient.

7.8.2. Edit Isotopes

The edit isotopes popup is displayed by pressing the isotopes button, as shown in Figure 7.12.

isotopes_popup		
Edit Isotopes		
Boron-10	521	6.01426E-8
Hydrogen	508	6.315E-2
Nitrogen	513	7.911E-4
Carbon	512	7.270E-3
Oxygen	505	2.680E-2
Edit isotope 1	0	0.0
Edit isotope 2	0	0.0
Save	Reset	Cancel

Figure 7.12. Isotope density input panel.

The edit isotopes popup allows the user to specify the composition of a particular material, which will be used for the dose edits. Generally, this is brain tissue, as shown in the default values, but any material for which a dose calculation is required may be used. The compositions are specified in atoms/barn-cm, and the list of edit isotopes is provided in the seraMC manual. Any isotopes may be requested, with the caution that the edits will always be titled as in the popup, i.e., boron dose, etc. It thus falls to the user to account for any deviations from human tissue in the edit isotopes specifications.

In addition, two edit isotopes may also be specified, which will be used to calculate reaction rate edits for arbitrary nuclides. Any nuclide provided in the seraMC material file may be used for these edits, and infinite-dilute reaction rates may be computed by supplying a very small density (usually 1.0E-24). For further details on available nuclides, please consult the seraMC manual.

7.9. Edit Directives

Note: The seraMC edit capability has been transferred to the seraPlan module and these options should only be used to perform edits for CG calculations. This section is provided for compatibility with input files for previous versions of seraCalc/seraMC.

Edit directives for the input file are entered in the large text widget shown in Figure 7.13, which corresponds to the lower right corner of the widget. An edit directive is chosen from the directives popup that appears when the Edit Directives Add button is pressed. The edit directives are interpreted after the transport runs are completed when a D mode run is specified.



Figure 7.13. Edit directives display.

These edit directives may be passive (setting variables, etc.) or active (specifying that a particular type of edit be performed). When an active edit directive is encountered during the seraMC run, the code performs the edit. After the active edit directive line, the user may change the variables and perform as many additional edits as desired. When the user selects the edit from the box shown in Figure 7.14, a line is added to the edit window showing the number and type of input words required as well as some defaults. In general, the user will change these values to those required for this problem. When a protocol is established, many of the edit directives are very similar, and those loaded from a previous case may need only slight modification for the current case. All entries are cleared when the Edit Directives Clear button is pressed.

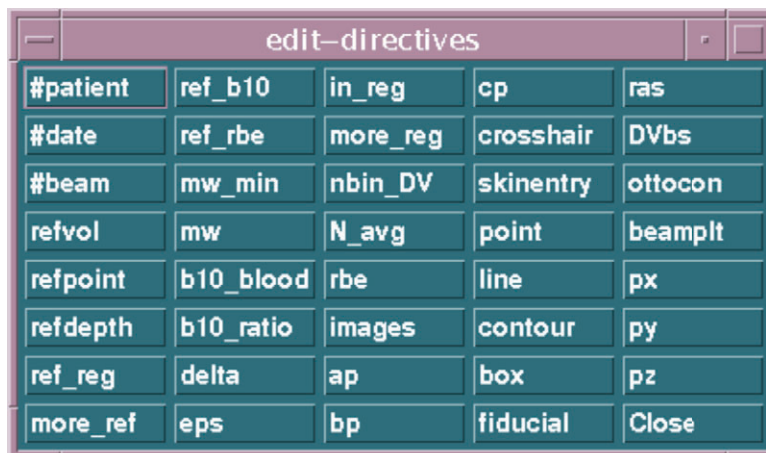
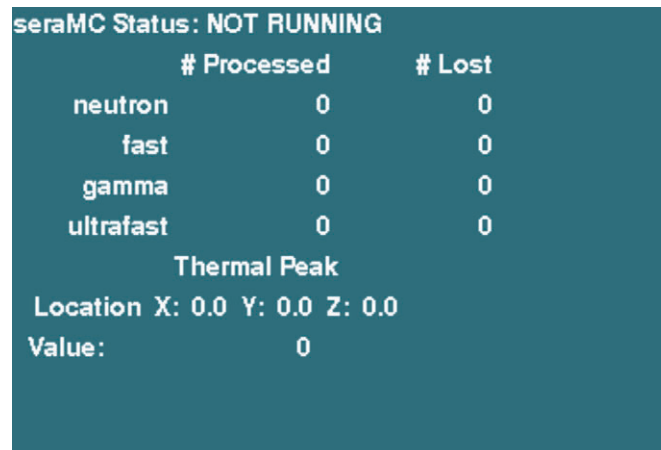


Figure 7.14. Edit directives creation widget.

The user can press as many of these directive buttons as desired before pressing the Close button to put the popup away. As each button is pressed, the appropriate edit directive template is entered into the text widget. The user can then use standard editing techniques to enter the desired values for the specific directive. Detailed descriptions of the edit directives are provided in the appendices.

7.10. Monitor Panel

The panel at the lower left is used to display running statistics for the seraMC calculation (see Figure 7.15). As the seraMC program and this program exchange data, the appropriate values are displayed in this panel. This is accomplished via a file called sera.mon, which is written into the seraMC working directory. The file is updated after each batch of particles and the peak location and value are written at the end of the run.



```
seraMC Status: NOT RUNNING
      # Processed      # Lost
neutron           0           0
      fast            0           0
      gamma           0           0
      ultrafast       0           0
      Thermal Peak
Location X: 0.0 Y: 0.0 Z: 0.0
Value:           0
```

Figure 7.15. Monitor Panel.

8. Statistical Dose Combination and Dose Editing

8.1. Introduction

Following the calculation of the dose results with seraCalc and seraMC for each field, it is necessary to properly combine the individual field dose results into an appropriate plan. This is accomplished by using the seraPlan tool. SeraPlan combines the dose results from up to 24 fields (maximum of four fractions, each with up to six fields), with source strength, boron, and exposure weighting, to provide an editable seraMC restart (.rst) file containing the weighted, normalized dose results. A plan file is also written, which contains the weighting data for each field, along with fraction and plan totals for boron and exposure. An existing plan file may be loaded from the File pulldown menu in seraPlan.

The dose combination is performed by weighting each specified field dose component with the appropriate weighting factors. All dose components are weighted by the source strength and exposure. The boron dose weight factor also includes the boron concentration, and the gamma dose weight factor includes the total gamma source for each field, which consists of both the source gamma and the capture gamma source from neutron capture in tissue, and the gamma repair factor (see Section 8.3.1.3.4) from the input display. The total dose is the sum of all the dose components. In all cases, the doses are normalized to unit exposure, unit source strength, and unit boron concentration. This allows the user to examine the effects of total plan exposure and average boron on the dose results, using the seraDose, sera3d, and seraPlot modules.

Dose editing allows the user to compute a variety of meaningful parameters to characterize the treatment plan. Doses may be computed at a point, along a line, as contours on a plane, or they may be integrated over a volume. Boron concentration and RBE values may be incorporated into these edits in several different ways. In addition to the extensive standard edits, the user may specify up to 20 additional point edits, 10 line edits, 10 volume edits, and 4 plane (contour) edits.

As with the other modules in the SERA system, seraPlan may be executed from the Main Menu, cross launched from any other module, or run from the command line by typing seraPlan. A sample seraPlan display is shown in Figure 8.1.

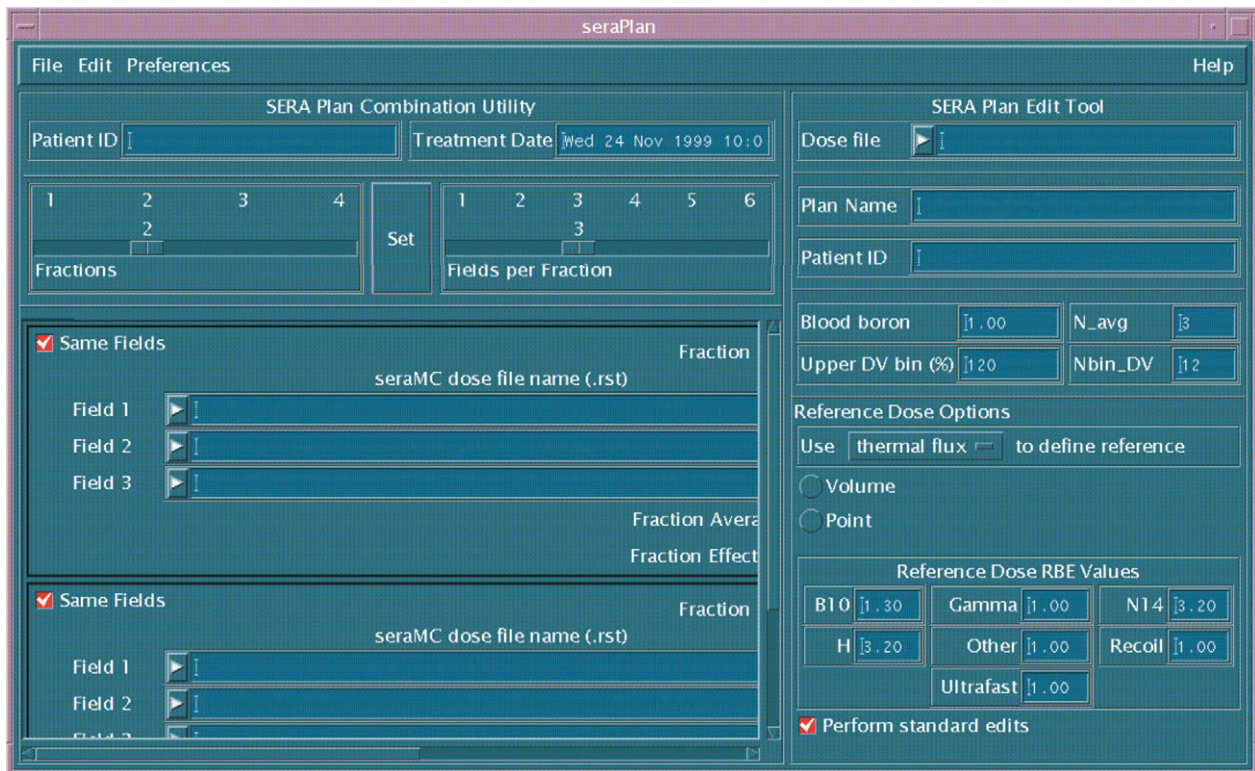


Figure 8.1. Sample seraPlan display.

8.2. Menus

8.2.1. File Menu

The File menu in seraPlan, shown in Figure 8.2, contains eight selections - Load plan file, Apply/save plan, Load edit file, Save edit file, Clear window, Check version, Launch, and Exit.

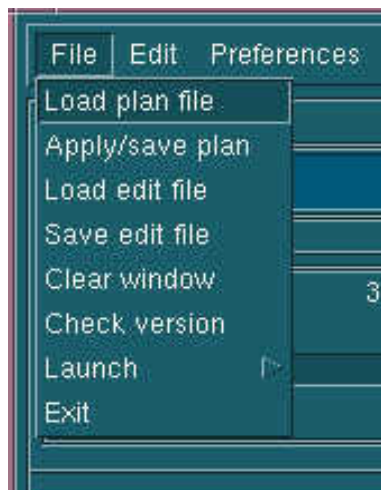


Figure 8.2. SeraPlan file menu display.

The Load plan file selection pops up a file selection widget that allows the user to traverse the file systems and choose a plan file to load into seraPlan. This will read all the relevant information from the selected file and display it in the seraPlan main window. Note that if an

attempt is made to load a file that is not a plan file, no error messages are generated, but the results are unpredictable. Generally, this will produce a default configuration display, with unintelligible information in the data fields. It is advised that the user load only files known to be plan files into seraPlan.

The Apply/save plan selection also pops up a file selection widget, but this only allows the user to choose a directory into which the plan and restart files will be written. After the directory selection is made, the plan file is written, and the dose combination is performed as specified. The plan file name and restart file name are determined by the patient ID entry.

The Load edit file and Save edit file selections pop up file selection widgets to allow the user to choose the edit file to load or the directory in which to save the edit data from the edit tool. The saved edit file name is determined by the specified edit plan name. Note that if an attempt is made to load a file that is not an edit file, no error messages are generated, but the results are unpredictable. Generally, this will produce a default configuration display, with unintelligible information in the data fields. It is advised that the user load only files known to be edit files into seraPlan.

The Clear window selection will re-initialize the data fields in the seraPlan display. This operation is not reversible - all information is permanently lost. A popup window verifies that this is what is actually desired by the user.

The Check version selection is no longer functional and produces no useful response.

The Launch selection produces a submenu listing the other modules in the SERA system. Selection of one will launch that particular module. This cross launch capability is built into all the modules in the SERA system.

The Exit selection exits seraPlan, without saving any information in the window. A popup verifies that this is actually the desired action.

8.2.2. Edit Menu

The Edit menu, shown in Figure 8.3, lists the four types of user-defined edits that may be defined. Each of the four (point, line, dose-volume histogram, and contour) activates a popup that allows the user to define the various parameters needed for each type of edit. Details on each are provided in Section 8.4.2.

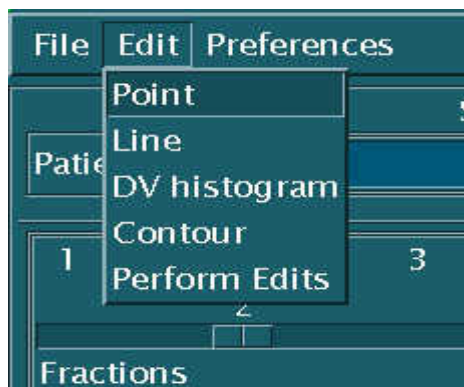


Figure 8.3. SeraPlan edit menu display.

The Perform Edits menu item launches the computation of the edits. Before the actual edit calculations begin, seraPlan prompts the user to define a directory into which all files will be written, and then performs a conflict check to see if any requested files are already defined in that directory. If the conflict check fails, the user is asked for permission to overwrite the existing files. If permission is not granted, the edit calculation halts, and the user must launch the computation again, specifying a different save directory. If execution is allowed to proceed, the standard edits are performed first, followed by the user-defined edits. A status window is displayed so that progress through the edit calculation may be monitored.

8.2.3. Preferences Menu

The Preferences menu in seraPlan, shown in Figure 8.4, has one selection, Preferences, which allows the user to set a variety of default values for both the combination and edit modules. The information entered is stored in the file \$SERA_HOME/Resources/SeraPlan/SeraPlan.rsc. Note that in order to use the preferences, this file must be set to read-write mode (644), by using the system command `chmod`.



Figure 8.4. SeraPlan preferences menu display.

8.2.3.1. Fields/Fractions Preferences

This panel, shown in Figure 8.5, allows the user to set the default number of Fractions and Fields per Fraction. These values are displayed on the appropriate slide bars in the Combination Utility.

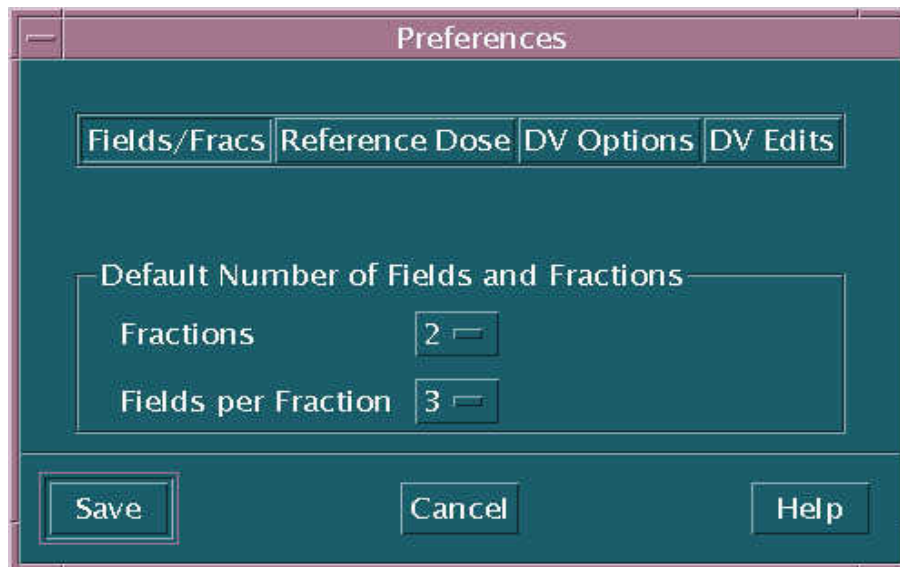


Figure 8.5. SeraPlan Fields/Fractions preferences panel.

8.2.3.2. Reference Dose Preferences

This panel, shown in Figure 8.6, allows the user to set the dose component used to determine the reference dose, and also to set the RBE values used in the reference dose calculation search. These are only applicable to the Volume search, as the Point reference dose merely gives the dose at the chosen location.

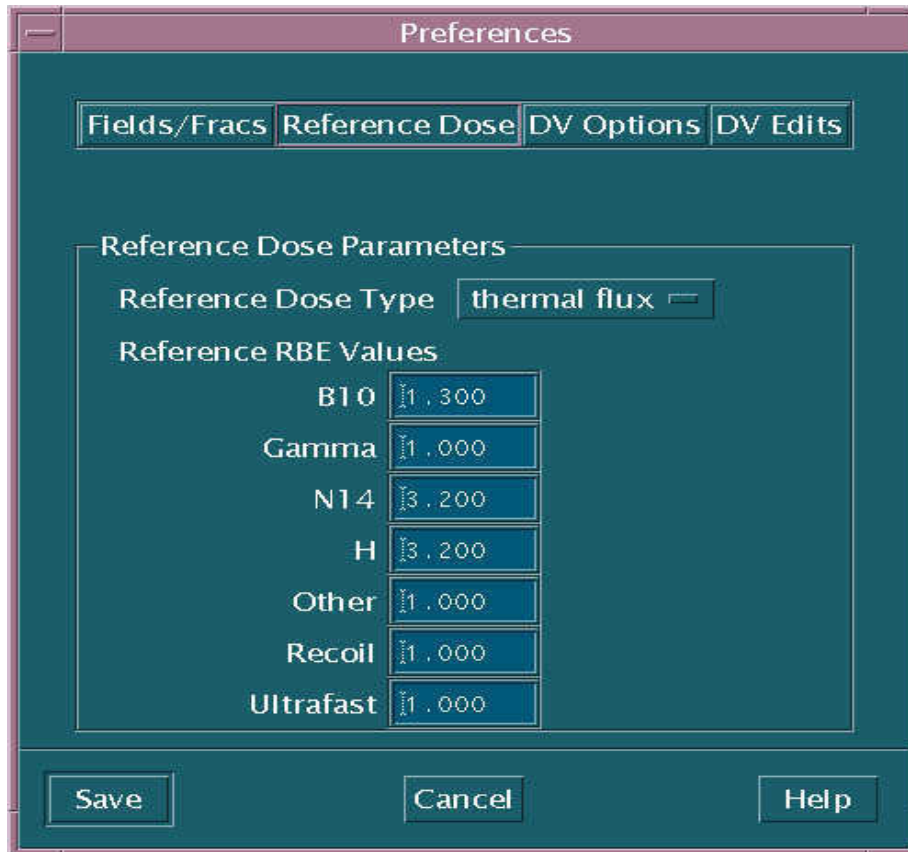


Figure 8.6. SeraPlan Reference Dose preferences panel.

8.2.3.3. Dose-volume Edit Options Preferences

This panel, shown in Figure 8.7, allows the user to set three parameters that affect the appearance of the dose-volume histogram edit output. The parameter N_{avg} controls the number of equal volume regions used. This calculation divides a body into N equal volumes, based upon the dose in each voxel, and then computes the average dose in each region. When only one region is specified (the default), this supplemental edit is not performed.

The parameters N_{bin_DV} and Upper DV bin (%) control the number of bins into which the dose-volume histogram edits are divided, and the upper boundary of the last fixed width bin. Thus, in the example shown in Figure 8.7, there would be 12 bins of 10% width each, ending at 120% of the reference dose, and one additional bin for the volume greater than 120% of the reference dose.

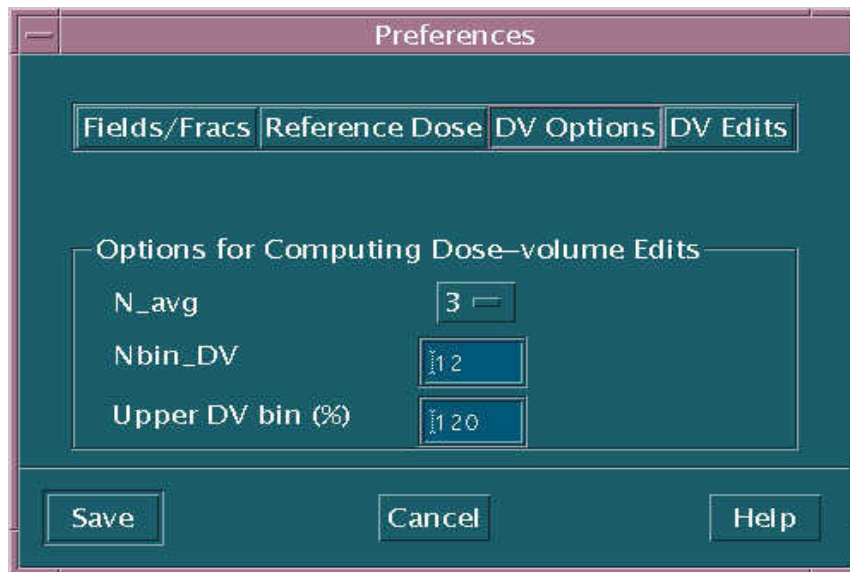


Figure 8.7. SeraPlan Dose-volume Edit Options preferences panel.

8.2.3.4. Additional Dose-volume Edits Preferences

This panel, shown in Figure 8.8, allows the user to specify up to 10 extra dose-volume histogram edits. These edits will be performed in addition to the standard dose-volume histogram edits, and each should be performed over the union of two or more bodies. The default value is zero (0). Please note that any edits specified here will reduce the number of problem specific dose-volume histogram edits allowed in the edit module (presently, up to 10 are allowed).

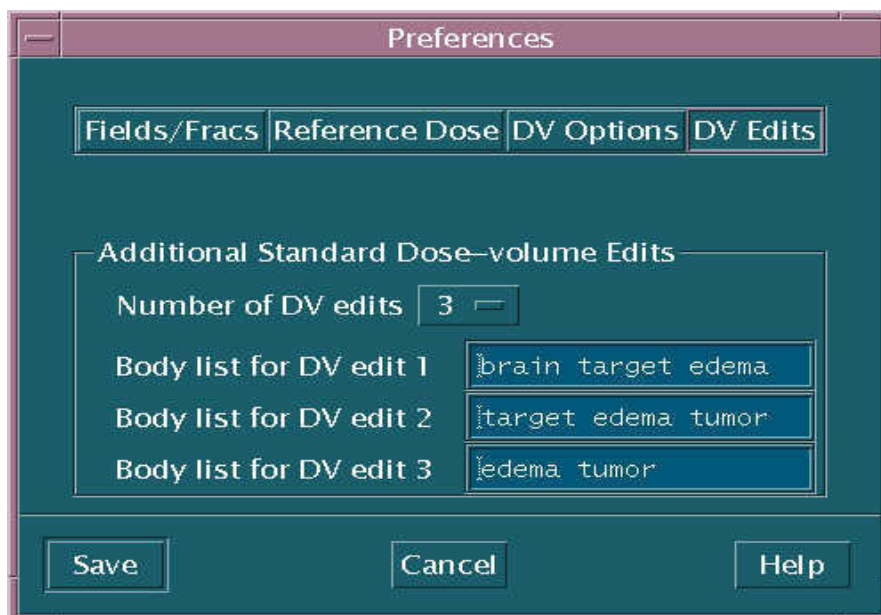


Figure 8.8. SeraPlan Additional Dose-volume Edits preferences panel.

8.2.4. Help Menu

The Help menu, shown in Figure 8.9, has one selection, HelpOnContext, which activates the context sensitive help facility for seraPlan. This facility provides explanations for various

features of seraPlan by clicking the right mouse button on the desired feature in the seraPlan window.

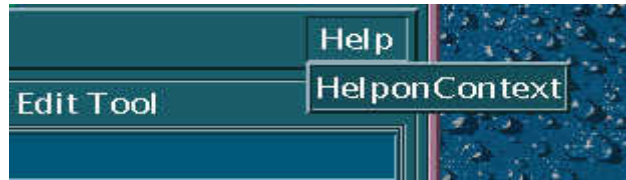


Figure 8.9. SeraPlan help menu display.

8.3. Plan Combination Utility

The plan combination utility allows the user to combine multiple (up to 24) dose files into a single dose file, with weighting by both exposure and boron concentration. This section describes the input required to perform the combination and the files that result.

8.3.1. Input Data

8.3.1.1. Identification Parameters

8.3.1.1.1. Patient ID

The Patient ID field in the seraPlan input allows the user to specify a character string (no whitespaces allowed) to identify the patient treatment plan. This parameter is used to construct all files created by seraPlan - the plan file and the fraction and plan dose (.rst) files. These filenames are constructed automatically from the Patient ID specified. Use of blanks and non-printing control characters is not permitted, as this will cause problems with file displays in the Unix systems. A Patient ID must be specified, or seraPlan will issue an error message in the Apply stage.

8.3.1.1.2. Treatment Date

The Treatment Date field provides a means for denoting the date on which the treatment was performed, or the date when the plan was constructed. The default value is the date when seraPlan was launched. This is only used presently in the plan file for identification purposes, and has no bearing on the dose combination.

8.3.1.2. Field/Fraction Specifications

8.3.1.2.1. Number of Fractions

The number of fractions (distinct boron administrations and/or treatment days) is set by using the Fractions slider. Up to four (4) fractions per plan are allowed. The default is two (2) fractions, unless a different value is set using the Preferences file (see Section 8.2.3). A dose (.rst) file is written for each fraction, as well as for the complete plan. The fraction dose file contains only the information and combined doses for that fraction, with the proper boron and exposure weighting.

8.3.1.2.2. Number of Fields per Fraction

The number of fields (distinct beam placements) per fraction is provided using the Fields per Fraction slider. Up to six (6) fields per fraction are allowed. The default is two (2) fields per fraction, unless a different value is set using the Preferences file (see Section 8.2.3).

8.3.1.2.3. Set Button

The Set button, located between the Fractions and Fields per Fraction sliders, fixes the number of fractions and fields per fraction for the plan, and generates the seraPlan display used to input the field data. Changes to the number of fractions and/or fields per fraction may be made, but the Set button must be activated to implement the changes. If either parameter is changed, any data input in the display is retained and, in the case of reducing either parameter, may be recovered by increasing the appropriate parameter. Any information that is overwritten is lost.

8.3.1.3. Field Data

8.3.1.3.1. Dose File Name

The filenames for the seraMC dose files to be used in the dose combination are given in the seraMC dose file text display. The filename can be typed in directly, or a search of the directory tree can be instituted by clicking on the arrow button to the left of the text widget. Full pathnames are recommended, as this allows access to the dose files from any directory. Note that these filenames will be written into the plan file exactly as entered in the text widget. Dose files must be specified for each active field in the plan.

8.3.1.3.2. Boron Concentration Data

The ^{10}B concentration for each field is entered in the ^{10}B text display. The units on the boron concentration are arbitrary, and largely irrelevant, as long as they are entered in a consistent fashion for all fields. This is a result of the normalization to unit boron concentration, described above in Section 8.1. Generally, the boron concentration is given in parts per million (ppm).

Running simple averages and exposure weighted effective boron concentrations are calculated and displayed in the fraction and plan sum displays. Details of these are provided in Section 8.3.2.

8.3.1.3.3. Exposure Data

The exposure for each field is supplied in the Exposure text display. The units on the exposure are, like those for the boron concentration, arbitrary and irrelevant, as long as they are consistent. Since the combined dose results are normalized to unit exposure, the exposure can be entered in units of minutes, MW-minutes, neutron flux or current, or any other convenient measure. Since variable intensity beams are not yet envisioned for BNCT, it is likely that the exposure will generally be given in terms of time.

Running summations of exposure for each fraction and the plan, along with fraction relative weights, are calculated and displayed in the fraction and plan sum displays. Details are provided in Section 8.3.2.

8.3.1.3.4. Gamma Repair Data

The gamma repair factor is defined as the fraction of the gamma dose that is considered to contribute substantially to multifraction exposures, i.e., the fraction of gamma dose induced damage that is not repaired between fractions. This factor will range from 0 to 1, with single fraction exposures using a factor of one (1). It should be noted that no internal "sanity checks" (other than to ensure that only numerical

data is entered) are performed on the gamma repair factor, so it is up to the user to verify that the values input are correct and appropriate.

8.3.1.3.5. Active Field

The Active button at the far right of the field data display is used to activate or deactivate the data for that particular field. A deactivated field is not included in the dose combination calculation, and all activated fields are included. By default, all fields are active. To deactivate a field, simply click on the red button, and click on the button again to reactivate.

8.3.1.3.6. Same Fields button

The Same Fields buttons, located at the upper left of each fraction subsection, allow the user easily to specify the same field dose file names for multiple fractions. When these buttons are activated, specifying a dose file name for one field in a fraction will propagate that file name to the same field in each fraction, for all fractions that have this button activated. Those fractions that allow Unique Fields (the deactivated state) will not be changed. Note that activation of this button will overwrite any existing file name(s) specified. The default is for the button to be activated for all fractions.

8.3.2. Summation Data

8.3.2.1. Fraction Sum

The fraction sum display is a cluster of four non-editable text widget displays at the end of each fraction. These display the fraction average and effective (exposure weighted) ^{10}B concentration, the fraction total exposure, and the fraction of the total exposure given in that fraction. The effective boron is given by summing the product of the ^{10}B concentration and the exposure for each field, and then dividing by the total exposure. These are running totals, and they are updated as each entry is entered in the field data displays.

8.3.2.2. Plan Sum

The plan sum display is very similar to the fraction sum display. It appears at the bottom of the seraPlan display scrolled window and shows the average and effective ^{10}B concentrations and the total exposure. Again, these are running totals, updated continuously as data are entered.

8.3.3. Output Files

8.3.3.1. Plan File

The plan file written by seraPlan contains all the information shown in the seraPlan plan combination display. It is an ASCII file, and it generally works on the keyword-value system. A sample plan file is shown below.

```
Patient: Patient
Treatment date: Tue 30 Mar 1999 14:15 MST
FRACTIONS=2 FIELDS=3
Fraction 0
Field 0
/users/cew/Patient/inel/I1/I1.rstb
17.12 16 1 1
Field 1
/users/cew/Patient/inel/P1/P1.rstb
```

```

15.86 10 1 1
Field 2
/users/cew/Patient/inel/S1/S1.rstb
13.94 6.5 1 1
Fraction sums 15.64 16.0963 32.5 0.488722
Fraction 1
Field 0
/users/cew/Patient/inel/I1/I1.rstb
16.34 17 1 1
Field 1
/users/cew/Patient/inel/P1/P1.rstb
13.74 10 1 1
Field 2
/users/cew/Patient/inel/S1/S1.rstb
12.71 7 1 1
Fraction sums 14.2633 14.8279 34 0.511278
Plan sums 14.9517 15.4478 66.5

```

All values are as displayed in the seraPlan display, with the exception of the Active button status. This is given as 0 for inactive fields, and 1 for active fields. The plan file generated is called XXplan, where XX is the Patient ID specified.

8.3.3.2. Combined Dose Files

The combined dose file is formatted exactly as the dose files created by seraMC. The individual field parameters from each specified field dose file (beam directional inputs, input files, etc.) are retained, and written in the combined dose file.

In all cases, a total plan dose file is written. For the case of multifraction irradiations, a combined fraction dose file is written for each fraction. The dose results given in the plan dose file are the weighted sum of all the fraction dose file results. The new dose files are called XXplan.fracN.rst and XXplan.rst, where XX is the Patient ID, and N is the fraction number (for multi-fraction plans).

8.4. Edit Module

The edit module allows the user to perform a variety of edits on a single dose file, which may be either a single field dose file, or one produced by the plan combination utility. This section describes the input required from the user, and the resulting files that are generated.

8.4.1. Basic Input Data

8.4.1.1. Dose File

This gives the name of the dose (.rst) file that will be used to generate the edits. It may be either a plan file created by the seraPlan combination utility, or a single field dose file written by seraMC. The filename may be typed in directly, or selected by traversing the directory tree with the file selection widget invoked by pressing the arrow key attached to the text widget.

8.4.1.2. Plan Name

This field provides a filename prefix that is used to construct the names of all generated files from the edit module. Any text string that does not contain spaces may be entered.

8.4.1.3. Patient ID

This field provides a text string that is used in the headers for various edit outputs. Its primary function is as a delimiter between line edits, but it also appears in the title of any dose-depth plots generated by seraPlot from these line edits. Any string that does not contain spaces may be entered.

8.4.1.4. Blood B10

The Blood B10 entry provides a concentration of boron-10 in the blood, which is used to calculate the boron concentration in all tissue types. This is done by using the tissue-to-blood ratios provided in the .uvh file. Generally, this entry is given in ppm boron-10. Note that this is not used for the reference dose determination, unless the Point option is selected as the search type (see Section 8.4.1.9).

8.4.1.5. N_avg

This parameter activates a supplemental dose-volume edit for each body. It divides the body into N_avg regions of equal volume, with the volume elements (in this case, the edit voxels) sorted by total dose, and computes the average dose in each region. Thus, if the parameter is set to 3, then the body will be divided into three regions, with the first region containing the voxels with doses above the 67th percentile, the second region containing the voxels with doses between the 33rd and 67th percentiles, and the third region containing the voxels with doses below the 33rd percentile. If the parameter is set to 1, then this supplemental edit is not performed. A default value for this parameter may be specified in the Preferences file (see Section 8.2.3.3).

8.4.1.6. Upper DV bin (%)

This entry, along with Nbin_DV, helps determine the number and width of bins into which each dose-volume histogram edit will be broken. The number of bins will always be Nbin_DV+1, and each bin will have a width given by the upper bin value specified divided by Nbin_DV. There will also be an additional bin, which will accommodate all regions with dose greater than the upper bin value.

It is recommended that this value be evenly divisible by Nbin_DV, although this is not required. A default value for this parameter may be specified in the Preferences file (see Section 8.2.3.3).

8.4.1.7. Nbin_DV

This entry determines the number of bins into which the dose-volume histogram edits will be broken. As explained in Section 8.4.1.6, there will always be Nbin_DV+1 bins. A default value for this parameter may be specified in the Preferences file (see Section 8.2.3.3).

8.4.1.8. Reference Dose Search Component

The reference dose search component is selected from the pulldown menu located in the Reference Dose Options subpanel, as shown in Figure 8.10. This dose component will be the only component compared for the Volume reference dose search (see Section 8.4.1.9), where the reference dose is the dose at the center of the edit voxel with the highest value of the selected dose component. At present, three options are available: total dose, thermal flux, and gamma dose. A default value for this parameter may be specified in the Preferences file (see Section 8.2.3.2).

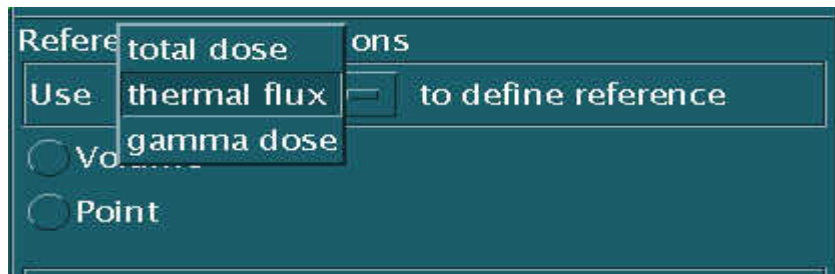


Figure 8.10. Pulldown menu used to select the dose component used in the reference dose search.

8.4.1.9. Reference Dose Search Type

There are two options for the type of reference dose search: Volume and Point. If the Point option is selected, the user is prompted for an (x,y,z) location, as shown in Figure 8.11. It is actually a misnomer, as there is no actual search for the reference point in this case; the reference point is at the specified spatial location, and the reference dose is the physical dose at this point, with the boron concentration given by the product of the blood boron concentration (see Section 8.4.1.4) and the tissue-to-blood ratio for the material at the specified location.

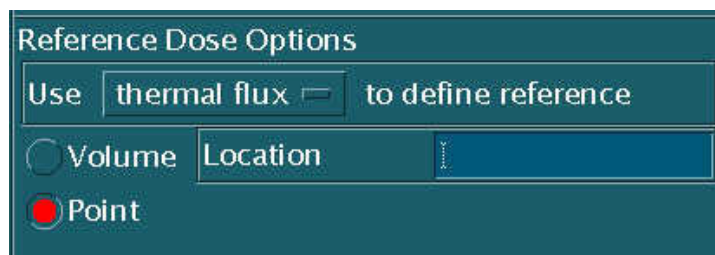


Figure 8.11. Display produced by selecting the Point type of reference dose search.

Selection of the Volume option causes the program to prompt the user for a boron concentration, and a list of bodies to be included in the search, as shown in Figure 8.12. The boron concentration provided will be used only for the reference dose search, and will not be modified by the tissue-to-blood ratios of the specified bodies. The list of bodies denotes the only bodies that will be included in the search. The search will encompass all edit voxels in the defined bodies and the reference point will be defined as the location of the center of the edit voxel with the largest value of the specified dose component (see Section 8.4.1.8), with the dose component(s) modified by the specified reference dose RBE values (see Section 8.4.1.10).

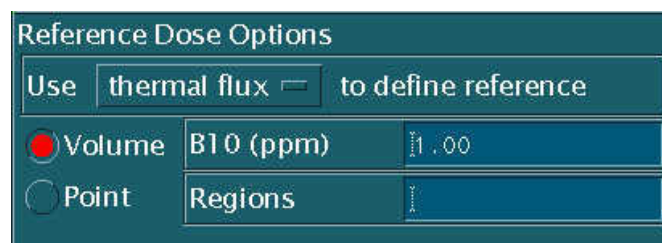


Figure 8.12. Display produced by selecting the Volume type of reference dose search.

8.4.1.10. Reference Dose RBE Values

The reference dose RBE values are defined in the seven text widgets below the Reference Dose Options subpanel and are used only for the reference dose search. At present, the Other and Ultrafast RBE values may both be defined, but only the Other dose RBE is actually applied. This will very likely change in future versions, to accommodate varying RBE values for high-energy (> 20MeV) neutron therapy. Default values for these parameters may be specified in the Preferences file (see Section 8.2.3.2).

8.4.1.11. Perform Standard Edits Button

This toggle button, located at the bottom of the edit module input panel, allows the user to choose to perform, or not perform, the standard edits (see Section 8.4.3). If toggled on, the standard edits will be performed. If toggled off, they will not be computed.

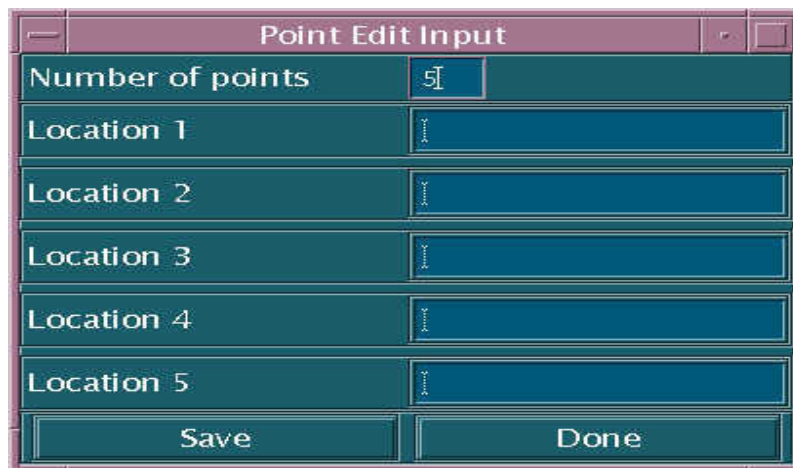
8.4.2. User-defined Edits Input Data

The input data for the four types of user-defined edits are activated from the Edit menu (see Section 8.2.2). Each popup will initially contain only a text widget, in which the user is asked to set the number of that particular edit type desired. After typing in the number, and hitting the Enter or Return key, the popup will then display the requisite widgets for the desired number of edits. When the required data is entered, the user must then click on the Save button at the bottom of the popup to store information. The Done button will then exit the popup.

Be aware that the information is not stored, and thus, the edits will not be performed, if the Save button is not clicked. However, if the user does not click the Save button, all is not lost. The information may be recovered by reactivating the popup.

8.4.2.1. Point

An example of the Point popup is shown in Figure 8.13. This shows the required input fields for five point edits. For each point edit, a triplet of values (x,y,z) must be provided, giving the spatial location of the requested edit. Checking is performed to assure that three, and only three, values are entered for each point edit location. Up to 20 user-defined point edits are allowed in a single problem.



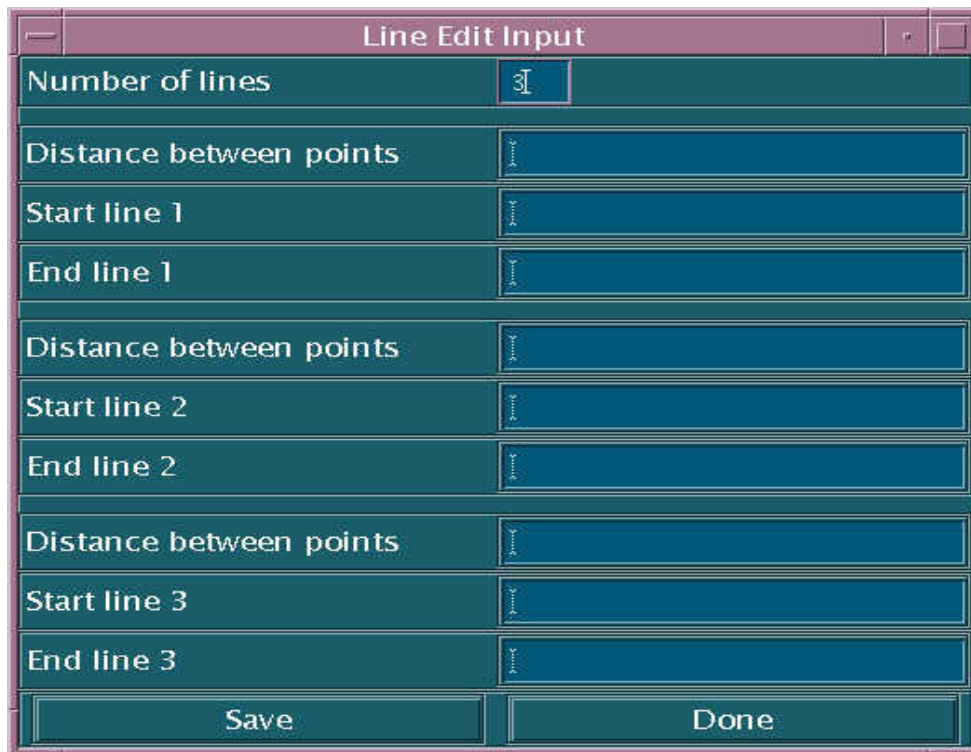
Point Edit Input	
Number of points	5
Location 1	x y z
Location 2	x y z
Location 3	x y z
Location 4	x y z
Location 5	x y z
Save	Done

Figure 8.13. The popup for the user-defined Point edits.

8.4.2.2. Line

A line edit is simply a series of point edits for a sequence of points along a line, with a regular, defined spacing between points. All dose components are defined in the line edit results.

An example of the Line popup is shown in Figure 8.14. This shows the required input fields for three line edits. For each line edit, three parameters are required. The first is the spacing, in cm, between points. The last two parameters define the initial and final point of the line and are given as triplets (x,y,z) representing a spatial location in the model. As with the point edit, checking is performed to assure the proper number of entries for each line edit input set. Up to 10 user-defined line edits are allowed in a single problem.



Line Edit Input	
Number of lines	<input type="text" value="3"/>
Distance between points	<input type="text"/>
Start line 1	<input type="text"/>
End line 1	<input type="text"/>
Distance between points	<input type="text"/>
Start line 2	<input type="text"/>
End line 2	<input type="text"/>
Distance between points	<input type="text"/>
Start line 3	<input type="text"/>
End line 3	<input type="text"/>
Save Done	

Figure 8.14. The popup for the user-defined Line edits.

8.4.2.3. Dose-volume Histogram

A dose-volume histogram edit integrates the dose over a body or set of bodies. In seraPlan, the integration is performed by calculating the dose at the center of each univel in the body, then summing the product of the dose in the univel and the univel volume over all univels in the list of accepted bodies. The results are reported as the fraction of the volume that falls within a range of dose values. These dose value ranges are generally reported as a percentage of the reference dose. Each individual dose component is reported separately.

An example of the Dose-volume Histogram popup is shown in Figure 8.15. This shows the required input fields for three DV histogram edits. The only required input for the user-defined DV histogram edits is a list of body names, with the body names separated by white spaces (spaces, tabs, or returns). Up to 10 user-defined DV histogram edits are allowed in a single problem.

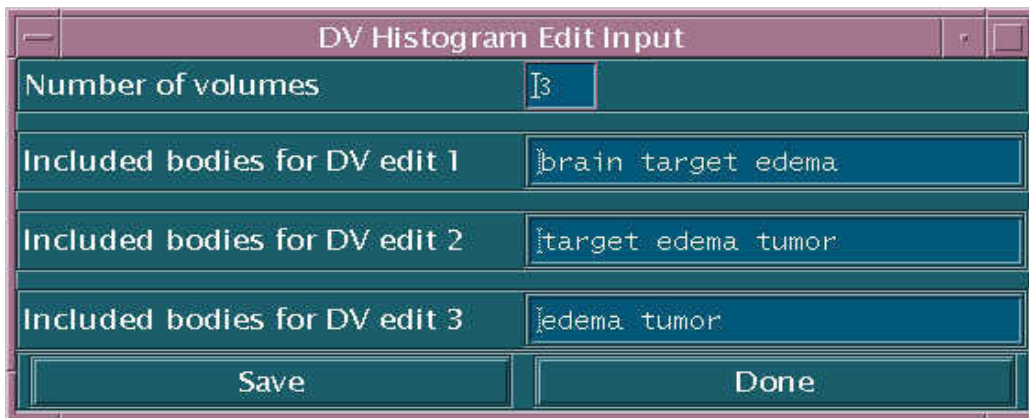


Figure 8.15. The popup for the user-defined Dose-volume Histogram edits.

8.4.2.4. Contour

A contour, or surface, edit is a dose edit over a regular array of points, all of which reside in a common plane. These data are then used by the seraDose and sera3d modules to construct isodose contours.

An example of the Contour (surface edit) popup is shown in Figure 8.16. This shows the required input fields for two contour edits. Each contour edit requires two basic input groups: the prefix for the files to which data is written; and the definition of the plane over which the edit is performed. The plane definition requires a base point and two vectors. The base point gives a starting location, and the two vectors, which must be normal to each other, then define the remainder of the plane inside the edit space. Please note that the edit plane will be oriented along the directions defined by the two vectors. This may cause the plane to be skewed within the edit voxel space, if the vectors are not tangent to coordinate planes (i.e., x-y plane, y-z plane, and x-z plane).

The skew is a result of the method employed to map the defined plane into a space more amenable to editing. First, the two vectors are normalized to unity, and a third vector, normal to the two defined, is constructed and normalized. This gives a set of three basis vectors, which function not only as the coordinate axes for the transformed space, but also as the self-orthogonal transformation matrix between the two spaces. The base point then defines the z-level in the new space and the plane tangential to the x-y plane in the transformed space. The edit is then performed by stepping, in a regular fashion, over the x-y plane, with the maxima and minima for the two axes defined by the maxima and minima for the edit space. Points that lie outside the univel or base edit space, i.e., those that have no body assignment, are assigned a zero dose.

Up to four (4) user-defined contour edits are allowed in a single problem.

8.4.3. Standard Edits

A set of standard edits is produced by seraPlan when the Perform Standard Edits button is toggled on. These include:

- point edits for the target point of each field in the plan
- point edits at each constraint point defined in the model
- line edits along the beamline of each field from the patient geometry entry point to the patient geometry exit point at 0.25cm intervals

- dose-volume histograms for each body defined in the model, except buffer
- contour edits for each image plane in the image set.

The image shows a dialog box titled "Surface Edit Input". It has a title bar with a close button on the right. The main area contains several input fields:

- "Number of surfaces" with a text box containing the number "2".
- "Contour file 1" with a text box and a small "Browse" button to its right.
- "Base point 1" with a text box and a small "Browse" button to its right.
- "Vector 1" with a text box and a small "Browse" button to its right.
- "Other vector 1" with a text box and a small "Browse" button to its right.
- "Contour file 2" with a text box and a small "Browse" button to its right.
- "Base point 2" with a text box and a small "Browse" button to its right.
- "Vector 2" with a text box and a small "Browse" button to its right.
- "Other vector 2" with a text box and a small "Browse" button to its right.

 At the bottom of the dialog, there are two buttons: "Save" on the left and "Done" on the right.

Figure 8.16. The popup for the user-defined Contour edits.

8.4.4. Output Files

Several output files are created by seraPlan. Each is written to the Save Directory defined at the beginning of the edit calculation (see Section 8.2.2). The all filenames, except those for the user-defined contour edits, are prefixed with the Plan Name defined in the input data. Thus, if the Plan Name is george, then the files will all have names george.xxx and the saved edit file (see Section 8.2.1) will be named georgeedit.

8.4.4.1. Reference Dose

The reference dose information is written to the .edit file. This edit is always performed, as it is needed for practically every other edit. It is the first edit to appear in the .edit file, immediately after the list of bodies defined in the problem.

8.4.4.2. Point Edits

All point edits, whether standard or user-defined, are written to the .edit file. The point edit results appear immediately after the reference dose edit. The target points are printed first, followed by the constraint points, and, finally, the user-defined points.

8.4.4.3. Line Edits

All line edits, whether standard or user-defined, are written to the .lin file. The beamline edits appear first, followed by any user-defined line edits.

8.4.4.4. Dose-volume Histogram Edits

All dose-volume histogram edits, whether standard or user-defined, are written to the .dvh file. This includes the supplemental dose-volume edits generated by setting N_{avg} . The standard DV histogram edits appear first, then the user-defined edits.

8.4.4.5. Contour Edits

All contour edits cause three files to be created. These are analogous to the QSH files, with a header file (.chd) for general information about the contour planes, and the data files for the dose data (.cdf) and mask information (.cmf).

8.4.4.5.1. Standard Contour Edits

The standard contour edits are written to a concatenated (3D) contour file. All planes appear in these files, which will be prefixed with the Plan Name.

8.4.4.5.2. User-defined Contour Edits

The user-defined contour edits will each create a set of contour files. The filenames will be prefixed with the name given for the filename in the contour popup (see Section 8.4.2.4).

9. Dose Contouring (seraDose)

Dose contours can be viewed in two dimensions using seraDose. SeraDose can be invoked from either the "Sera Dose" button on the seraMenu (see Figure 3.2) or from the command line by entering seraDose, assuming the executable has been correctly installed in the SERA Target/bin directory and the paths are set correctly.

Upon invoking seraDose, the user will be presented with two windows (Figure 9.1). The first window, entitled "seraDose," is the main working window. The second window, entitled "seraDose: Image Contouring," is used to view single images.

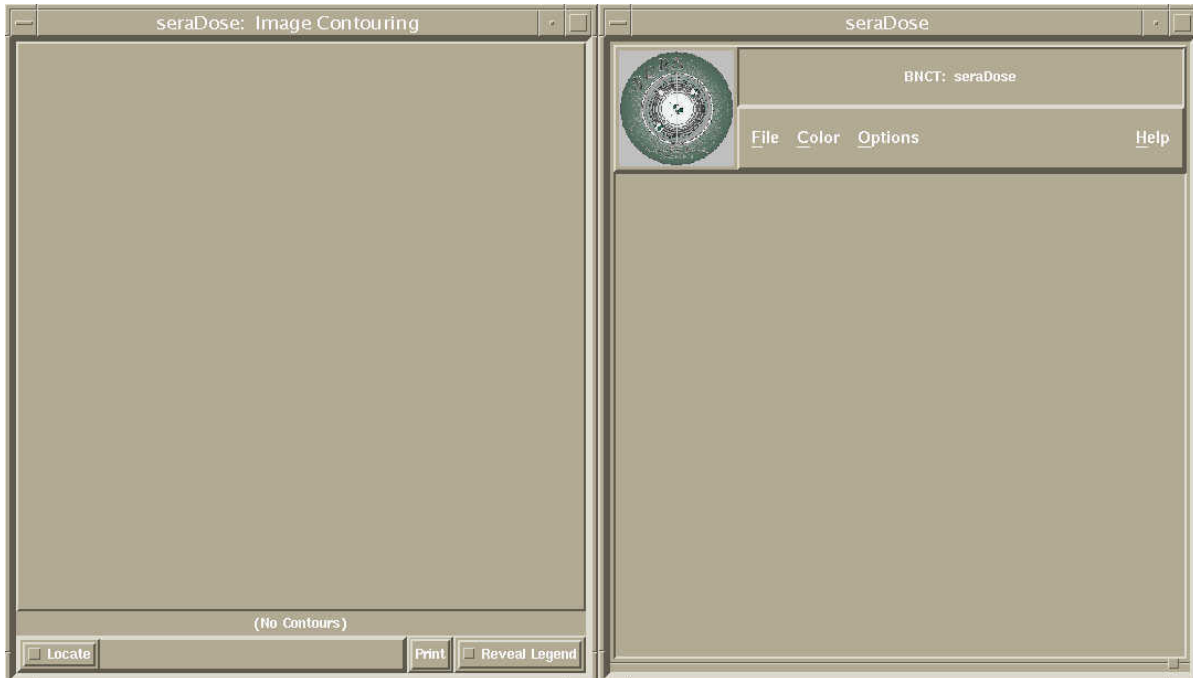


Figure 9.1. SeraDose main windows.

The menu bar on the seraDose main window contains the following selections, as shown in Figure 9.2.



Figure 9.2. seraDose Main Window. Invoking seraDose presents the user with four main options seen here.

9.1. File Menu

After the seraDose main window appears, pull down the File menu item and the following options appear:

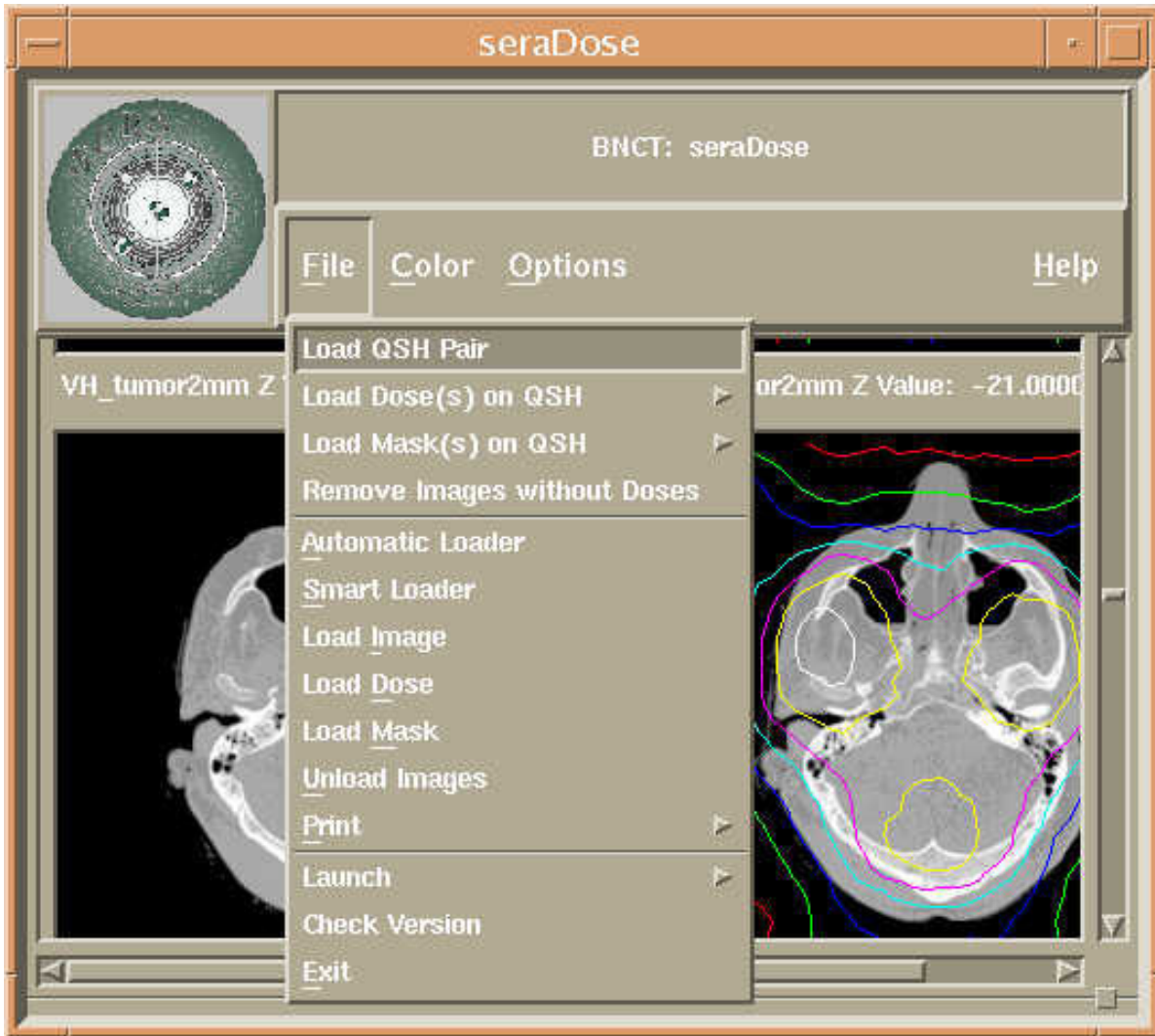


Figure 9.3. The seraDose File menu.

Load QSH Pair	Loads images in the QSH format. It will invoke the QSH load selection window.
Load Dose(s) on QSH	Allows user to load doses on QSH images. From this menu, there are three options, described in Section 9.5.
Load Mask(s) on QSH	Allows user to load masks on QSH images. From this menu, there are three options, described in Section 9.6.
Remove Images without Doses	Allows the user to unload images that do not have dose values, once doses have been loaded on QSH images.
Automatic Loader	Automates the process of loading single image, contour, and mask files.
Smart Loader	Allows the user to continually load any number of image, contour, and mask files. The file selection box will continue to prompt the user to load the next file until Done is selected in the box.

Load Image	Allows user to load a single image file.
Load Dose	Allows user to load a single contour file (ending in ,contour).
Load Mask	Allows user to load a single mask file (ending in ,mask).
Unload Images	Unloads all images, contours, and masks.
Print	Provides the ability to print either a window or the selected image.
Launch	Allows any other SERA module to be invoked by selecting in from this submenu.
Exit	Closes seraDose. Note: seraDose creates temporary files during its execution. It is important to close seraDose properly using this Exit option for seraDose to clean up the temporary files.

9.2. Color Menu

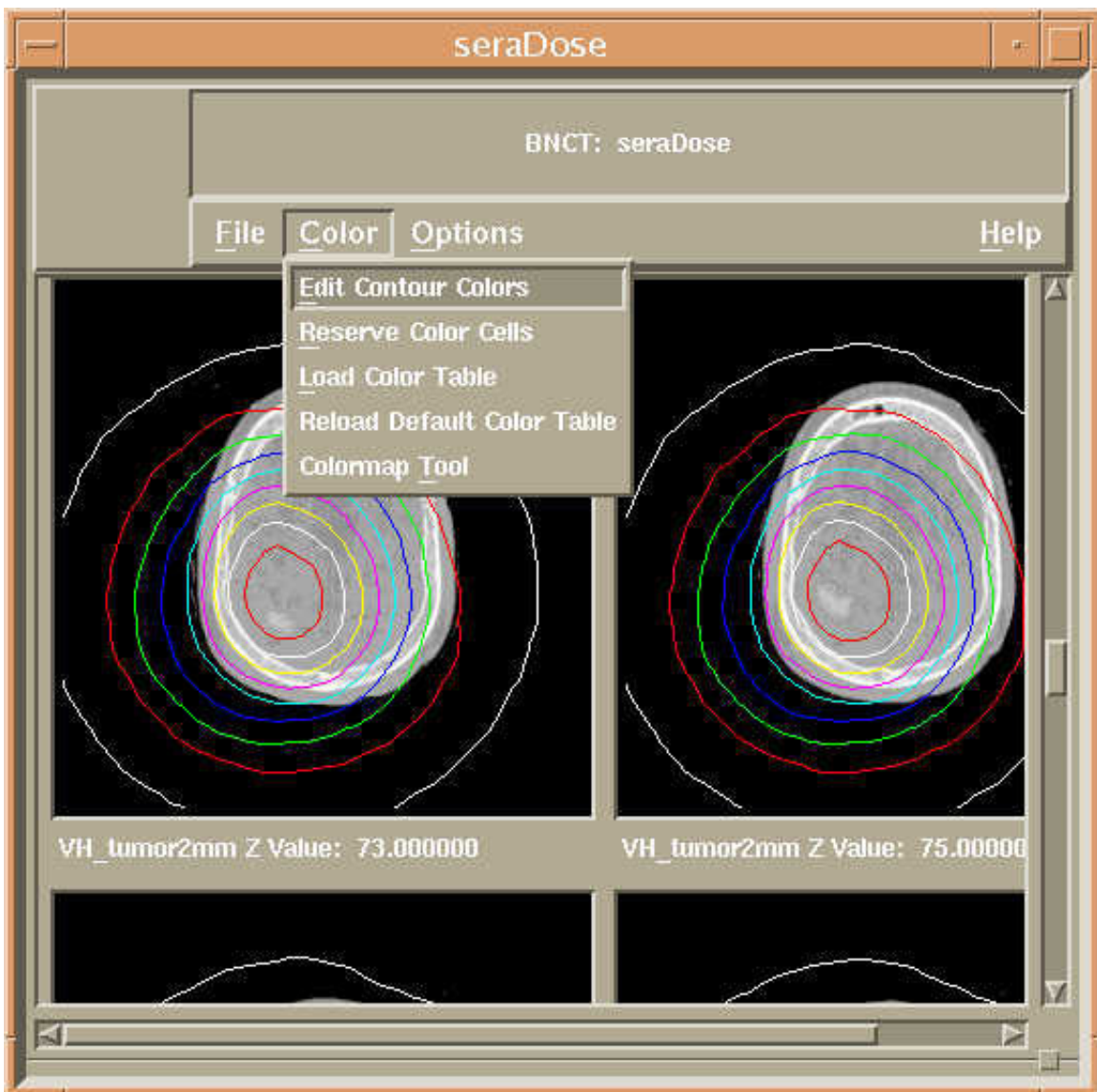


Figure 9.4. The seraDose Color menu.

Edit Contour Colors	Invokes a window that allows user to enter colors of the dose contours.
Reserve Color Cells	Allows user to change the number of color values reserved for the images.
Load Color Table	Loads a colormap (cmap) from a file.
Reload Default Color Table	Reloads the default colormap for the current view.
Colormap Tool	Invokes a window to view and fine tune colormap values.

9.3. Options Menu

The SeraDose Options menu is shown in Figure 9.5.

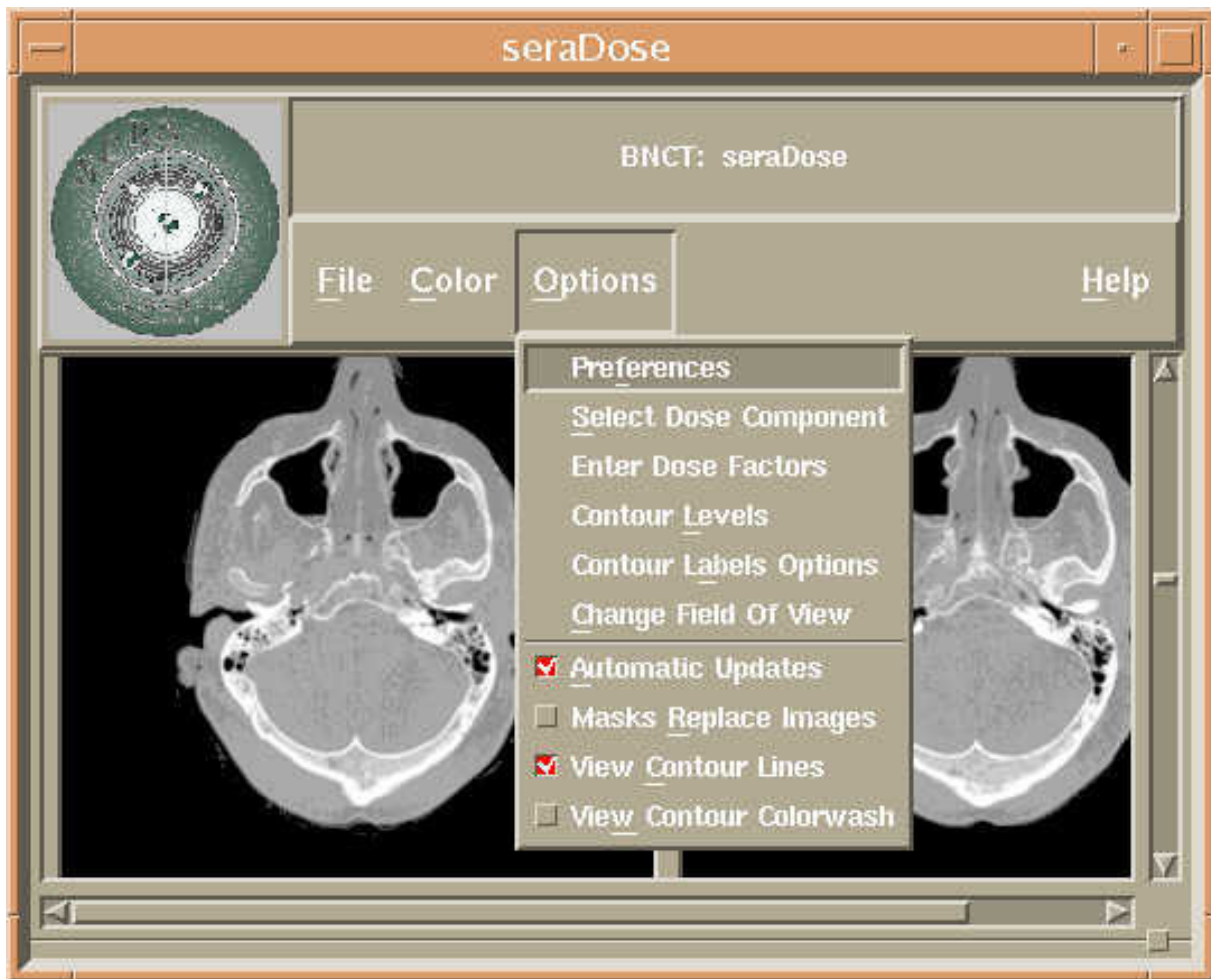


Figure 9.5. The seraDose Options menu.

The following table describes each selection on the menu.

Preferences	Edits and saves seraDose preferences.
Select Dose Component	Chooses which dose component to view.
Enter Dose Factors	Changes dose factors and recomputes contours.
Contour Levels	Selects which contour levels to view.
Contour Labels Options	Changes contour labeling options, such as font size.
Change Field of View	Changes the current field of view.
Automatic Updates	Allows changes to affect all images when toggled on. Limits changes to affect only the currently selected image when toggled off.
Masks Replace Images	Views the mask files as images.
View Contour Lines	Shows colorwashed images when toggled off. Shows contour lines when toggled on. Contour lines or colorwashes can only be viewed when dose files are loaded.
View Contour Colorwash	Shows contour lines when toggled off. Shows images contour colorwashed when toggled on.

9.4. Help Menu

The Help menu has one option, Help On Context. When this menu is selected, the cursor will look similar to a question mark with an arrow. For help with an option in seraDose, simply click on the option with the cursor and you will be directed to the web version of this document.

9.5. Loading Images

SeraDose can load two types of image formats. The first type of image format is the QSH pair format. QSH images are contained in a file with the .qim extension. A corresponding header file with the .qhd extension is needed to properly load the images. SeraDose extracts the necessary information from the QSH pair and loads all images in the files. The user can see what image set is currently being worked with by viewing the label under each image. Once several images have been loaded, an image can be selected by clicking on it in the main seraDose window. The current image is indicated with a red border, as shown in Figure 9.6.



Figure 9.6. The label below each image contains information about the image set and the image. This particular label indicates that this is the VH_tumor2mm set and that this image has a z value of 5.0.

The second type of image format seraDose is capable of loading is raw images. SeraDose provides three ways to load single raw images:

- Load Image - Use this option to load single raw images. This option allows the user to load one image at a time. Once invoked from the File menu, the Load Image option presents the user with the file selection box pictured in Figure 9.7.

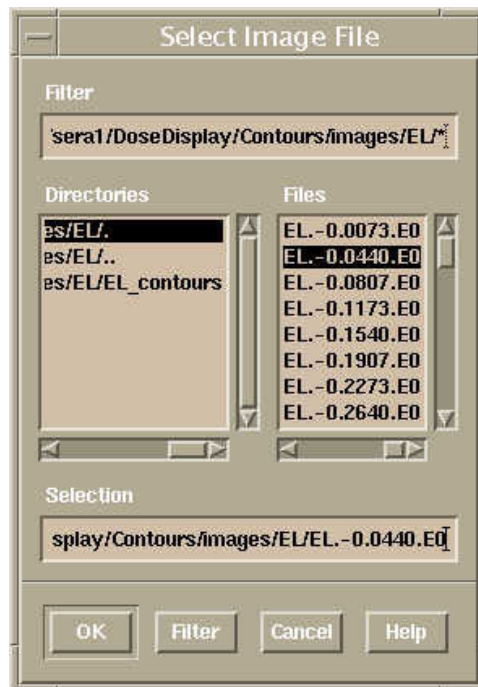


Figure 9.7. Selecting Load Image from the File menu presents the user with a file selection box.

- Automatic Loader - The automatic loader speeds up the process of loading individual images, contour files, and mask files. Upon selecting the automatic loader, the user will be prompted to select an image file and the image's corresponding contour and mask files. Once these three files have been selected, the automatic loader can load all other corresponding image, contour, and mask files.

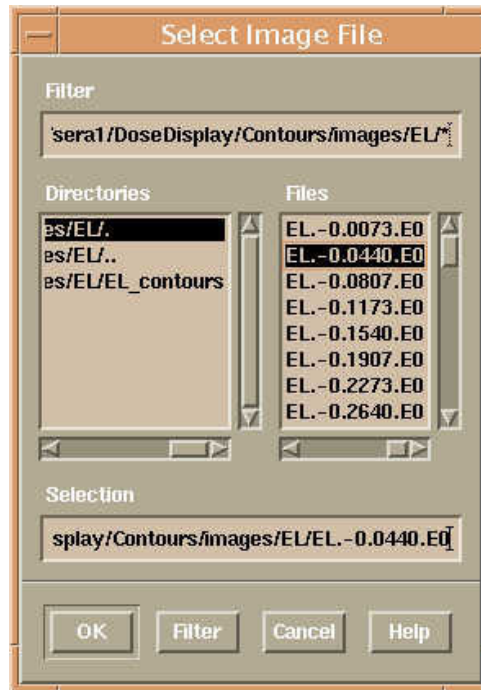


Figure 9.8.a. Using the Automatic Loader presents the user with the Select Image File box.

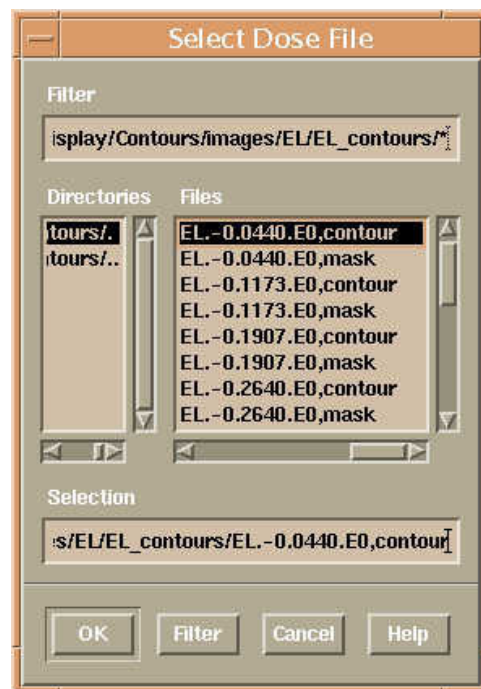


Figure 9.8.b. Using the Automatic Loader presents the user with the Select Dose File box.



Figure 9.8.c. Using the Automatic Loader presents the user with the Select Mask File box.

- Smart Loader - Loads the images along with their corresponding doses and masks utilizing a convenient set of prompts. After choosing the Smart Loader option, select the name of the first image to be loaded from the file selection widget. As soon as an image is loaded, a new popup prompts for this image's corresponding dose. After the dose is selected, another popup asks for this same image's corresponding mask. Finally, after the mask is selected, seraDose will prompt for any other images to be loaded. For each succeeding image, seraDose will automatically attempt to load the corresponding dose and mask files if these files exist. This is done by matching the file name prefixes and looking in the same directory in which the first dose and mask files were located. It is assumed by convention that dose files end in ,contour and mask files end in ,mask. The image files do not have to be in the same directory where the dose and mask files are located. Press Done to stop loading images.

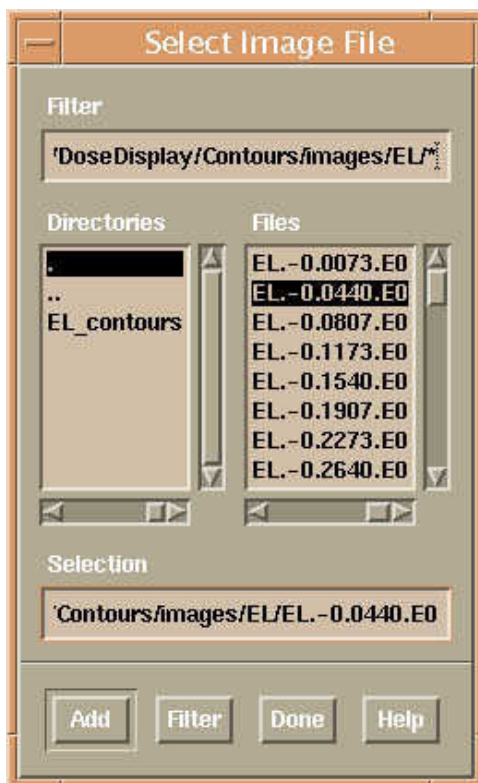


Figure 9.9.a. Smart Loader file selection window for loading images.



Figure 9.9.b. Smart Loader file selection window for loading contours.



Figure 9.9.c. Smart Loader file selection window for loading masks. Continue until Done is selected.

9.6. Loading Contours

Once an image or an image set is loaded, dose contours can be viewed. The contour files this section describes are generated by seraPlan and the contour lines are computed using a marching squares strategy [9]. The result of these operations will be similar to that shown in Figure 9.10.

There are several ways to load contour files.

- Loading Single Contour Files - This option allows loading a single contour file. It is important to first select the image on which the contour file is to be displayed. To accomplish this, click on the image in the main seraDose window. Next, select Load Dose from the File menu. This option will bring up a file selection box where selection of one contour file can be made.
- Loading Contours in the Automatic or Smart Loader - After the files are selected in the Automatic Loader, all corresponding contours will be loaded. Similarly, the Smart Loader provides a convenient way to load contour files.
- Loading Contours on QSH Images - There are four ways to load contours on QSH images.
 - Load Dose - This is the same option as described above. It can be used to load any contour file on any image.
 - Load All Doses - User selects a directory containing single dose contour files. Selecting a directory is accomplished by selecting any file in the desired directory using the file selection box. SeraDose matches each contour file (ending in ,contour) on an image with the closest z value.
 - Load Dose on Current Image - User selects a directory containing single dose contour files. SeraDose searches the directory for the contour file with the closest z value to the current selected image.

- Load 3D Contour File - Loads a seraZipped, concatenated contour file (ending in .cdf.sz). A header file (ending with .chd), which corresponds to the contour file, must be in the same directory as the contour file.

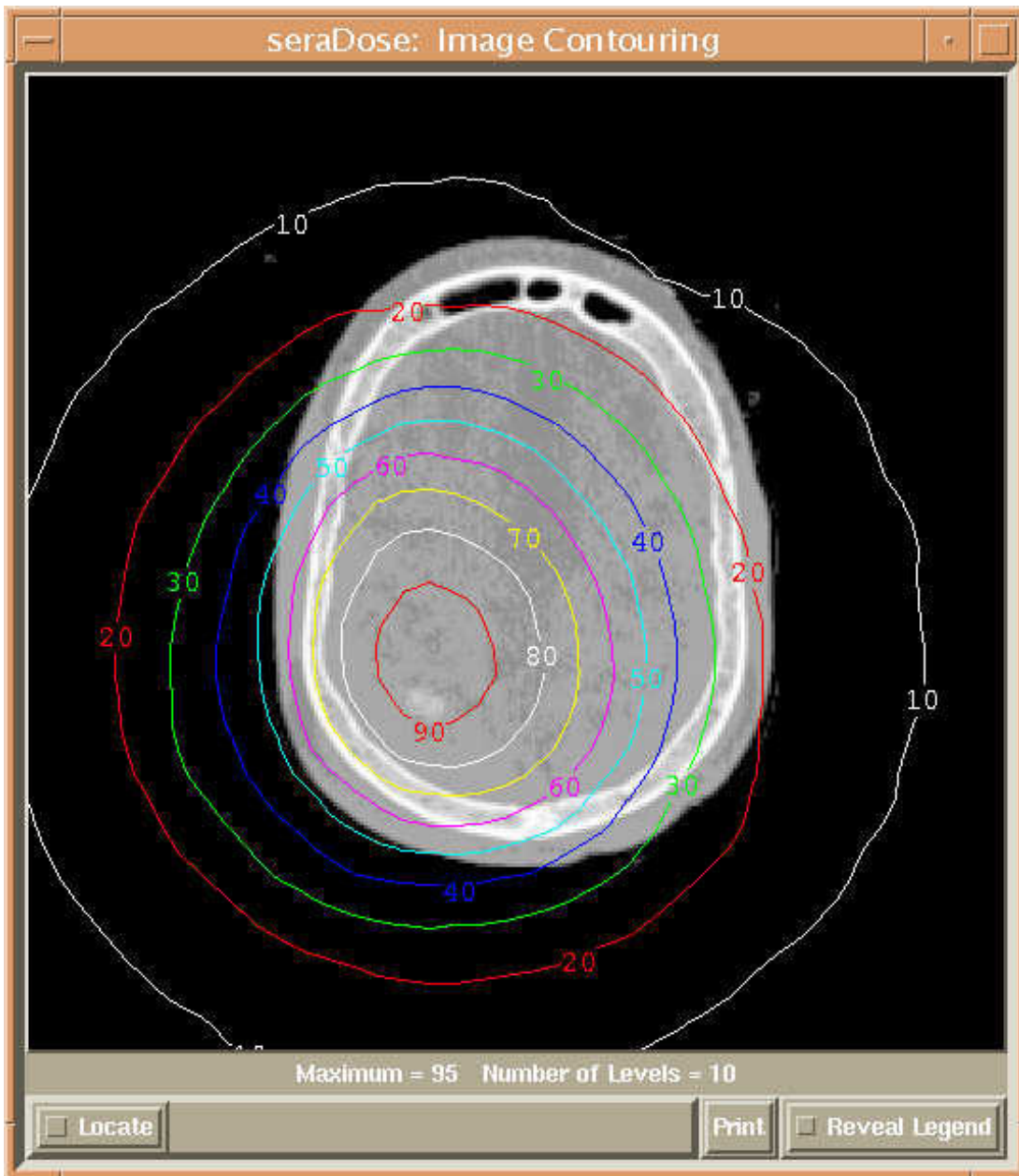


Figure 9.10. An image with contours in the Image Contouring window.

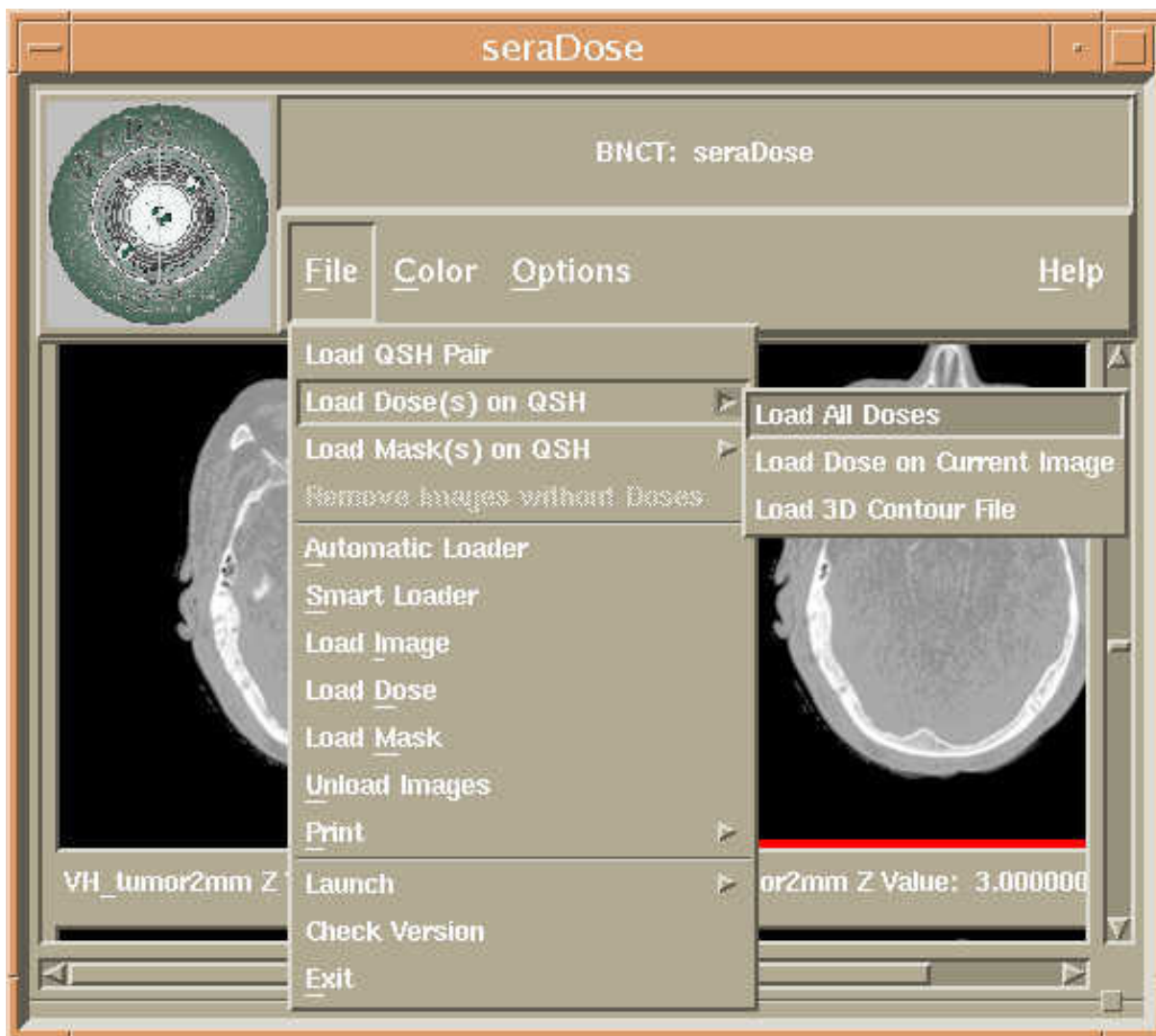


Figure 9.11. Multiple ways of loading contours on QSH images.

9.7. Working with Contours

Once the contours have been loaded, there are several options provided to work with them.

- Editing Contour Colors - This option is described in detail in Section 9.10.
- Changing Contour Levels - To change, add, or remove contour levels, select Contour Levels from the Options menu. This will bring up the Contour Levels window (Figure 9.12). To change the contour levels, enter each contour level on a line in the window. Each contour value should be between 0 and 100. To apply the new contour values, click the Apply button. To close the window without applying the new contour values, click the Cancel button.
- Contour Labels Options - The Contour Labels Options window (Figure 9.13) provides a variety of labeling options. This window allows the user to control the label sizes and fonts for the numerical isodose contour labels on the images. Separate sliders are available for both the large (single) image window labels and the small (multiple) image window labels. No changes are implemented in the labels until they are applied with the Apply to Images button. To close the window without applying any changes, double click on the window menu button in the top left corner of the window.

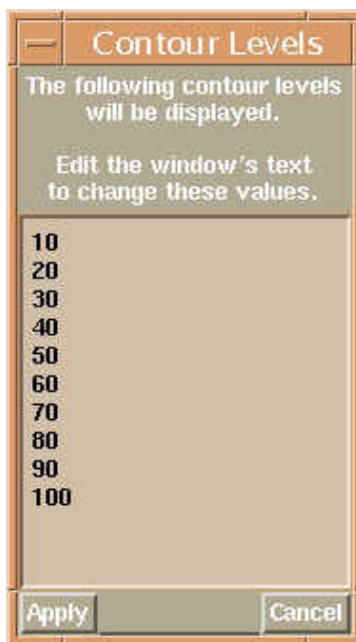


Figure 9.12. The Contour Levels window.

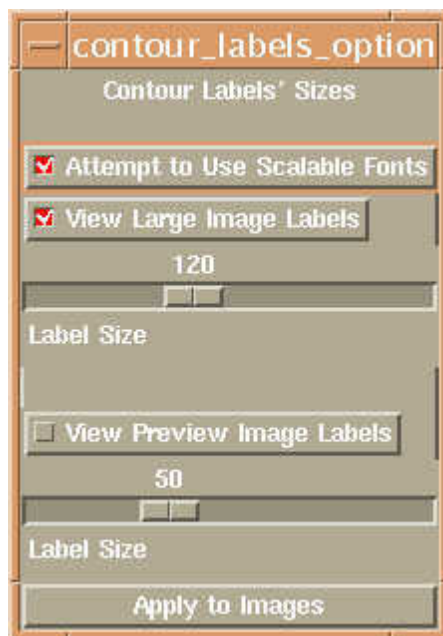


Figure 9.13. The Contour Labels Options window.

9.8. Loading Masks

Once an image or an image set is loaded, it is possible to load mask files. A mask file is a raster image containing the regions of the transport model corresponding to the appropriate medical image. This image is generated by seraPlan and will appear in a popup window (Figure 9.14). This image is used to remove contour lines from unwanted areas in the image, and at the same time, retain a smooth appearance for the existing contour lines. To perform this operation, position the cursor in one of the raster areas and click the mouse. The corresponding contour lines from the image window will be removed. An example is shown in Figure 9.15.

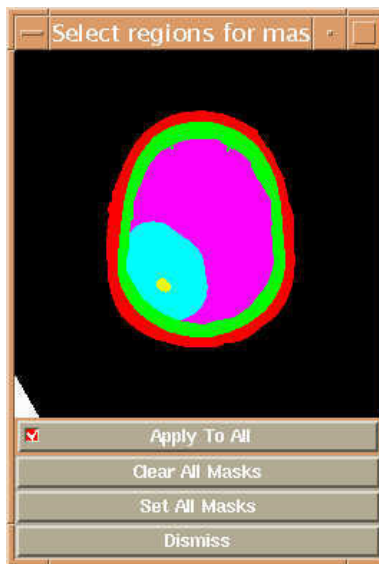


Figure 9.14. The seraDose Mask window.

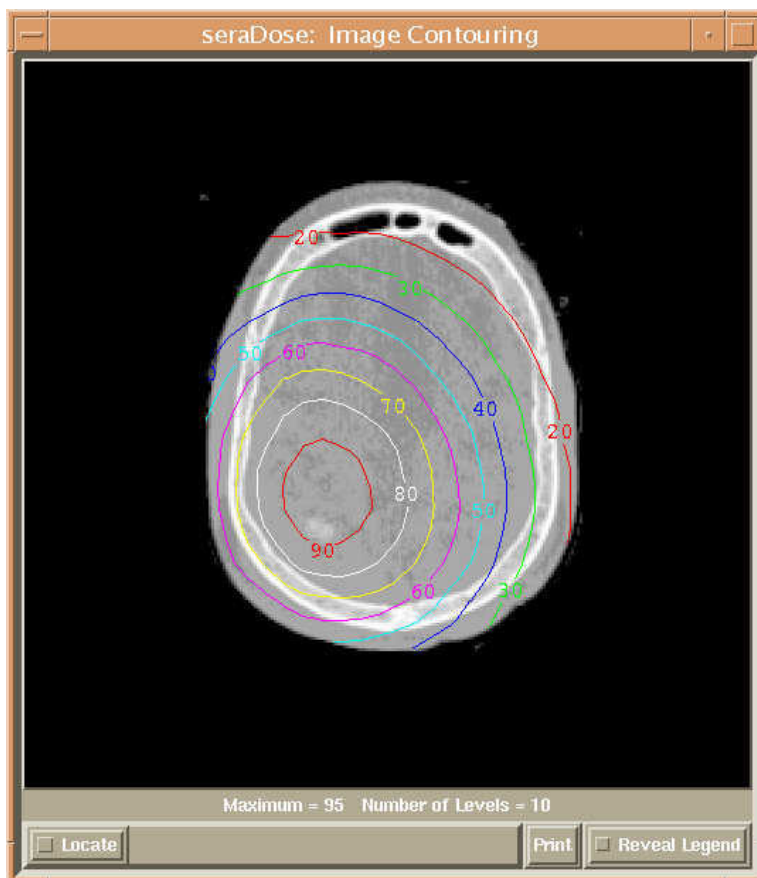


Figure 9.15. An image with contours after masking is applied.

There are several ways to load mask files. The ways to load mask files are very similar to those of loading contour files.

- Loading Single Mask Files - This option allows loading a single mask file. It is important to first select the image to which the mask file is to be applied. To accomplish

this, click on the image in the main seraDose window. Next, select Load Mask from the File menu. This option will bring up a file selection box where selection of one mask file can be made.

- Loading Mask in the Automatic or Smart Loader - After the files are selected in the Automatic Loader, all corresponding masks will be loaded. Similarly, the Smart Loader provides a convenient way to load mask files.
- Loading Mask on QSH Images - There are four ways to load masks on QSH images.
 - Load Mask - This is the same option as described above. It can be used to load any mask file on any image.
 - Load All Masks - User selects a directory containing single mask files. Selecting a directory is accomplished by selecting any file in the desired directory using the file selection box. SeraDose matches each mask file (ending in ,mask) on an image with the closest z value.
 - Load Mask on Current Image - User selects a directory containing single mask files. SeraDose searches the directory for the mask file with the closest z value to the current selected image.
 - Load 3D Mask File - Loads a seraZipped, concatenated mask file (ending in .cmf.sz). A header file (ending with .chd) that corresponds with the mask file must be in the same directory as the mask file.



Figure 9.16. Multiple ways of loading masks on QSH images.

9.9. Working with Masks

Once the masks have been loaded, several options are available:

- **Apply To All** - This toggle button is linked with the Automatic Updates button in the Options menu. When these buttons are toggled on, all operations are performed to the entire image set. Toggle them off, and changes only occur to the currently selected image.
- **Clear All Masks** - This option will clear any masks that have been previously applied.
- **Set All Masks** - This option will set all the masks. Essentially, this will cause all contours with corresponding masks to be removed from the images.
- **Dismiss** - Closes the Mask window (Figure 9.14). The Mask window will appear when an image is selected.
- **Masks Replace Images** - This option, found in the Options menu, will replace all the images with a gray scale mask image.

9.10. Working with Colors

There is a variety of options available to adjust colors to personal taste or to aid in image viewing.

- **Editing Contour Colors** - The colors of the dose contours can be edited if desired. To edit the contour colors, select Edit Contour Colors from the Color menu. Upon selecting this option the Contour Color Window will appear (Figure 9.17). To change the color of a contour, first select the contour in the list on the bottom left of the window. The contour's current color will be indicated on the right. To change the color, simply click on a different color button on the left. Once the colors have been adjusted, select Apply. Saving the colors for future runs of seraDose must be done in the Preferences window.
- **Reserving Color Cells** - To change the number of color cells used to display the gray scale image, select Reserve Color Cells from the Color menu. The default is 128 gray scale values.
- **Loading a Color Table** - It is possible to load a colormap other than the default colormap used for the images. To load a different colormap, select Load Color Table from the Color menu. Several colormap (cmap) files are provided in the SERA Resources/Shared/colormaps directory. See Figure 9.19 for an example of loading a different colormap.
- **Loading the Default Color Table** - To reload the default colormap, select Reload Default Color Table from the Color menu. SeraDose will load the default colormap depending on whether contour lines or colorwashed images are being viewed.
- **Adjusting Colors** - SeraDose provides a color adjustment tool that can be accessed by selecting Colormap Tool in the Color menu. The colormap tool (Figure 9.20) allows the user, via sliders, to select a new background level or saturation level, change the gamma correction, or rotate through the colors of the current colormap. Adjusting the sliders in this window can aid in viewing images. Color adjustment is performed interactively. To reset to the original values, click on the Reset button. To close the Color Adjustment window, click on the Done button.

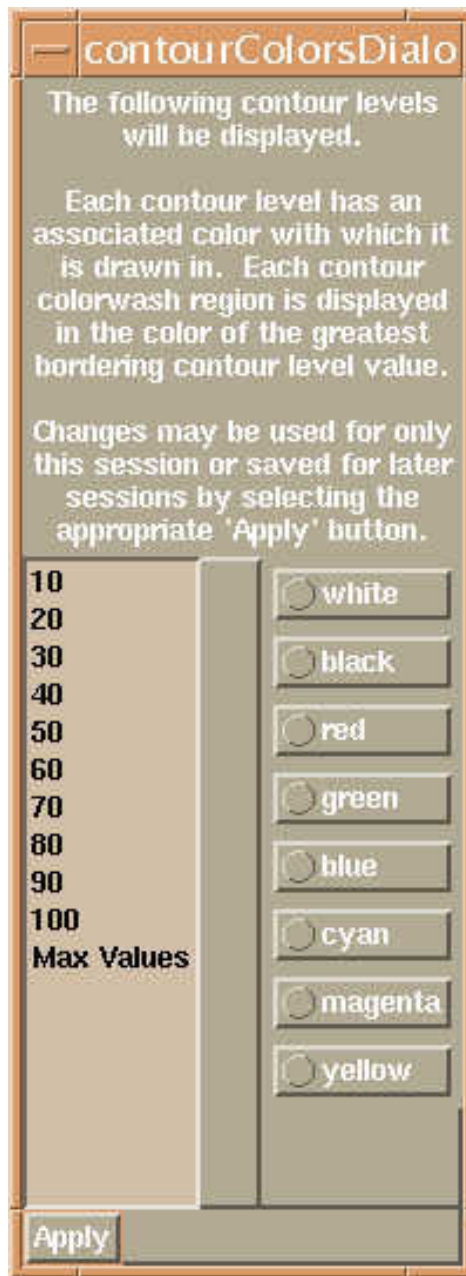


Figure 9.17. The contour color editing window.

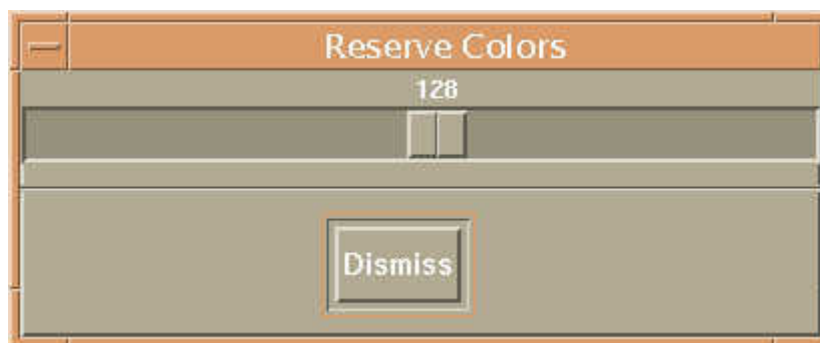


Figure 9.18. Multiple ways of loading masks on QSH images.



Figure 9.19.a. Image is the default colormap.

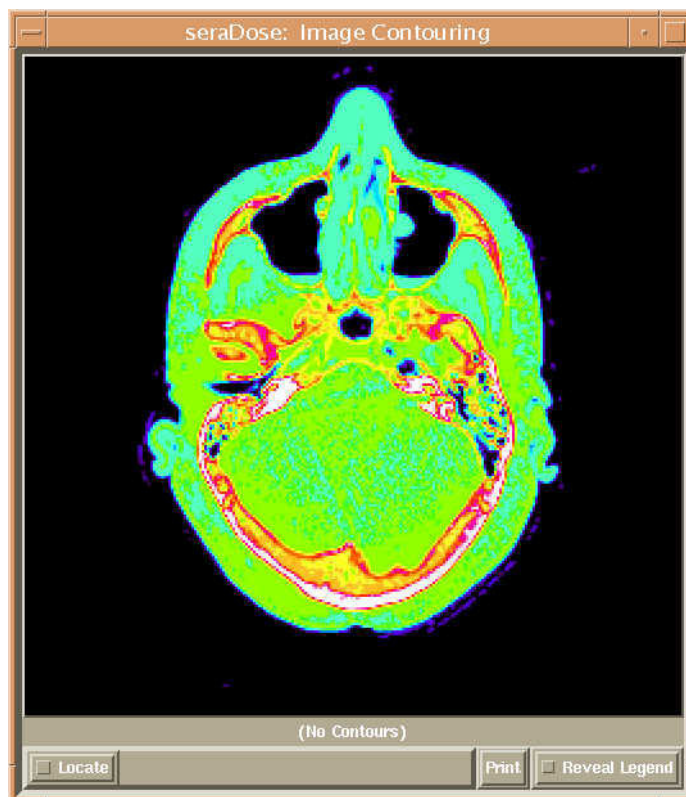


Figure 9.19.b. Image after loading sock.cmap.

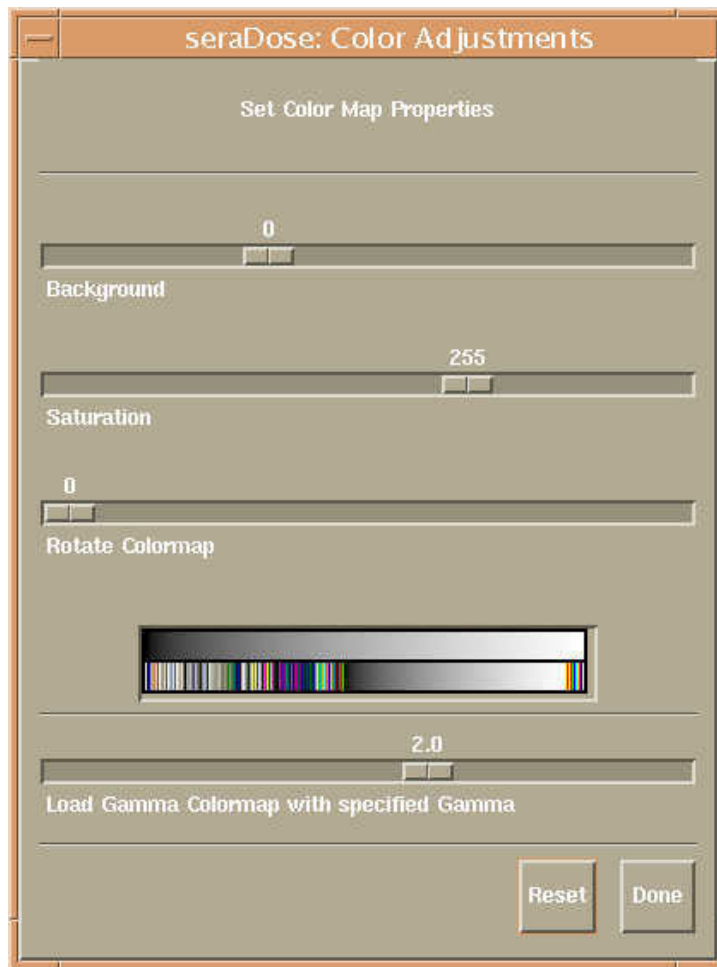


Figure 9.20. The Color Adjustment Window.

9.11. Preferences

The user can set and save individual preferences by selecting the Preferences option in the Options menu. This will bring up the Preferences window (Figure 9.21).

To save a change to the preferences, select a preference category and toggle on the individual preference to be saved. The next time seraDose is invoked, the preference will be as it was saved.

For example, the current default dose is Total Dose. The user wants to change the default dose to Boron Dose. To accomplish this, the user chooses Select Dose Component from the Options menu. From the Dose Options window, the user selects Boron Dose and clicks the Select button, then dismisses the window by clicking the Dismiss button. Now, the user chooses the Preferences option from the Options menu. Under the Defaults preferences, the default dose is now Boron Dose. In order to save Boron Dose as the default dose in the future, the user now toggles the Dose Component button on and then clicks Save.

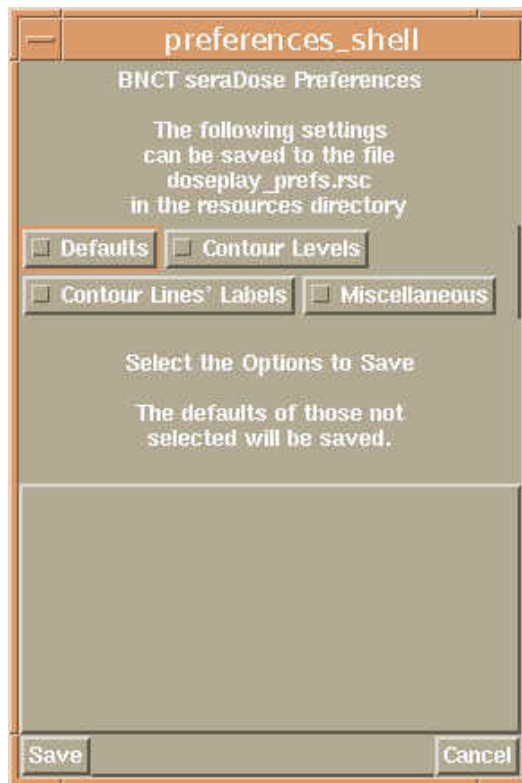


Figure 9.21. The Preferences window.

There are four preference areas in the Preferences window:

Defaults

- View Contour Lines (not colorwash)
- Masks Replace Images
- Dose Component

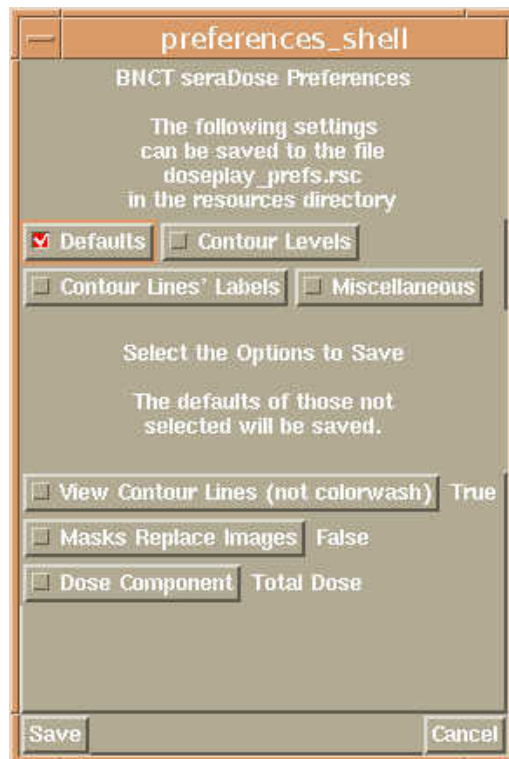


Figure 9.22. The Preferences window showing Defaults.

Contour Levels

- Default Contour Color
- Contour Levels



Figure 9.23. The Preferences window showing Contour Levels.

Contour Lines' Labels

- Attempt To Use Scalable Fonts
- View Large Image Labels
- Large Image Labels' Size
- View Preview Image Labels
- Preview Image Labels' Size

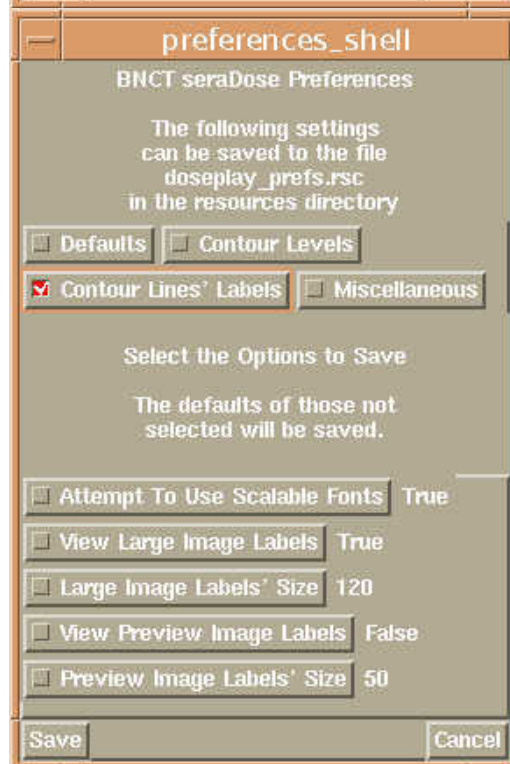


Figure 9.24. The Preferences window showing Contour Lines' Labels.

Miscellaneous

- Currently no preferences

9.12. Entering Dose Factors

This option allows the user to specify the dose factors and concentrations for each of the dose components. New reference values are computed and new dose values are displayed. To open the Dose Factor window, select Enter Dose Factors from the Options menu.

The Dose Factor window (Figure 9.25) contains three main sections:

- The first section contains patient information. The four fields in the patient section are filled with information found in the QSH header file (ending in .qhd). If information is missing in the header file or single images were loaded, the user can enter the information by clicking in the desired field and typing the information.
- The second section of the Dose Factor window is used to enter reference boron values. The Tissue to Blood value is taken from the body data file. The Tissue to Blood value for each body in the body data file (i.e., body_data.txt.sz) can be selected in the Body pull down menu. To enter a Reference Boron value, click in the Reference Boron text box and type in the value. To adjust the dose factors, click Apply. The factors in the third section of the Dose Factor window will be adjusted. To select a body data file different from the default file loaded, click Load File. The body data file must be a seraZipped (ending in.sz) text file.
- The third section contains six rows. The first four rows are for the primary BNCT dose components, the $^{10}\text{B}(n,\alpha)^7\text{Li}$ dose (Boron), the Gamma Dose, the $^{14}\text{N}(n,p)^{14}\text{C}$ dose (Nitrogen), and the Fast (Hydrogen Recoil) component. There are three columns for each component. These three columns display a Factor, a Concentration, and a Reference Value. After loading the contour file, seraDose reads the default data from the file. The reference values are those values for the edit voxel at the location of the reference value (the default reference volume is the voxel with the peak thermal flux). These reference values have been multiplied by the values read in for Factor and Concentration. The value for Factor is usually a Relative Biological Effectiveness (RBE) or a Compound Factor (CF). The values for Concentration vary depending on the component. For the boron dose, concentration is usually parts per million by weight (ppm). For the gamma dose, concentration is usually not meaningful and is set to unity. For the nitrogen dose, concentration is percent by weight. For the fast concentration, percent by weight of hydrogen is specified. It is usually less confusing to run the seraMC problem normalized to one MW-minute with all factors set to unity, with 1.0 ppm for the ^{10}B concentration, and with the actual concentrations for nitrogen and hydrogen. The fifth row is the total dose, or just the sum of the four components weighted by Factor and Concentration. Once the values have been entered, click on Update Contours to recompute the contours. To reset the Dose Factor Window to the default values, click on Set to Default. To close the window, click Dismiss. The Print button is not implemented.

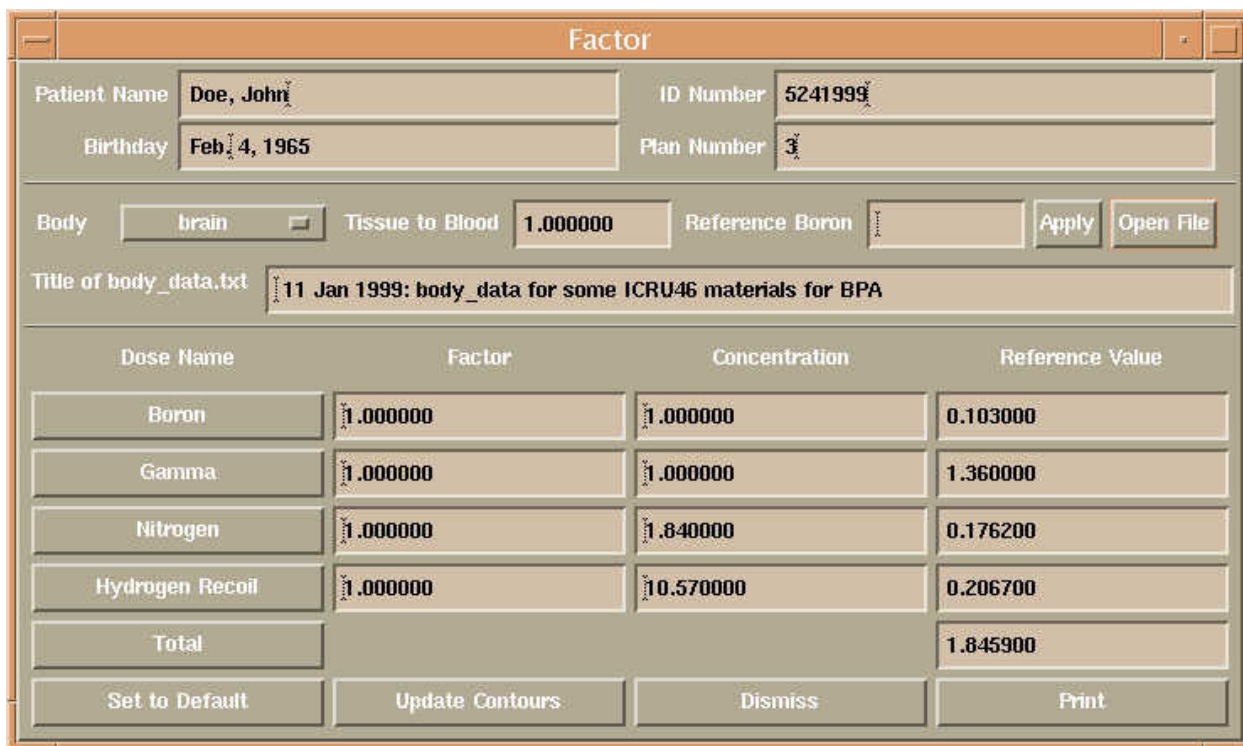


Figure 9.25. The Factor Window.

9.13 Selecting Dose Components

To select a dose component to view, choose Select Dose Component from the Options menu. As shown in Figure 9.26, this option allows the user to select which dose or fluence component to display, including:

- Boron Dose
- Gamma Dose
- Nitrogen Dose
- Fast Dose
- Group 1 Fluence
- Group 2 Fluence
- Thermal Fluence
- Other Dose
- Total Dose



Figure 9.26. The Dose Options Window used for selecting which dose component to view.

9.14 Changing Field of View

To change the field of view of an image, select Change Field of View from the Options menu. This allows the user to enter a new field of view (FOV) value in mm. If a QSH image set is loaded, the field of view is automatically calculated. If you have loaded raw images, the default is 250mm. The field of view must be calculated or entered before using the Locate option in the Image Contouring Window.

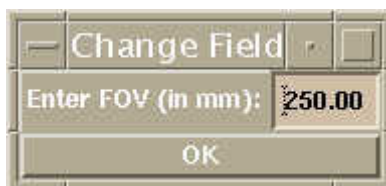


Figure 9.27. The Change Field of View window.

9.15 The Image Contouring Window

The Image Contouring Window displays a large view of the current image. To change the image shown in the Image Contouring Window, select the image in the main seraDose window simply by clicking on the desired image. The Image Contouring Window includes the following features:

- Locate - Clicking the Locate button will display the cursor location within the image as the cursor passes over the image in the Image Contouring Window. The cursor location is dependent upon the field of view. The location, $x = 0.0$, $y = 0.0$, is centered in the image. If contours are loaded, the dose value at the current cursor location is also displayed.
- Info - Clicking the Info button will display information about the current image and the current dose, and information found in the site file. The site file is located in `$$SERA_RESOURCES/Site/` and is named `info.txt`.
- Print - Allows user to print the Image Contour Window to a printer or to a file.
- Reveal Legend - Extends the window to show the contour legend. From within this window, the contour values can be changed by clicking on Change Contour Level Values. This button will open the Contour Levels Window. The colors can also be adjusted by clicking on Change Contour Level Colors. This button will open the Contour Colors Window.

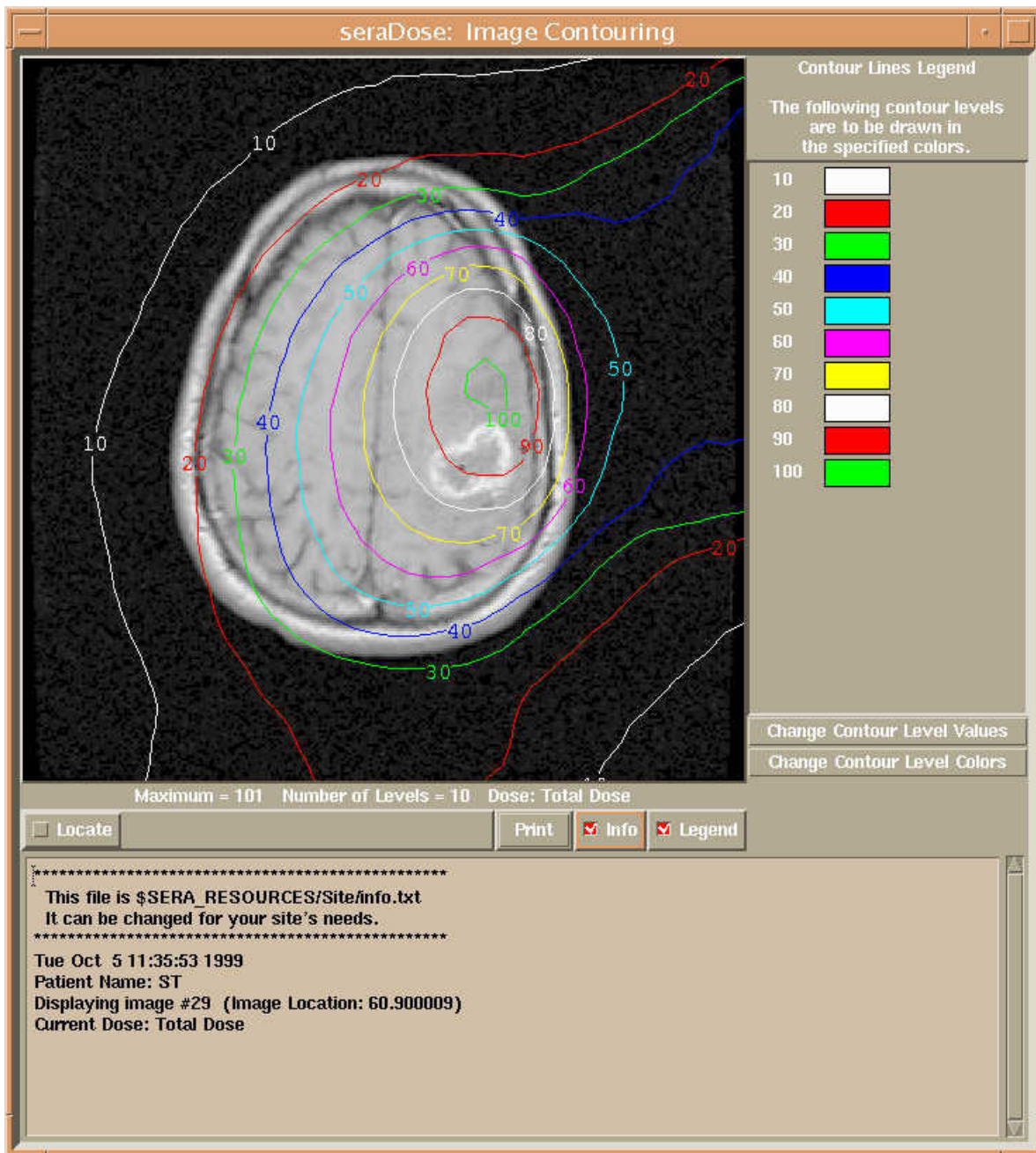


Figure 9.28. The Image Contouring Window.

10. 2D Dose Plotting Tools

10.1. Introduction

After a plan is developed, two separate tools are used to plot the dose results, and a third tool is used to determine the patient positioning. These tools are located in \$SERA_BIN, and all are currently launchable from the command line. The plotting tools may also be launched from the SERA main menu or cross launched from any other SERA module by selecting the desired action. The two plotting modules all invoke the xmgr two-dimensional plotting code to display the results of the request.

Note: the plotting tools have been re-written, and they are no longer compatible with output files from older (pre-1B0) versions of seraMC. For these tools to be used with previous seraMC runs, the edits MUST be re-performed with seraPlan.

10.2. Simplot

Simplot is an ancillary program that reads the seraPlan line-edit output file (.lin) and invokes the xmgr plot code for each instance of a line edit.

In the line edit, dose/depth data are available for:

- 0 Total dose
- 1 ¹⁰B dose
- 2 Gamma dose
- 3 ¹⁴N dose
- 4 Hydrogen (fast) dose
- 5 Other dose
- 6 Group 1 (fast) fluence
- 7 Group 2 (epithermal) fluence
- 8 Thermal fluence
- 9 Gamma production
- a Ultrafast gamma production
- b Reaction rate 1
- c Reaction rate 2

where the leading single digit (in hex) represents the code provided to force that dose component to be plotted.

As an example, the simplot program might be invoked with:

```
simplot case1.lin case1.xmgr 2 96 12.3 0 2 3 4
```

Here, case1.lin is the seraMC output file, and case1.xmgr is a new file that will contain the plot commands and data for xmgr. If there is more than one plot, simplot adds an increment to the plot file name.

Simplot begins generating plot data from 0.0 to the depth obtained from the last point in the line plot. The third argument to simplot (2) is a sample rate, or stride distance. This specifies that every other point in the line plot is used for the plot. The next two arguments (96 and 12.3) are the normalization factor and boron concentration to be folded into the dose data. Generally, the normalization factor is either 1, for no normalization, or the exposure in MW-min. The boron

concentration is applied only to the ^{10}B dose and the total dose, where the total dose is the sum of all the dose components.

The final 0 2 3 4 argument tells simplot to provide plots for the total dose, gamma dose, the ^{14}N dose and the fast dose. An xmgr window is then opened for each line edit instance, and the user can modify the title and perform any other editing of the plot. Then the user can print the plot and save the plot file using the capabilities of xmgr.

If one simply enters simplot, the routine will prompt the user for the necessary input data, and then present the requested components in an xmgr window.

Simplot may be executed from the main menu by selecting Dose-Depth Plot under the SeraPlot menu (see Figure 3.6). A widget is then presented to the user as shown in Figure 10.1. Once the Start button is selected, plots of the line edits will appear. This widget may also be called from the command line by typing runSimplot.

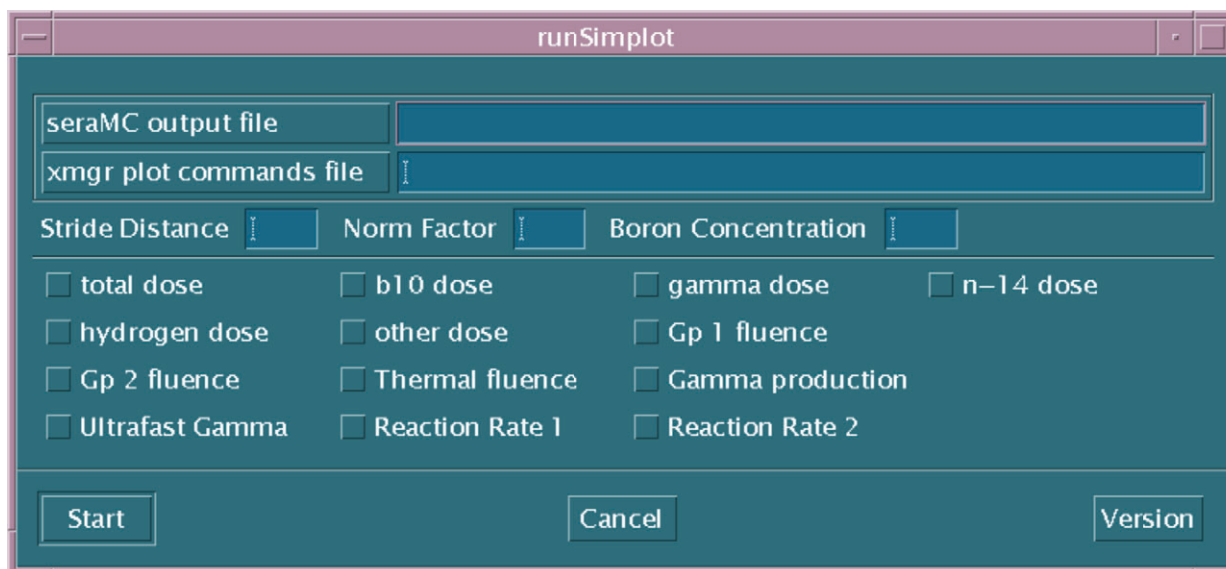


Figure 10.1: Simplot Widget.

10.3. DVall

DVall is an ancillary program that reads the seraPlan dose-volume histogram output file (.dvh) from up to six different cases and provides dose-volume histogram plots for as many as four selected bodies on a single page. This allows multiple treatment plans to be compared simultaneously and easily.

The DVall code is entirely widget-driven and cannot be run directly from the command line. To invoke the code, either type DVall from the command line, or select DVall from the Dosimetry Display menu of the main menu. This will pop up the widget, which is shown in Figure 10.2.

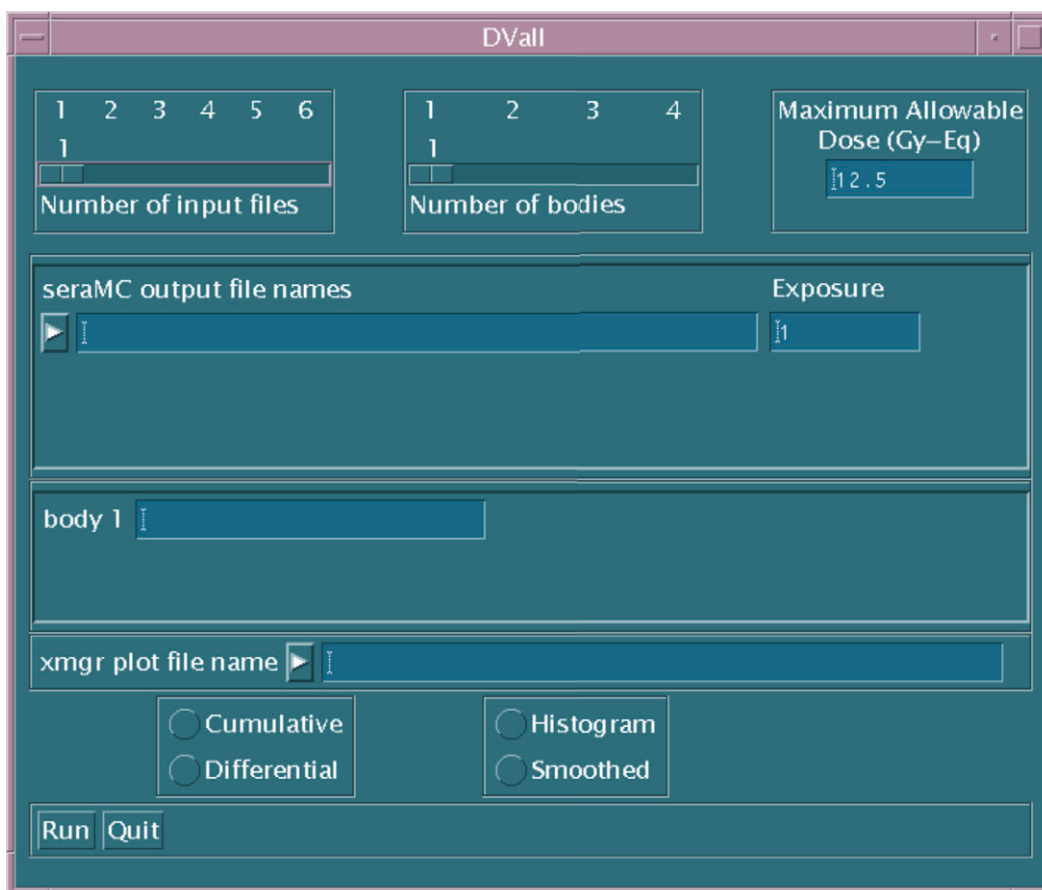


Figure 10.2. DVall Widget.

The widget interface asks the user to provide the number and name of each seraMC output file and the number and name of each body in these files. The body names must be the same in all seraMC cases; DVall will not pick up the information for non-matching body names. The Maximum Allowable Dose input parameter causes DVall to calculate the exposure time necessary to achieve this dose at the reference point in healthy tissue. This information is written to the output file (`plot_file_name.out`).

Dose-volume data may be displayed as cumulative or differential dose-volume plots and in histogram or smoothed form. After the user has supplied the necessary information to the widget interface, the code is executed by selecting the Run button from the bottom of the widget. DVall then produces an xmgr plot file, written to `plot_file_name.xmgr`, containing the dose-volume histogram data for all requested cases and bodies. A sample plot output for two cases and three bodies is shown in Figure 10.3.

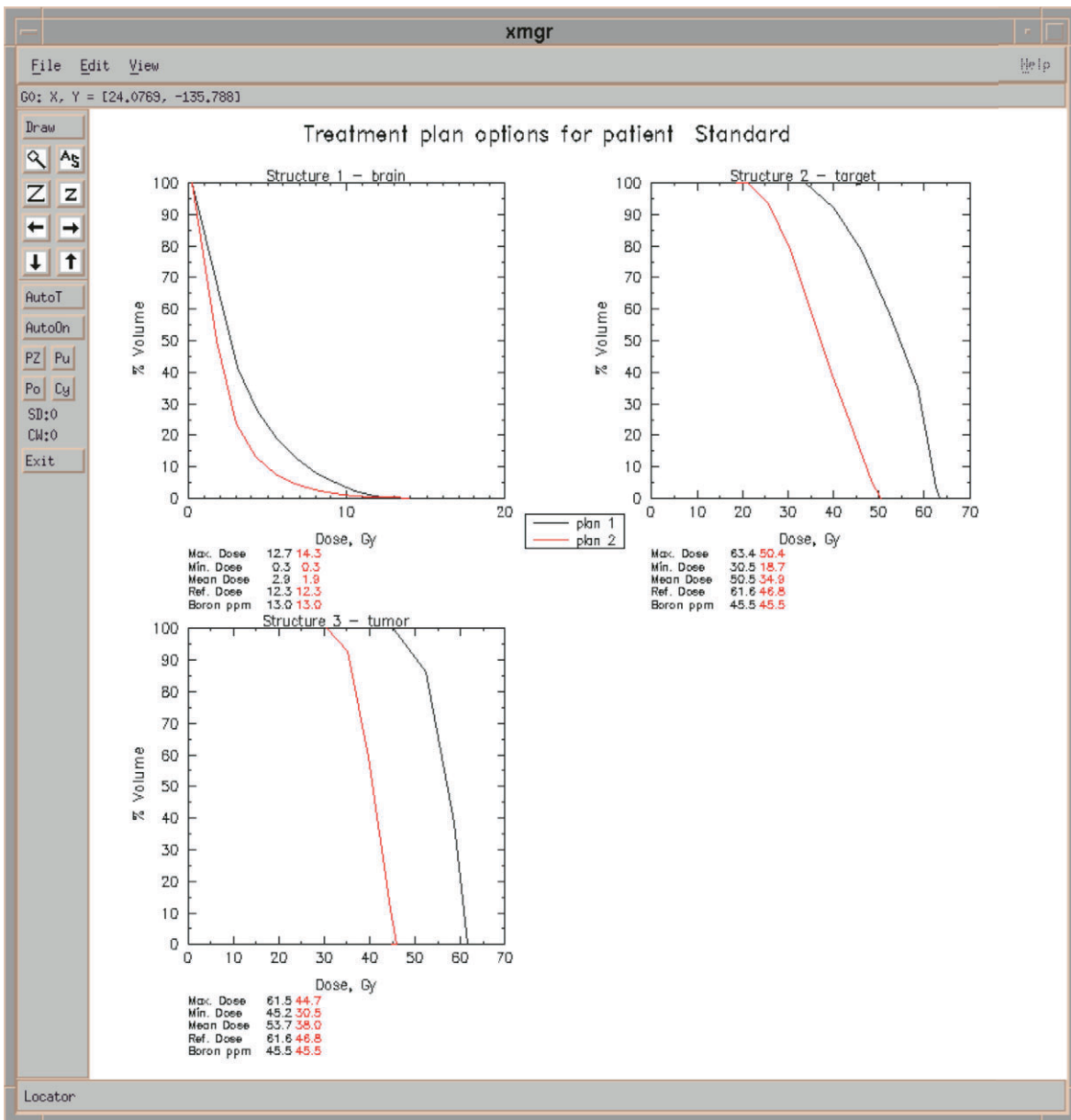


Figure 10.3. DVall plot display.

10.4. Patient Positioning

Patient positioning for BCNT currently employs a system of beads and tattoos placed on the patient during final imaging procedures. This method is simple to implement and should be sufficiently accurate if care is taken. It is also very laborious to position the patient using this method and it is likely that the method will be either improved or replaced.

Because the preferred patient positioning method has yet to be determined, patient positioning is treated in an ad hoc fashion, and both the source and executable files used for patient positioning are supplied with the software distribution. The source is being supplied until a preferred method is determined so that the field changes may be made if necessary. To run this code, enter from the command line:

position input_file

The user will be prompted for the x-, y-, and z-coordinates of the various beads. These coordinates are obtained by using the locate button on the slice widgets displaying the beads.

The input file may appear as:

THETA	PHI	numBEADS	
-13.0	75.0	4	
TARGET	X	Y	Z
	-14.00	41.80	48.00
BEAM	X	Y	Z
	-20.92	71.77	56.24,

where THETA is the azimuthal patient rotational angle and PHI is the polar rotational angle.

A more detailed discussion of the coordinate system used is available in the seraMC manual (see Appendices A-D), but note that, by necessity, the beam is rotated instead of the patient, so that the patient rotational angles are specified counter to seraMC angles. The target and beam coordinates are generated by seraMC, and the beam values are given with the beam rotated. The target point is the origin of rotation, so it is invariant.

The position code rotates the beam back to its original position. Then it rotates the patient (beads) by the angles specified in the input file. The final bead coordinates are expressed relative to the beam exit point.

Admittedly, this is a cumbersome and complicated process, and it is necessary to provide one-on-one training of the user. Much work remains to get this process automated.

11. System Requirements

- Unix O/S -- (Solaris SPARC 2.x, Solaris X86 2.x, Linux X86)
- X Windows environment (X11R6)
- Motif 1.2
- minimum 128 Mbyte RAM -- developed and tested using 128-256 Mbyte RAM; the amount desired will of course depend on the size of the image sets loaded
- minimum 100 Mbyte disk space -- 100 Mbyte disk space is a minimum and the actual amount required depends on image and data file requirements.

12. Troubleshooting

- Make certain Environment Variables are set up correctly.
- Check whether .serarc file exists -- sets up the environment variables.
- \$SERA_HOME/Target/bin directory exists and contains valid executables.
- \$SERA_HOME/Shared and \$SERA_HOME/Resources directories exist.
- When encountering difficulties with any of the SERA modules, it is recommended that the programs be reloaded rather than simply reset. Though the internal routines should allow the program to be reset (rather than reloaded), the fear of potential memory leaks and the difficulty of reinitializing or resetting the program state make it safer simply to exit and reload the program when beginning a new set of work. Though no known problems exist, the thinking is that it is simply safer to begin afresh whenever appropriate.

13. References

1. M.W. Frandsen, D.E.Wessol, and F.J. Wheeler, *Rapid Geometry Interrogation for a Uniform Volume Element-Based Monte Carlo Particle Transport Simulation*, 1998 ANS Radiation Protection and Shielding Division Topical Conference, Volume 2, April 19-23, 1998.
2. F. J. Wheeler, *Radiation Transport in Tissue by Monte Carlo*, EGG-BNCT-11178, January 1994.
3. R. W. Scheifler, J. Gettys, and R. Newman, *X Window System: C Library and Protocol Reference*, Digital Press, 1988.
4. Open Software Foundation, *OSF/Motif Programmers Reference*, Prentice Hall, 1991.
5. *OpenGL, The Industry's Foundation for High Performance Graphics*, ArtLab, Oct 15, 1997, (<http://www.opengl.org>.)
6. G. Q. Maguire, Jr. and M. E. Noz, *QSH Reference Manual*, Columbia University, Department of Computer Science; New York University, Department of Radiology, 1990.
7. B. S. Baxter, L. Hitchner, and G. Q. Maguire, Jr., *A Standard Format for Digital Image Exchange*, AAPM Report No. 10, American Institute for Physics, 1982.
8. M. E. Noz and G. Q. Maguire, Jr., *QSH: A Minimal but Highly Portable Image Display and Handling Toolkit*, *Computer Methods and Programs in Biomedicine*, Vol. 27, 1988, pp. 229-240.
9. W. Schroeder and B Lorenson, *3-D Surface Contours*, *Dr. Dobbs Journal*, p. 26, July 1996.

Appendix A. Transport Calculation Module (seraMC)

A.1. Introduction

The seraMC program is specifically tailored for the simulation of neutron irradiation and the determination of absorbed dose distributions for research and patient treatment planning. The SERA system provides a means to develop a three-dimensional geometry from medical images, apply a simulated neutron and gamma beam, and determine three-dimensional dose patterns. These neutron and gamma dose distributions are displayed via zero-, one-, two-, and three-dimensional edits which can be viewed using the display systems of the SERA system. The seraMC program can also be operated in stand-alone mode using geometry constructs based only on Combinatorial Geometry (CG).

The program is targeted toward boron neutron capture therapy (BNCT) using an epithermal neutron beam but can be adapted for other types of beams (e.g., thermal neutrons, fast neutrons, x-rays, and gamma rays) and/or other binary agents (e.g., gadolinium). Modification of the software may be required for optimization in these other modes. Other transport programs could also conceivably be used in conjunction with SERA for purposes of validation or independent assessment. Modification of these programs' outputs would probably be required to interface with SERA.

There were several motivating factors for development of a new Monte Carlo (MC) transport code specifically for clinical BNCT treatment planning. First, existing Monte Carlo programs are usually targeted toward general applications or toward specific applications for other ends and are not necessarily efficient or adequate for BNCT applications. Creating a new code allows generality to be utilized where needed, but it allows tailoring of the code structure and computational efficiency for BNCT where applicable. Second, detailed results are required for BNCT. Most MC programs provide only volume edits and sparse, special purpose point or surface edits. Therefore, if the required detail is to be obtained, a cumbersome amount of physical modeling is necessary and computer requirements become large. An entirely new code would not be burdened with the need for general tallies, as the desired tallies for BNCT are well known in advance and rarely, if ever, change. Third, anatomical features are generally not amenable to reconstructions using analytical surface representations. Thus, approximations must be made to the geometry. The seraMC module allows the geometry to be represented as a set of uniform volume elements (univels) that may be as small as the volume represented by an individual pixel extended over a single image plane. Since representation of a structure (e.g., a patient's head) may require millions of univels, a method was developed to rapidly perform the geometric ray tracing required by the Monte Carlo process. The method developed requires an order of magnitude less computation time than the earlier B-spline geometric representations. Finally, a generalized editing capability is required, and the ability to re-edit a previous run with new edit and spatial directives can be useful. SeraMC has built-in editing capability for zero-, one-, two-, and three-dimensional edits that can provide all information required. The output from a run is stored in a file that can be used to obtain any additional edits desired after the run is completed.

In addition, other features enhancing efficiency, usability, or function have been incorporated. For example, special biasing techniques are available to rapidly converge the recalcitrant fast neutron flux distributions, and outputs can be combined and normalized to an appropriate beam output with little user effort.

A.2. General Problem Input

This section provides a description of the various general problem input parameters. These inputs control the number of histories to simulate, beam direction, edit mesh, and the files to be created and accessed during execution. The parameters are described in the order they appear in the input file.

A.2.1. Line 1 - Title

The title card allows a single descriptive line, up to 80 characters in length, to be entered. The title card is used throughout the output and in the restart file for identification purposes.

A.2.2. Line 2 - nbatch, nhist, wncut

The first two entries on Card 2 define the total number of particle histories to start from the source. The entry nbatch is the number of statistically independent batches of particles, and nhist is the number of particles per batch. These parameters also control the number of particles started for the biased fast neutron component, if such is requested.

The third entry on Card 2, wncut, defines the weight cutoff for the transport algorithm. Each collision reduces the particle weight, initially one, by a factor defined as the ratio of the scattering cross section to the total cross section. This accounts for the effects of absorption, while allowing all particles to make significant contributions to the tallies, efficiently improving the convergence of the edit tallies. When the particle weight drops below wncut, the particle is killed. The default value for wncut is 0.01.

A.2.3. Line 3 - Target Point

Positioning of the beam requires a target point to be specified, which is the point about which the beam is rotated when the beam is positioned. The center of the beam passes through this point. This point is specified as a triplet of values (xp,yp,zp), with all three dimensions given in centimeters. Note that the entire source beam geometry, as specified in the source file, is rotated about the target point.

A.2.4. Line 4 - Zb, phi, theta

Positioning of the beam requires the distance from the source to the target point to be specified. Zb is this distance in centimeters from the target point (xp,yp,zp) to the center of the source plane. For all cases, including the initial input value for searches, the value of Zb must be such that beam geometry and patient geometry are separated by the buffer region and the patient geometry is completely enclosed by the buffer. Otherwise, geometry problems will occur, resulting in lost particles and/or terminations from errors. Figure A.1 shows how Zb helps determine the beam position.

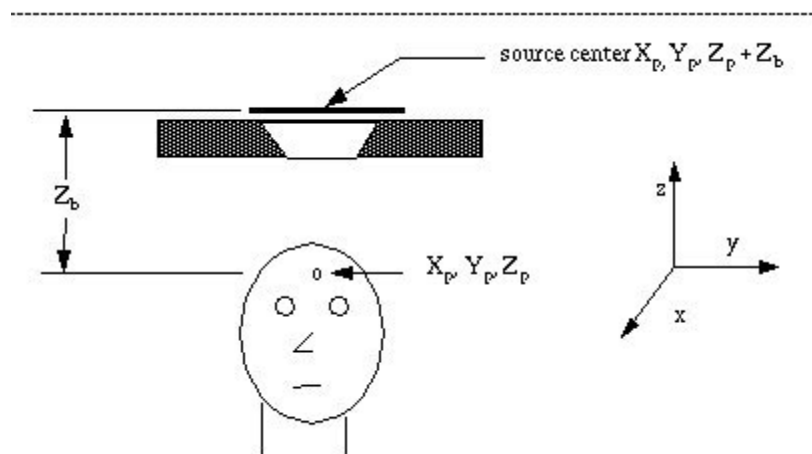


Figure A.1. Example beam positioning showing the target point and Z_b .

The angle ϕ is the polar angle from the reference beam location at the top of the head to the current beam location, with the superior axis used as the reference direction; the angle θ is the azimuthal angle from the right side of the head to the current beam location. Note that positive values of θ rotate the beam towards the front of the head, and negative values rotate it towards the back of the head. The default value of θ is zero. In Figure A.2, where ϕ is set at 90 degrees and θ is unspecified (i.e., left at the default value), the beam is rotated from the superior axis to the right side of the head. In Figure A.3, where the previous polar angle setting is supplemented by adjusting θ to -90 degrees, the beam is rotated from the right axis to the back of the head.

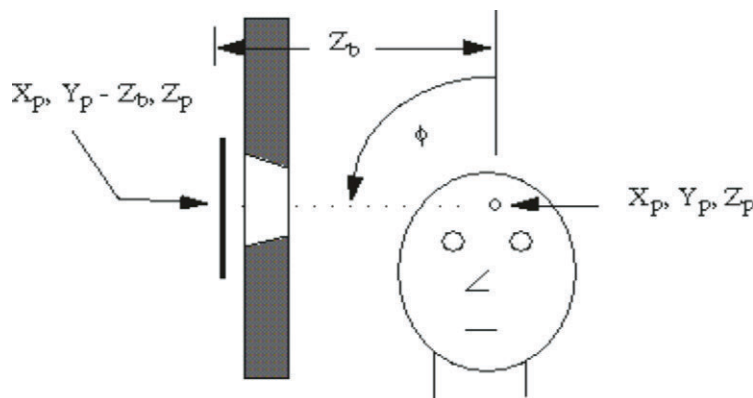


Figure A.2. Beam positioning showing polar angle ϕ .

A.2.5. Line 5 - Combinatorial Geometry (CG) file

The Combinatorial Geometry (CG) file consists of a description of the geometry included in neither the source file nor the univel file. When univel geometry is used, the CG geometry file may be a null file, or it may contain information pertaining to the patient (e.g., shields, table, etc.) that supplements the univel geometry. The CG body and region descriptions and options available are provided in Appendix B. The CG input consists of a title line, containing up to 80 characters of descriptive information, and records which assign region indices and material names. These indices must be consistent with region indices in the source file. A comment line, which is any line with the # character in column 1, may be placed anywhere in the file.

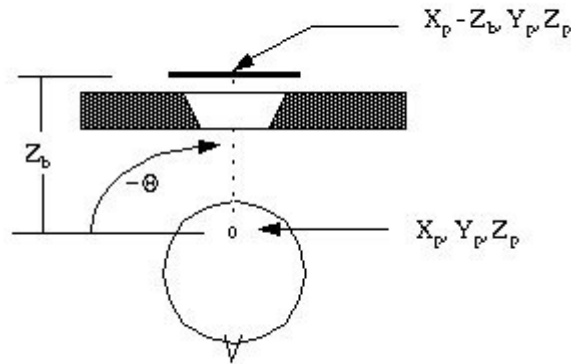


Figure A.3. Beam positioning showing the azimuthal angle theta.

A.2.6. Line 6 - Univel file

Patient, or univel (.uv), geometry is that defined using the seraModel reconstruction software, which may be used to read medical images (e.g. MRI, CT) and construct 3D models of the patient. Output from SeraModel consists of the files xx.uv and xx.uvh that are read by seraMC and used in the radiation transport calculation. Region indices from nreg_cg+1 to nreg, where nreg_cg is the number of CG bodies and nreg is the total number of bodies, are assigned to the univel regions in the order determined by the 3D reconstruction. Only the .uv file is specified on this line as the .uvh file uses the same prefix and will be read automatically.

All univels must have a region name (assigned during reconstruction). All regions (except buffer) must be assigned a material name. Those univels not assigned to a region will be designated unnamed, and a warning message will be issued by SeraModel, when a model is saved with unnamed univels. A model can be saved with unnamed univels as model construction proceeds, but cannot be used in a transport calculation.

When constructing the model with SeraModel, it is necessary to assign all space outside the patient anatomy to the buffer. Be careful to generate no internal buffer zones, as they could cause errors in the transport calculation.

A.2.7. Line 7 - nedit2 and nnegp

Line 7 contains two parameters. The first, nedit2, is the number of voxels along each coordinate axis of the edit mesh array, with the assumption that the edit mesh is a cubic grid of uniform cubic voxels. The default value for this parameter is 30 and the maximum is 120. The second value, nnegp, is the number of neutron energy groups specified for edits. The default value is 3, and the performance for values other than 3 is not guaranteed. Generally, the three groups are defined as thermal, epithermal, and fast.

A.2.8. Line 8 - Neutron energy group boundaries

This card gives the lower energy boundaries for the nnegp edit neutron energy groups. The boundaries are given as the group numbers of the 94-group seraMC neutron cross section energy structure. The default values are 32, 72, and 94, which set the boundaries at 9.12keV, 0.414eV, and 0.0eV. The neutron energy group structure is given in Table A.1.

Table A.1. Upper neutron energy boundaries for the seraMC 94-group neutron cross section library.

Group	Energy (eV)	Group	Energy (eV)	Group	Energy (eV)	Group	Energy (eV)	Group	Energy (eV)
1	1.69046E+07	2	1.49182E+07	3	1.34986E+07	4	1.19125E+07	5	1.00000E+07
6	7.78801E+06	7	6.06531E+06	8	4.72367E+06	9	3.67879E+06	10	2.86505E+06
11	2.23130E+06	12	1.73774E+06	13	1.35335E+06	14	1.05399E+06	15	8.20850E+05
16	6.39279E+05	17	4.97871E+05	18	3.87742E+05	19	3.01974E+05	20	2.35177E+05
21	1.83156E+05	22	1.42642E+05	23	1.11090E+05	24	8.65170E+04	25	6.73795E+04
26	5.24752E+04	27	4.08677E+04	28	3.18278E+04	29	2.47875E+04	30	1.93045E+04
31	1.50344E+04	32	1.17088E+04	33	9.11882E+03	34	7.10174E+03	35	5.53084E+03
36	4.30743E+03	37	3.35463E+03	38	2.61259E+03	39	2.03468E+03	40	1.58461E+03
41	1.23410E+03	42	9.61117E+02	43	7.48518E+02	44	5.82947E+02	45	4.53999E+02
46	3.53575E+02	47	2.75365E+02	48	2.14454E+02	49	1.67017E+02	50	1.30073E+02
51	1.01301E+02	51	7.88932E+01	53	6.14421E+01	54	4.78512E+01	55	3.72665E+01
56	2.90232E+01	57	2.26033E+01	58	1.76035E+01	59	1.37096E+01	60	1.06770E+01
61	8.31529E+00	62	6.47595E+00	63	5.04348E+00	64	3.92786E+00	65	3.05902E+00
66	2.38237E+00	67	1.85539E+00	68	1.44498E+00	69	1.12535E+00	70	8.76425E-01
71	6.82560E-01	72	5.31579E-01	73	4.13994E-01	74	3.80000E-01	75	3.60000E-01
76	3.30000E-01	77	3.10000E-01	78	3.00000E-01	79	2.90000E-01	80	2.80000E-01
81	2.60000E-01	82	2.40000E-01	83	2.20000E-01	84	1.80000E-01	85	1.60000E-01
86	1.40000E-01	87	1.00000E-01	88	8.00000E-02	89	6.00000E-02	90	4.00000E-02
91	2.53000E-02	92	1.50000E-02	93	7.00000E-03	94	4.00000E-03		

A.2.9. Line 9 - X0, Y0, Z0

These values give the origin of the edit voxel mesh. To provide the detail required by treatment planning, the model geometry is enclosed within a three-dimensional array of edit cubes (voxels). In seraMC, there are nedit x nedit x nedit cubes in a regular array, each having dimension delw. The user must specify an origin and a dimension delw (usually 1.0 cm) so that the edit mesh entirely encloses the model of the patient. As an example, if the model is centered about (0,0,0), then specifying the voxel mesh origin at -15,-15,-15 and delw = 1.0 would provide edits for the range $x = -15.0$ to $x = 15.0$, $y = -15.0$ to $y = 15.0$, and $z = -15.0$ to $z = 15.0$.

A.2.10. Line 10 - delw, ntrk

The parameter delw is the size of each cubic voxel within the edit mesh. The default is 1.0cm. A second parameter, ntrk, is also included on this card. If ntrk is entered as 1, then particle track information will be written to a file input.pp (where input is the input file name prefix) for display in sera3d. This option is equivalent to the Test mode in seraCalc.

A.2.11. Line 11 - Old restart file

The seraMC software writes case information and all of the edit mesh tallies to a file, usually with .rst appended to the name. Regardless of the form of the filename specified, seraMC writes the file for the first transport mode. If subsequent modes are specified (see Section A.2.15), subsequent output files are written to filenameea, filenameeb, etc. The run directives are commonly NFGD, which would result in a filename file, a filenameea file, and a filenameeb file. After a new restart file is written, the previous version is discarded, since all the information is

written to the new file. For edit-only runs, or other cases where a restart is being performed, a filename must be specified. For fresh runs, the value should be given as **none**.

A.2.12. Line 12 - New restart file

The new restart file behaves in a similar fashion to the old restart file. Both filenames may be specified in the case of a restart run, but generally only one is given. For a fresh run, the new restart filename must be supplied. For edit-only runs, the new restart filename is given as none.

A.2.13. Line 13 - Source file

A description of the incident coupled neutron-gamma field is required to perform the transport calculation. The seraMC source file provides this description. A description of the source file, and the accompanying formats, is found in Appendix C. A source file must be specified for all runs, regardless of the directives specified.

A.2.14. Line 14 - irand_num and ied

Particle transport in seraMC is accomplished by simulating many pseudo-histories of particles using machine-generated random number sequences. The value `irand_num` is used to determine the initial seed and the stride distance. A unique value of `irand_num` will provide a unique random number string and an uncorrelated tally estimate so that results from two or more runs can be combined without introducing bias. The recommended values for `irand_num` are from 1 to 10, inclusive. If the value of `irand_num` is more than 10, unnecessary additional computer time is required.

The parameter `ied` allows the knowledgeable user to write debug edits in seraMC. The default value, 0, prints the normal output edits. If a value of 1 is specified, then voluminous and rather ugly debugging output is printed. It is recommended that the default value be used at all times.

A.2.15. Line 15 - run_dir and run_date

SeraMC is designed to run in many modes, combining the result of modes when appropriate. For example, it is common to set `run_dir = NFGD`, which specifies a standard neutron calculation, followed by a biased fast neutron calculation for better convergence of fast dose, followed by a gamma calculation, where both the beam gammas and the induced gammas are tracked, followed by an edit run. After the calculation in each mode is finished, a restart file is written. Table A.2 shows the different modes and combinations of modes that may be specified.

Table A.2. Allowed run directives and combinations for seraMC.

Allowed modes		Allowed combinations
N	normal neutron calculation	NFGD, NFD, NGD, FGD
F	biased fast neutron calculation	UPGD, UPD, UGD, PGD
G	gamma calculation	ND, FD, GD, UD, PD
U	ultrafast neutron calculation	
P	ultrafast recoil proton tracking	Note: can also run without the trailing 'D'
D	edit	

Note: When the F mode is specified, the contribution to the hydrogen dose is made from incident neutron energies greater than the lower energy of group igned(1). For example, if igned(1) = 32, the hydrogen dose above 9.118keV is tallied; for igned(1) = 70, the hydrogen dose above 0.68eV is tallied. When the P mode is specified, the contribution to the proton recoil dose is from incident energies above 16.9MeV. The group 1 flux will be tallied from the lower energy of group igned(1). Specifying igned(1)=0 will tally all energies above 16.9MeV.

The run_date variable allows the user to specify a date, which is written to the restart file for identification purposes only. No computation is performed with the date.

A.2.16. Line 16 - Materials File

In seraMC, materials are specified by a character name instead of listing isotopes and elemental (isotopic) densities. The densities are stored in a materials file that may be modified by the user to specify compositions. The Materials File is simply a compilation of composition definitions with associated names for the various materials available. In seraMC, materials are assigned to regions using character descriptors (e.g. brain, water) that must be defined in the Materials File. The input data for this file are quite simple and in fixed format. The input consists of data items defined in Table A.3. This table defines the isotopes and atomic densities for defined materials. The list of possible isotopes is given in Table A.4, and the list of materials presently in the seraMC materials file is shown in Table A.5. The file may contain comment lines (lines with a # character in column 1). The data entries are straightforward, except that it is possible to have confusion about the way in which thermal scattering data are incorporated. An example for a simple molecule (H₂O) will help to show how the data are retrieved from the library. This molecule can be specified by setting id_one = 1 (free hydrogen), with dens = 0.06668 and id_two = 3 (free oxygen for above-thermal energies and H₂O for thermal energies). Thus the specification for H₂O could be:

```
H2O
1
1 0.06668 3
```

The number of primary atoms per molecule is given in the library so the code knows to divide the hydrogen density by two in order to specify the oxygen density or the molecular density. Other examples, such as CH₂, are available in the Materials File.

Table A.3. Description of Materials File format

Record	Variable	Format	Description of value
1	lmat	i12	Total number of materials in the materials file
2	mati	1x,a40	Name of material
2a	niso_m	i12	Number of isotope identifier/density pairs for this material
3	id_one(i) dens(i) id_two(i)	i5 e15.5 i5	Isotope identifier Isotope density (atoms/barn-cm) Thermal scattering data identifier

Record 3 is repeated for each of the niso_m identifier/density pairs.

Records 2, 2a, and 3 are repeated for each of the lmat materials.

Table A.4. Listing of isotopes available in library to describe material compositions

Identifier	Edit Identifier	Fast neutron/gamma	Thermal neutron element/molecule	Note
1	508	Hydrogen	none	must be combined with thermal
2	505	Oxygen	Oxygen	Free gas kernel
3	505	Oxygen	H ₂ O molecule	used in conjunction with Hydrogen
4	509	Aluminum	Aluminum	
5	510	⁶ Lithium	⁶ Lithium	
6	511	⁷ Lithium	⁷ Lithium	
7	512	Carbon	Carbon	
8	513	Nitrogen	Nitrogen	
9	521	¹⁰ Boron	¹⁰ Boron	
10	522	¹¹ Boron	¹¹ Boron	
11	525	Sodium	Sodium	
12	528	Sulfur	Sulfur	
13	532	Calcium	Calcium	
14	533	Magnesium	Magnesium	
15	536	Deuterium	Deuterium	
16	541	Chlorine	Chlorine	
17	542	Potassium	Potassium	
18	543	Phosphorus	Phosphorus	
19	830	Carbon	CH ₂	must be combined with Hydrogen
20	507	Lead	Lead	
21	537	Fluorine	Fluorine	
22	508	Hydrogen	Hydrogen	Free gas kernel
23	523	Cadmium	Cadmium	
24	-	Hydrogen	none	Thermal sigma=1.0 (not used for edits)
25	-	Hydrogen	none	Thermal sigs=1.0 (not used for edits)
26	554	Bismuth	Bismuth	
27	-	Silicon	Silicon	

Table A.5. List of materials available in Materials File

Fairchild_tissue	Fairchild_tissue+5ppm	Fairchild_tissue+10ppm
Fairchild_tissue+20ppm	Duck_cranium	brain_MIT
brain_MIT+5ppm	brain_MIT+10ppm	brain_MIT+20ppm
void_edit	buffer	Li6_polymer_shield
lucite	poly_LIOH_enr	pure_poly
Rx_201	Moran_wgtd_doghead	H2O
tissue	plywood	A150
boroflex	Rx_215_nat	styrofoam
Rx_215_enr	Li6	lexan
polybor	Pb-B-Wax	Cadmium
Li6_epoxy	lead	PMMA
concrete	search_light	lung
true_void	brain_ICRU_46_adult	skin_ICRU_46_adult
skeleton-cranium_ICRU_46	skin_ICRU_46_adult_5ppm	skin_ICRU_46_adult_10ppm
skin_ICRU_46_adult_20ppm	skin_ICRU_46_adult_30ppm	skin_ICRU_46_adult_40ppm
brain_ICRU_46_adult_5ppm	brain_ICRU_46_adult_10ppm	brain_ICRU_46_adult_20ppm
skin_ICRU_46_adult_50ppm	glassbead	eye_lens_ICRU46_adult

A.2.17. Line 17 - Cross section file

The cross section file contains all the physics data required to accurately determine interaction rates and secondary energy and angular distributions, as well as probabilities for generation of secondary particles. The Evaluated Nuclear Data Files (ENDF/B), and other sources, coordinated through the National Nuclear Data Center, are used.

A.2.18. Line 18 - Edit isotopes

In addition to the normal region and edit mesh tallies, there are four special tallies on the edit mesh to provide the dose distributions required for treatment planning, and two special reaction rate tallies that can be tailored to the user's purposes. The four dose tallies can be applied to any isotope in the fast neutron library, but usually are applied to:

- 521: The ^{10}B capture dose
- 508: Hydrogen recoil dose
- 513: ^{14}N capture dose
- 512: Carbon dose
- 505: Oxygen dose ,

where the last two are combined into the Other Dose edit. Note that no matter what isotopes are invoked on the edit isotopes line, the edits in the seraMC output are always labeled boron dose, hydrogen dose, nitrogen dose, and other dose. The two reaction rate tallies may utilize any isotope in the fast neutron library, and are designated Reaction Rate 1 and Reaction Rate 2 in the output. These last two are only included for point and line edits, not dose-volume histograms or contour edits.

The fast neutron IDs and densities must be supplied on this input line. The nuclide KERMA data used by seraMC are retrieved from the cross section libraries. The densities are assumed constant for all edit mesh voxels. This is done to produce coherent contour profiles for these doses.

Concentrations can be modified in the seraMC edit directives or the seraDose widget for specific tissue of interest.

A.2.19. Line 19 - Run script for seraMC

The seraMC run script launches seraMC and performs some file management activities. It accepts one argument, the name of the seraMC input file. It assumes that the file is named name.input, and it creates an output file named name.out. Invoking this from the seraCalc widget runs seraMC in the background. This is included only for the seraCalc interface, as the run script is invoked either from seraCalc or the command line.

There are three environmental variables that must be set when using seraMC. These can be initiated in a login script or set with a command such as setenv:

```
setenv SERAPATH $SERA_HOME  
setenv SERA_BIN $SERAPATH/Target/bin ,
```

where \$SERA_HOME is the directory where the software resides.

The input to seraMC is provided in a file with the suffix .input (e.g., case1.input). A seraMC run can be made by using the run_seraMC script as:

```
run_seraMC case1
```

In this example, the output from seraMC is written to the file case1.out, which may be processed by ancillary software to provide desired edits. The region fluxes and specified edits are written to the .out file and other files as required by the edits. The detailed flux and dose distributions are stored in the edit-voxel arrays that are written to the file case1.rst, which is specified in the input. If more than one transport mode were selected (e.g., NFG), additional files are written; in this case, they would be case1.rsta and case1.rstb. The file case1.rstb contains all information from the previous transport modes, and after successful completion of a run, files case1.rst and case1.rsta will be discarded.

The raster files view.1, view.2, and view.3 are created automatically. These files provide three orthogonal perpendicular views centered at the target point. These files may be viewed in seraDose; they are very useful for geometry checking.

During execution, seraMC writes a file called sera.mon, which presents a current run status, the number of particles processed and lost, and the reference flux location and value of thermal flux. In version 1B0, the flux and location are written only at the end of the transport calculation.

Finally, seraMC writes a temporary file called RUN.INPUT. This is actually an echo of the edit directives and additional generated directives, and it is removed after seraMC completes. The purpose of this file is to prevent two seraMC runs from being made simultaneously in the same working directory. If a seraMC case is running and the user tries to execute another seraMC run in the same directory, the run_seraMC script detects the presence of RUN.INPUT and blocks execution of seraMC, which could destroy results of the already-running case. Occasionally, seraMC will terminate abnormally and leave RUN.INPUT in the directory even though seraMC is not running. In this case, it is necessary to manually remove RUN.INPUT before another run can be initiated.

A.2.20. Line 20 - Proton range file (U or P mode only)

When the ultrafast mode calculations are invoked, data for the proton transport are needed by seraMC. These data consist of tables of proton ranges and detour factors in various materials at energies from 100MeV to essentially zero. These allow the proton dose to be computed at all points within the edit mesh. Data are included for all materials likely to be encountered in normal use of seraMC. The default file is \$SERAMC/lib/range.seraMC, and it is recommended that this file always be used.

A.2.21. Line 21 - Ultrafast directory file (U or P mode only)

The ultrafast directory file provides information needed for reading and processing the ultrafast cross section file. This consists primarily of the isotopes for which information is provided, and the locations in the cross section file for each one. Appended to this information is composition information for the same materials contained in the seraMC materials file. These provide material mixing data for the ultrafast computation. The default file is \$SERAMC/lib/dirult, and it is recommended that this file always be used.

A.2.22. Line 22 - Ultrafast cross section file (U or P mode only)

The ultrafast cross section file contains the cross section data for 11 isotopes, for the energy range 16.9MeV to 100MeV. This includes secondary particle energy-angular distributions, and gamma production data. These data are provided as linearly interpolated pointwise information in energy and angle. The isotopes available at present are: H, Be, C, N, O, Al, Si, Ca, Fe, W, and ²³⁸U. The default file is \$SERAMC/lib/ultralib, and it is recommended that this file always be used.

A.3. Coordinate Transformations

This section follows the required input section (Block 1) and is located before the automatic beam locator section. The transformation input is not required if the correspondence between image geometry and model geometry is the standard for the sera system.

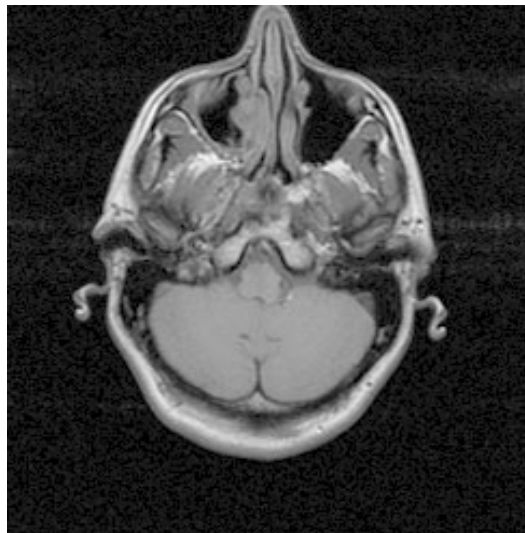


Figure A.4. Transverse MRI image of a healthy (volunteer) individual.

In the seraMC model, the x-axis vector direction is posterior to anterior (back of head to front), the y-axis vector direction is right to left, and the z-axis vector direction is inferior to superior

(feet to head). Thus, for transverse (axial) images, the image columns are assigned to the y-axis and the rows are assigned to the x-axis. Since the reconstruction software puts the material assignments in row, column, slice order, the rows and columns are switched by the `trans_bs` directive. The default model is based on transverse images and corresponds to:

```
trans_bs
  0.0 0.0 0.0
  1.0 1.0 1.0
  0.0 0.0 0.0
  1.0
  2 1 3
```

The 3x3 matrix is a limited transformation matrix, with the rows denoting three vectors, a, b, and c. The 1.0 is a scaling factor, and the 2 1 3 is a re-index vector which assigns x to columns and y to rows. For this default case, the `trans_bs` section is not required.

The relation between the initial vector x and the resultant vector x' is given by the following system of equations:

$$\begin{aligned} x'(1) &= a(1) + b(1) * \{x(\text{in_x}) + c(1)\} \\ x'(2) &= a(2) + b(2) * \{x(\text{in_y}) + c(2)\} \\ x'(3) &= a(3) + b(3) * \{x(\text{in_z}) + c(3)\}, \end{aligned}$$

where `in_x`, `in_y` and `in_z` are the re-index vector elements. For the default case, `in_x` = 2, `in_y` = 1, and `in_z` = 3, which results in the x- and y-axes being switched. Obviously, much more complex transformations are possible, but unlikely.

One could, however, input the matrix, scaling factor and re-index vector as desired to choose another correspondence between image and model space. For example, dogs usually have the top of the head at the top of the image, so the coordinate system transform could be accomplished by:

```
trans_bs
  3 2 1
```

Here, the code recognizes that the matrix and scale factor are not present and uses the default values. Since the coordinates in the patient geometry files are in cm, the scale factor and transform matrix would depend upon a ratio of the field of view (FOV) in image space to the desired FOV rather than on the actual value of the desired FOV.

A.4. Automatic Beam Locator

Block 3 of the `seraMC` input consists of the key word `iop` and necessary input required to allow a search for orienting the beam in a specified manner. The geometry must be such that the beam geometry does not coincide with the patient geometry, and the patient geometry is always surrounded by the CG buffer region for all search orientations, or the search may fail. The geometry for the source positioning search is shown in Figure A.5, and applies for all beam positioning search modes. For more information on the beam location parameters, see Section A.2.4.

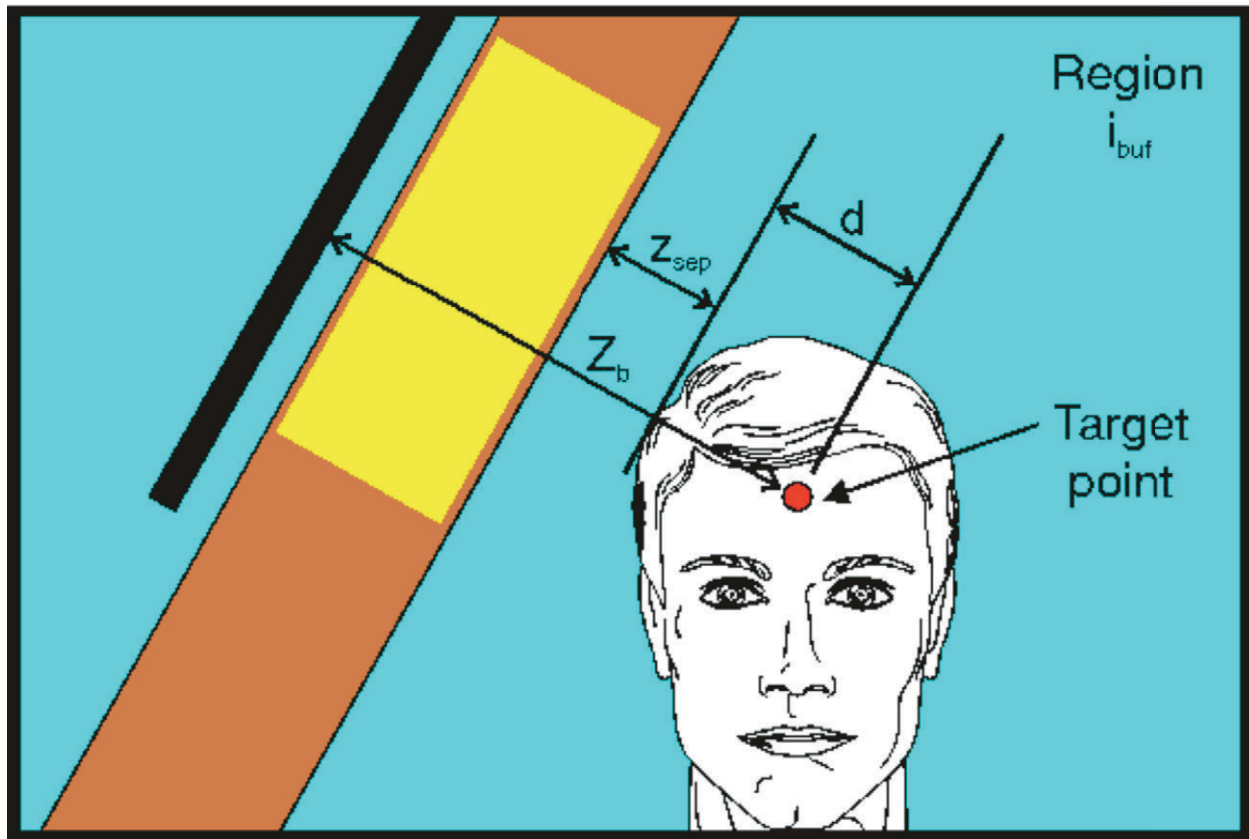


Figure A.5. Beam positioning schematic, showing the various variables employed for the positioning search.

A.4.1. Positioning when $iop = 1$ (default)

The beam location is determined by the source geometry, the target point (X_p, Y_p, Z_p), the distance from the target to the source center (Z_b), and the two rotation angles (ϕ and θ) specified in Block 1. The iop block is not required for the default case.

A.4.2. Positioning when $iop = 2$

A sample input for a case where $iop = 2$ is shown below.

```
iop
2 aperture buffer
0.5
-3.0 2.0 5.0
```

The first input datum sets $iop = 2$, and the next two items are the names of regions encountered between the source plane and the patient, **in the order a source particle would cross them**. Aperture is the name of the first region encountered in the source geometry; it is shown as the yellow region in Figure A.5. Buffer is the name of the region outside the patient; it is shown as the turquoise region in Figure A.5. The region indices could be used instead of region names, but it is much more clear if the region names are used. Note that only two regions are allowed for the specification of the iop input. This will affect the allowed geometry of the source specification (see Appendix C).

The next item is zsep (0.5), which is the desired distance from the skin outer surface to the aperture region along the beamline. The user should ensure that the zsep value specified is sufficiently large to prevent any overlap of the patient geometry and the source geometry, as this overlap will result in lost particles and possible termination of the seraMC run. The last three data items are the x,y,z coordinates of a point which the beamline must intersect between the target point and the source center.

Note that if a plus sign is added to the iop keyword (iop+) then a search will be conducted on zsep to determine the minimum distance required to avoid overlap of the source and patient geometries. This also applies to the other iop search options.

A.4.3. Positioning when iop = 3

A sample input for a case where iop = 3 is shown below.

```
iop
3 aperture buffer
0.5
```

In this instance, iop = 3, and the input is the same as for iop = 2, except that the positioning point is not included. When iop = 3, seraMC automatically finds the beamline such that the distance d from the target point to the skin surface is a minimum. Often, this results in a near optimum beam for single-field applications. The user should ensure that the zsep parameter specified is sufficiently large to prevent any overlap of the patient geometry and the source geometry, as this overlap will result in lost particles and possible termination of the seraMC run.

A.4.4. Positioning when iop = 4

A sample input for a case where iop = 4 is shown below.

```
iop
4 aperture buffer
0.5
```

Now iop = 4 and the input is the same as for iop = 3. In this case, however, seraMC does not do a search for beam angle. The beam angles entered in Block 1 are used, and seraMC merely adjusts Zb to obtain the desired separation distance, zsep. The user should ensure that the zsep parameter specified is sufficiently large to prevent any overlap of the patient geometry and the source geometry, as this overlap will result in lost particles and possible termination of the seraMC run. This is a very commonly used option.

A.4.5. Positioning when iop = 5

A sample input for a case where iop = 5 is shown below.

```
iop
5 aperture buffer
0.5
```

Now iop = 5 and the input is the same as for iop = 3 again. In this instance, however, the seraMC search for the minimum distance d is performed with the azimuthal angle theta fixed at the value specified in the Block 1 input; the polar angle phi is allowed to vary.

A.5. Edit Directives

Following the transport computations or during a restart run, seraMC may be run in D mode, in which case the code looks for edit directives to provide desired outputs. Some edits, such as the region edits and the peak voxel edits, are obtained automatically. To obtain additional edits, the user must insert the edit directive block of the input, headed by the key word `edit_dir`. SeraMC then reads all edit directives, pausing to perform the edit whenever an active edit directive is encountered. There are four types of edit directives. These are:

- plot headers
- directives that, if used, must be placed before active edit directives
- directives that set parameters and can be placed anywhere
- active edit directives.

Plot headers are not used by seraMC, but they are listed in the output so that seraPlot can post-process the seraMC output and use the headers as labels in plot files.

There are certain edit directives, which, if used, must appear before any active directive in order to be effective. This is the case because the first thing seraMC does when encountering the first active directive is to search for the reference edit voxel(s), where the default reference voxel is the voxel having peak thermal flux. Therefore, directives such as `refvol` and `refpoint`, which affect the reference volume definition, must appear before the first active edit directive.

Before performing an edit, the user usually needs to specify certain parameters that will be used in the edit. When these directives are encountered, parameters such as boron concentration, RBE, etc., are set and later used when the next active edit is encountered. These may be changed after the edit is performed, and will be used in a subsequent edit. Finally, there are active edit directives that cause the actual edits to be performed. A variety of active edit directives is available to the user, allowing nearly any desired dose or flux quantity to be generated.

A detailed listing and description of all the available edit directives is given in Appendix D.

It is highly recommended that edits be performed with the SeraPlan module for univel geometry cases. The edit directives should be reserved for CG geometry cases only, which cannot be edited in SeraPlan.

Appendix B. seraMC Combinatorial Geometry

B.1. Introduction

The Combinatorial Geometry (CG) method describes three-dimensional region configurations by considering Boolean combinations of simple bodies such as spheres, boxes and cylinders. The geometry description subdivides the problem space into unique regions. In all descriptions, there must be a region, completely enveloping all other regions (including the buffer in univel patient descriptions), that contains the material *ficitious*. When particles enter this region, they are tossed away (i.e., leak from the system). Each region is the result of combining one or more of the following geometric bodies:

1. RPP - rectangular parallelepiped aligned with axes
2. BOX - an RPP randomly oriented in space
3. SPH – sphere
4. RCC - right circular cylinder
5. REC - right elliptical cylinder
6. TRC - truncated right angle cone
7. ELL – ellipsoid
8. RAW/WED - right angle wedge
9. ARB - arbitrary polyhedron of 4, 5, or 6 sides
10. RVW - right vertical wedge
11. RGE - restricted general ellipsoid.

Body types 2-9 may be arbitrarily oriented with respect to the x, y, and z reference coordinate axes. Body type 1 must have sides parallel to the xy, xz, and yz planes. Body type 10 must have an axis parallel to the z-axis and top and bottom planes parallel to the xy plane. Body type 11 must have all three axes parallel to the three coordinate axes, but may be centered at an arbitrary point in space. It is important to remember that every spatial point in the geometry must be located in one, and only one, region. Failure to follow this will produce unpredictable results from the SeraMC calculations.

B.2. Body Input Specifications

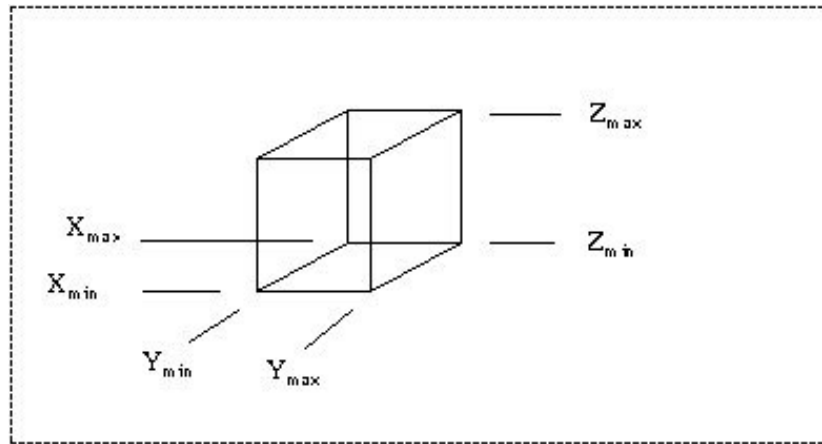
In general, a point is specified as a triplet of values (x,y,z) defining the point in space. A vector is defined by specifying its x, y, and z components with the vector length included in this description, i.e., the vector is normalized to the length of the body component. All angles are given in degrees, and coordinates and other dimensions in space are given in centimeters.

The general format for defining a CG body is:

```
DIR  N  data for body
```

where DIR is the body specification type in (2x,a5) format, N is a body number in (i5) format, and the data for the body are given in (6e10.3) format. Note that when more than six entries are needed for a body, the remaining entries are entered on additional lines, with (10x,6e10.3) format. The entries for each body are described in the order in which they must be entered on the CG body specifications.

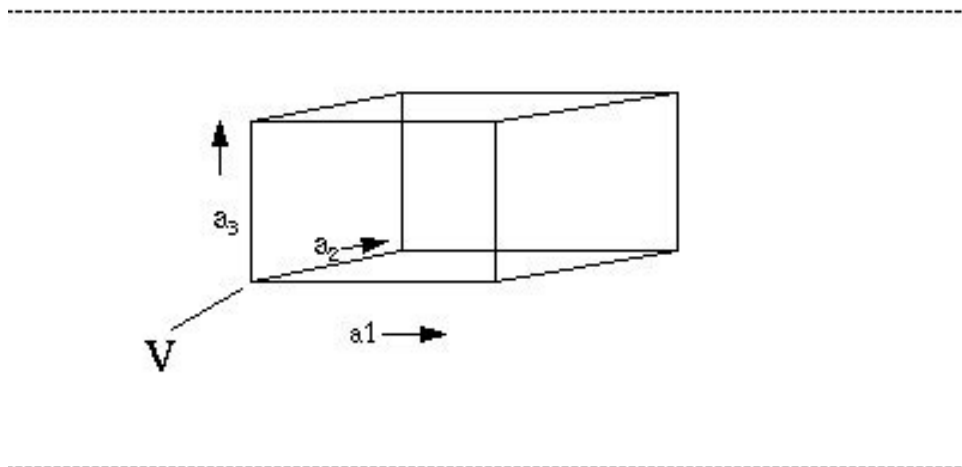
B.2.1. Rectangular Parallelepiped (RPP)



Rectangular Parallelepiped body (RPP)

Specify the minimum and maximum values of the x, y, and z coordinates which bound the parallelepiped, i.e., x_{min} , x_{max} , y_{min} , y_{max} , z_{min} , z_{max} .

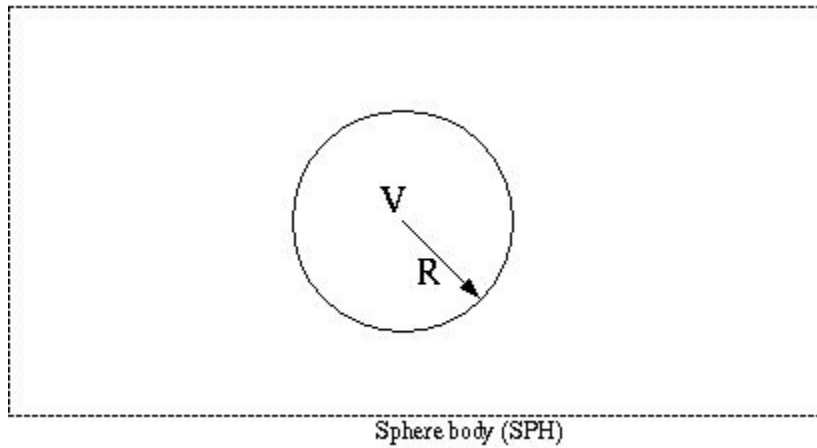
B.2.2. Box (BOX)



Box body (BOX)

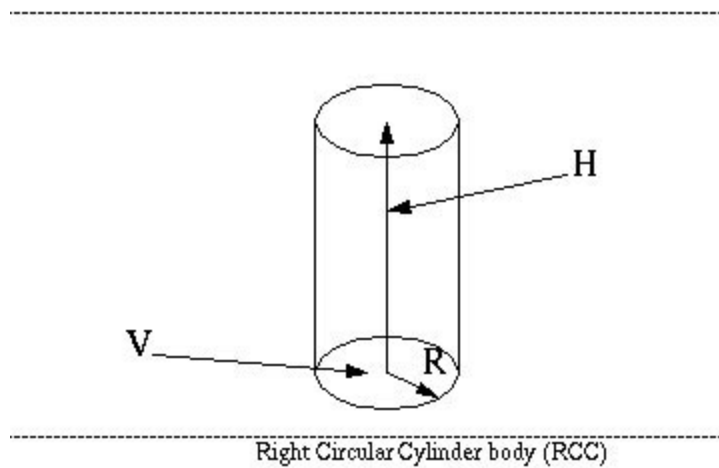
Specify the vertex V at one of the corners by its (x,y,z) coordinates. Specify a set of three mutually perpendicular vectors (a_1 , a_2 , and a_3), representing the height, width, and length of the box, respectively.

B.2.3. Sphere (SPH)



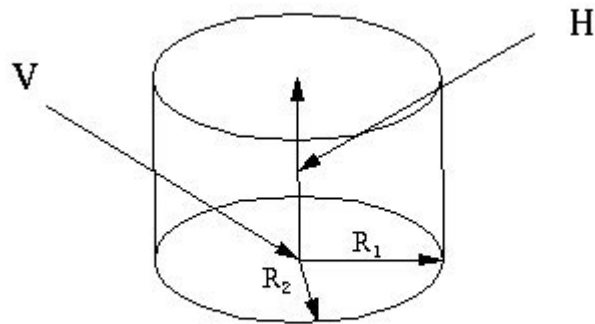
Specify the point V at the center of the sphere, and the scalar R denoting the radius.

B.2.4. Right Circular Cylinder (RCC)



Specify the point V at the center of one base, a height vector H , and a scalar R denoting the radius.

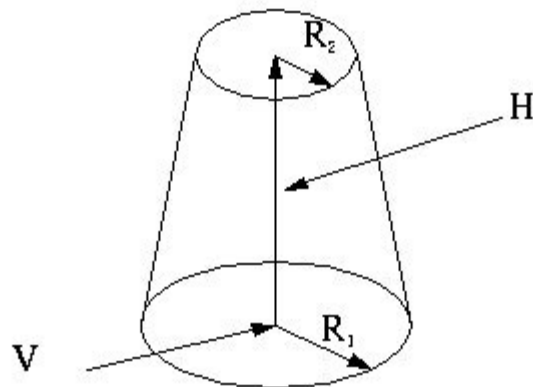
B.2.5. Right Elliptical Cylinder (REC)



Right Elliptical Cylinder body (REC)

Specify the point V at the center of the base ellipse, a height vector H , and two vectors R_1 and R_2 in the plane of the base, defining the major and minor axes.

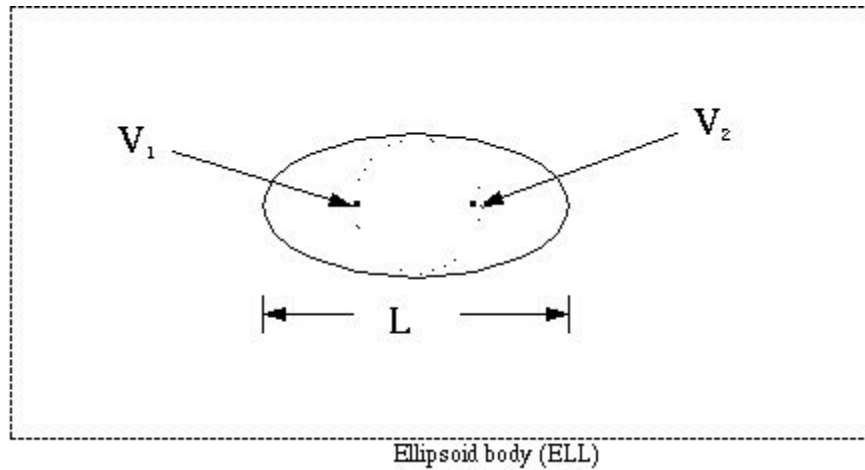
B.2.6. Truncated Right Angle Cone (TRC)



Truncated Right Angle Cone body (TRC)

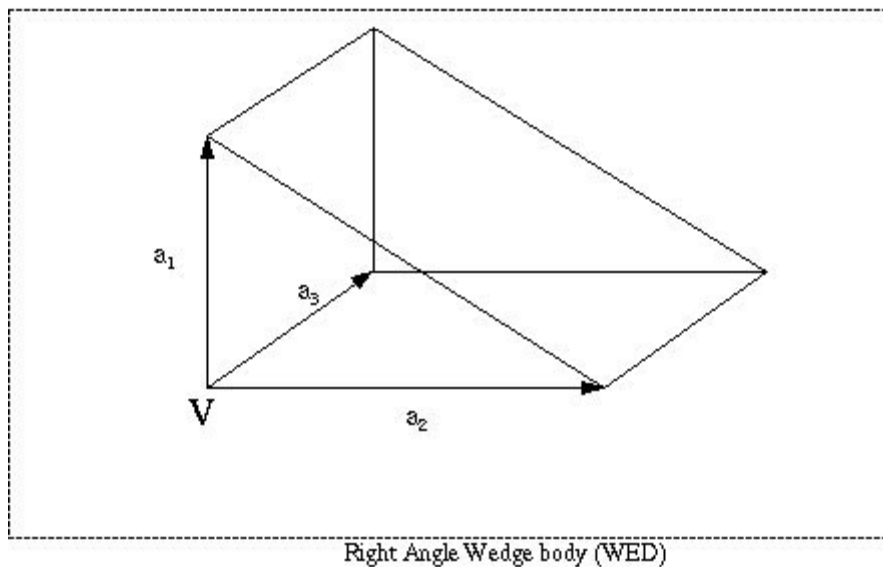
Specify a point V at the center of the lower base, the height vector H , and two scalars R_1 and R_2 denoting the radii of the lower and upper bases.

B.2.7. Ellipsoid (ELL)



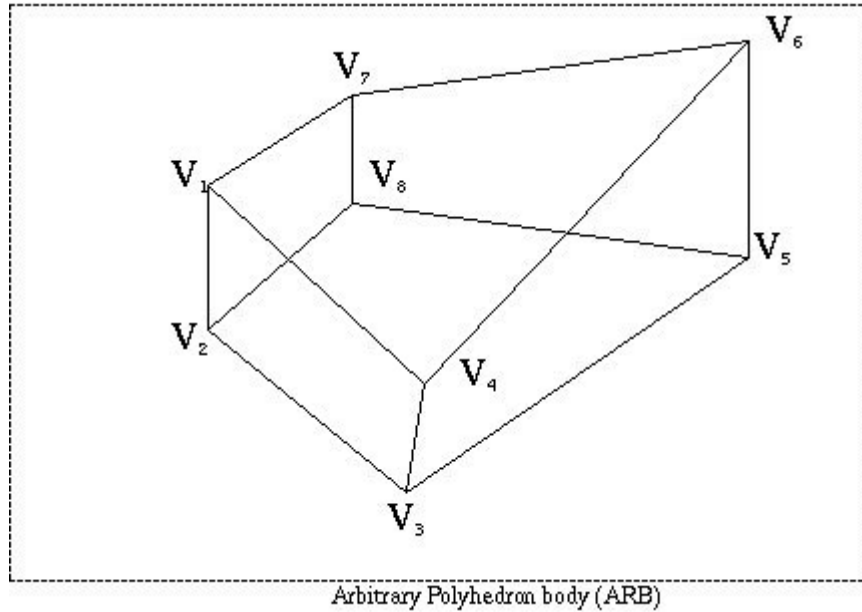
Specify two points V_1 and V_2 denoting the coordinates of the foci, and the scalar L denoting the length of the major axis. **Note: this is no longer in use. Use RGE instead.**

B.2.8. Right Angle Wedge (WED)



Specify the vertex V at one of the corners, and a set of three mutually perpendicular vectors a_i , with the vectors a_1 and a_2 describing the two legs of the right triangle of the wedge.

B.2.9. Arbitrary Polyhedron of 4, 5, or 6 sides (ARB)

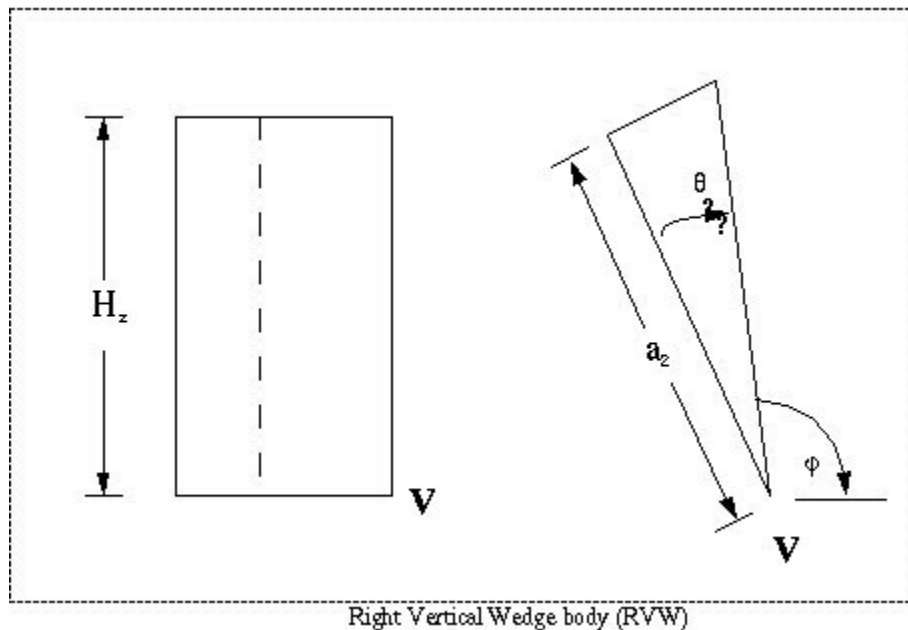


Assign an index (1-8 for a six-sided ARB) to each vertex. For each vertex, give the x, y, and z coordinates. Each of the faces is then described by a four-digit number giving the indices of the four vertex points in that face. For each face, these indices must be entered consistently in either clockwise or counterclockwise order.

An example of the ARB body, in this case, describing a simple unit cube, looks as follows:

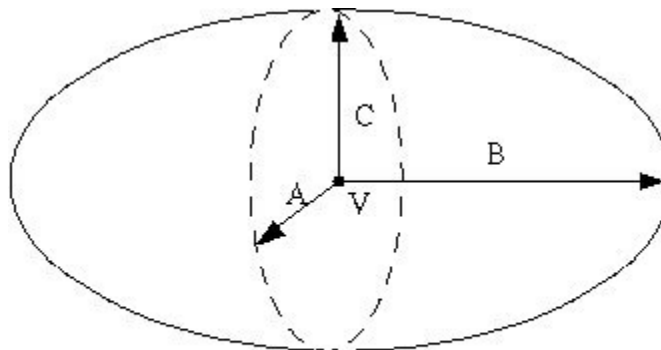
ARB	1	0.0	0.0	0.0	1.0	0.0	0.0
		1.0	1.0	0.0	0.0	1.0	0.0
		0.0	0.0	1.0	1.0	0.0	1.0
		1.0	1.0	1.0	0.0	1.0	1.0
		1234	1584	1562	5678	2673	4873

B.2.10. Right Vertical Wedge (RVW)



Specify the vertex V at the lower plane of the wedge. Specify a scalar defining the width of the side joining the vertex and the right-angled corner (a_2). Specify theta, the angle subtended by the wedge at the vertex. Specify phi, the angle between the x-axis and the wedge face opposite the right-angled corner, and H_z , the scalar height of the wedge. This body could also be obtained with the WED body, but the RVW is easier to specify and has been found to give fewer round-off problems.

B.2.11. Restricted General Ellipsoid (RGE)



Restricted General Ellipsoid body (RGE)

Specify the half-lengths of the three major elliptical axes A , B , and C , and the center point V . This will generate a general ellipsoid (not an ellipsoid of revolution, as for the ELL body) with axes parallel to the three coordinate axes, centered at V . The elliptical axes A , B , and C are parallel to the x -, y -, and z -axes, respectively. This body is particularly useful for modelling the so-called Snyder phantom, which is commonly used for beam measurements.

B.3. Combinatorial Geometry Descriptions

The basic technique for the description of the geometry consists of defining the location and shape of the various regions in terms of the intersections and unions of the geometric bodies. A special operator notation involving the symbols +, -, and OR is used to describe the intersections and unions. These symbols are used by the program to construct information relating material descriptions to the body definitions. If a body appears in a region description with a + operator, it means that the region being described is wholly contained in the body. If a body appears in a region description with a - operator, it means that the region being described is wholly outside the body. If the body appears with an OR operator, it means that the region being described includes all points in the body list. OR may be considered as a union operator. In some instances, a region may be described in terms of zones lumped together by a union of these zones. When OR operators are used there are always two or more of them, and they refer to all body numbers following them, either + or -, that is, all body numbers between ORs or until the end of the region input for that region are intersected together before ORs are performed. Figure B.1 visually demonstrates the results of the Boolean operators.

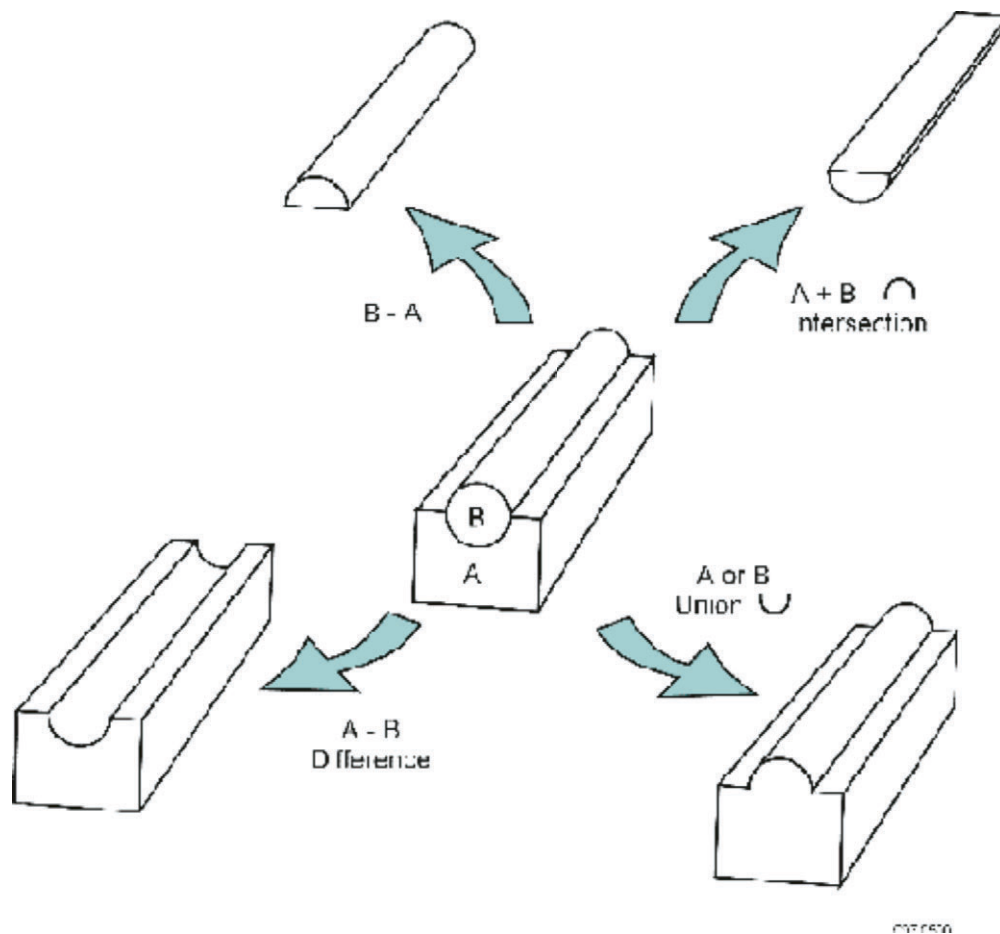


Figure B.1. Results of Boolean operations in CG geometry.

B.4. Examples of CG Descriptions

Techniques for describing a particular geometry are best illustrated by examples.

B.4.1. Example 1

This example is an object composed of a sphere and a cylinder, as shown in Figure B.2.

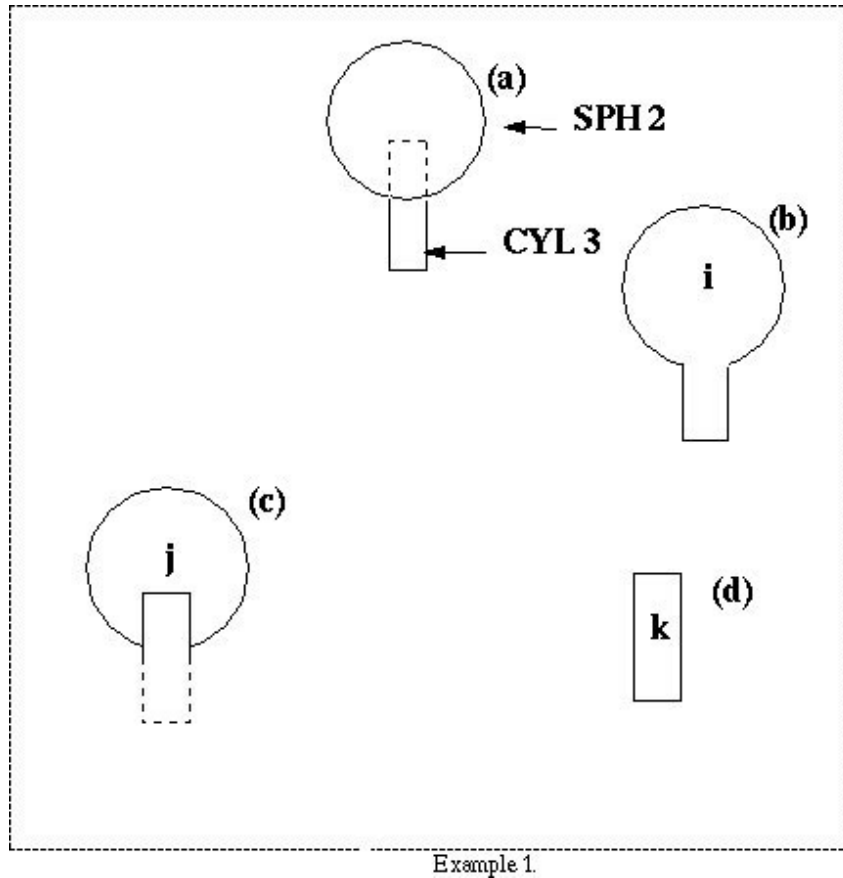


Figure B.2. Geometry for Example 1.

To describe the object, assume a spherical body penetrated by a cylindrical body (Figure B.2(a)). If the materials in the sphere and cylinder are the same, then they can be considered as one region, say region *i* (Figure B.2(b)). The description of region *i* would be:

$$i = \text{OR} + 2\text{OR} + 3$$

This means that a point is in region *i* if it is either inside body 2 or inside body 3. If different materials are used in the sphere and cylinder, then the sphere with a cylindrical hole in it would be given a different region assignment (region *j* in Figure B.2(c)) from that of the cylinder (region *k* in Figure 2d). The description of region *j* would be:

$$j = +2 - 3$$

This means that points in region *j* are all those points inside body 2 which are not inside body 3. The description of region *k* (Figure B.2(d)) is simply all points inside body 3, or:

$$k = +3$$

Combinations of more than two bodies and similar region descriptions could contain a long string of +, -, and OR operators.

B.4.2. Example 2

As a more complicated example of the use of the OR operator, consider the system shown in Figure B.3, consisting of the shaded region A and the unshaded region B.

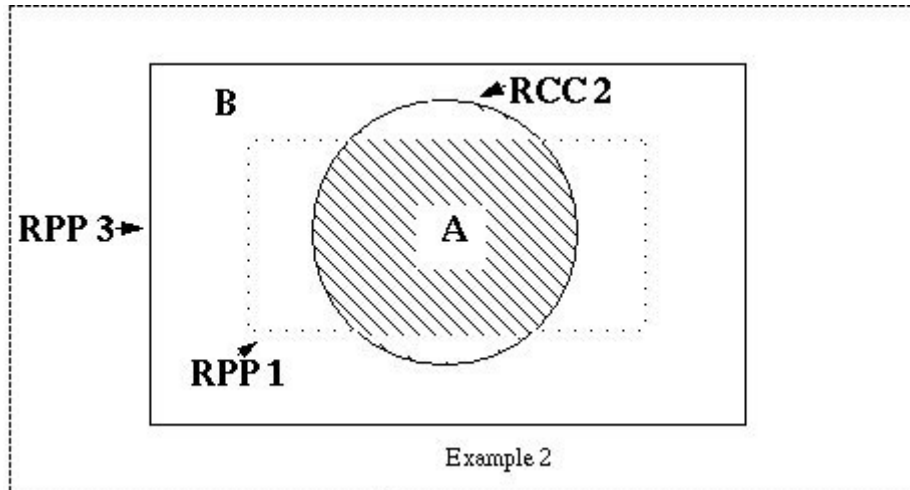


Figure B.3. Geometry for Example 2.

These regions can be described by the two RPP's, bodies 1 and 3, and the RCC, body 2. The region description would be:

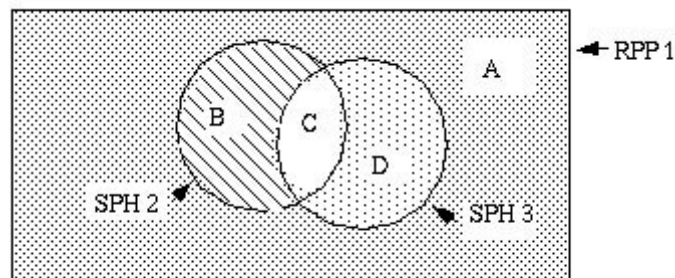
$$A = +1 +2$$

$$B = \text{OR } +3 -1 \text{OR } +3 -2$$

Notice that the OR operator refers to all following body numbers until the next OR operator is reached, so that the verbal description of B would be "all points inside body 3 and outside body 1 plus all points inside body 3 and outside body 2."

B.4.3. Example 3

This example consists of two overlapping spheres within a rectangular parallelepiped, as shown below in Figure B.4.



Example 3

Figure B.4. Geometry for Example 3.

Five regions may be described, as shown below.

Region A - composed of all points interior to body 1 and exterior to bodies 2 and 3.

$$A = +1 -2 -3$$

Region B - composed of all points interior to body 2 and exterior to body 3.

$$B = +2 -3$$

Region C - composed of all points that are common to bodies 2 and 3.

$$C = +2 +3$$

Region D - composed of all points interior to body 3 and exterior to body 2.

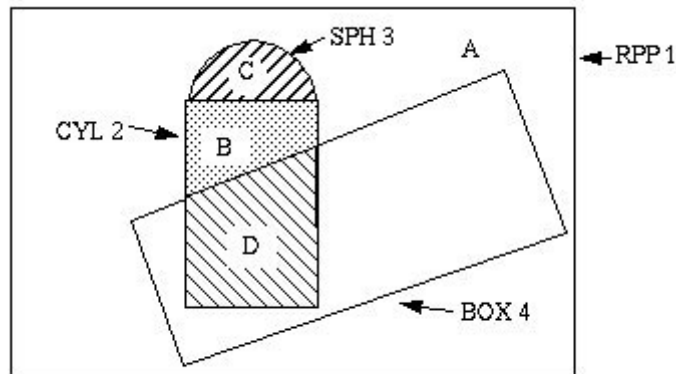
$$D = +3 -2$$

and Region E - composed of all points interior to bodies 2 and 3.

$$E = OR +2OR +3$$

B.4.4. Example 4

Example 4 consists of a cylinder divided into two regions by a box and with a sphere at one end, as shown below in Figure B.5.



Example 4

Figure B.5. Geometry for Example 4.

Region A - all points interior to the circumscribed parallelepiped and exterior to the cylinder and sphere. Note that region A includes all of the space contained inside the box except that portion inside the cylinder and sphere.

$$A = +1 -2 -3$$

Region B - all points interior to the cylinder and exterior to the box.

$$B = +2 -4$$

Region C - all points interior to the sphere and external to the cylinder.

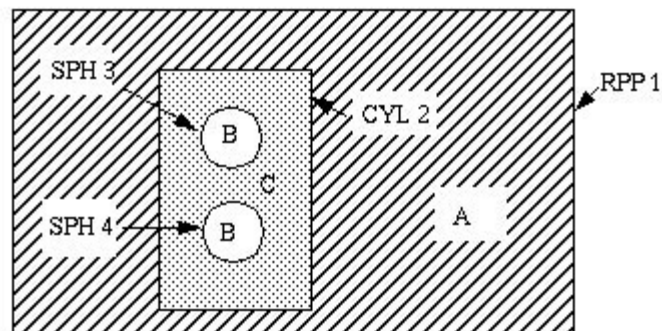
$$C = +3 -2$$

Region D - all points interior to the cylinder and also inside the box.

$$D = +2 +4$$

B.4.5. Example 5

Example 5 consists of a cylinder containing two spheres, enclosed inside a parallelepiped, as shown in Figure B.6.



Example 5

Figure B.6. Geometry for Example 5.

Region A - all points interior to the parallelepiped but exterior to the cylinder.

$$A = +1 -2$$

Region B - all points interior to the two spheres.

$$B = OR +3OR +4$$

Region C - all points interior to the cylinder but exterior to the spheres.

$$C = +2 -3 -4$$

B.5. Format of Combinatorial Geometry File

The format of the CG file is given below. It consists of a title record, several records defining the bodies, and several more records defining the Boolean operations to describe physical regions. Table B.1 is a listing of the record contents.

Table B.1. Format of combinatorial geometry file

Record	Format	Description of entries
1	(20x,10a6)	Title record - enter any descriptive information desired in columns 21-80.
2	(2x,a3,i5,6e10.3, (10x,6e10.3))	Body type directive (three characters), body number, data for particular body type specified.
3	(2x,a3)	Enter END in columns 3-5 to denote end of body descriptions.
4	(2x,a3,i5,9(a2,i5))	Region name (three characters), region number, and the Boolean directives to describe the region in terms of the bodies defined above. Body numbers from other files with geometry information, i.e., source file, may also be used in these operations.
5	(2x,a3)	Enter END in columns 3-5 to denote the end of the CG file.

Appendix C. seraMC Source Description

C.1. Introduction

The source file is generally constructed for a specific neutron and coincident gamma source. Determination of the source characteristics is usually the result of a grueling campaign conducted by several experts. The source file, once constructed and validated, is rarely changed thereafter. It contains the spectral and angular distributions of the incident neutron and gamma currents from a source plane that represents the original source, tailored by filters, beam collimators, etc. The source file also contains information about source location, extent, and magnitude, and possibly geometric information about beam delimiters, collimators, shielding, and other structure that is rigid with respect to the source.

C.2. Formats

The source file is specified in fixed format, with allowance for imbedded comment lines. Comment lines begin with a # character in the first column. Note that univel geometry is not allowed in the source region, but only the combinatorial geometry described in Appendix B is permitted. A sample source file in Table C.1 is provided for guidance.

C.3. Variable Description

Most of the variables in the source file are well described in Table C.1. However, some require further elaboration. These are described more fully in the sections below.

C.3.1. nsorcs

To accommodate a variety of radial source distributions, seraMC allows the user to split the source plane into multiple radial zones. This allows the user to specify a radial intensity shape, or to vary the energy or angular distributions with radius, or any combination of the above. This provides great flexibility in the source definition. The variable nsorcs defines the number of radial regions into which the source is subdivided. In the case of a disk source, the regions are concentric annuli. For a square source, the regions are still concentric, but are now square annuli.

C.3.2. nmnu

The angular distribution of the neutron and gamma currents employs a set of equiprobable angular bins. This provides sufficient flexibility to describe any anisotropic angular distribution with relatively few bins. The variable nmnu defines the number of equiprobable angular bins.

C.3.3. xtnsn and xtng

These are the total neutron and gamma currents, respectively, for each of the nsorcs concentric source regions. These currents are integrated over all energies and angles; they are used to normalize the unnormalized energy distributions given on Records 7a and 7b. These values also form the basis for the normalization factors for the edits. The dose normalization factors are constructed by multiplying the nsorcs source region normalization factors by the respective region areas, and summing over all source regions. Note that the units of the normalization factors are assumed to be particles/cm²/second, and any deviation from this will require adjustments on the part of the user. This is true for both the edits produced by seraMC and seraPlan.

Table C.1. Sample source file.

Record	Format	Name	Description of entries
1	a80	sname	Descriptive name for file
2	3e12.5	Xc, Yc, Zc	Center point of the planar source (cm)
	3i12	is_type	0 for disk source, 1 for rectangular source
		intrp	0 for 1/E energy interpolation, 1 for linear energy
		nr	0 for flat radial shape, 1 for buckling-determined radial
	f10.4	br	Buckling value to use for radial shape computation (only if nr = 1; nsorcs = 1 required)
3	5i12	nsorcs	Total number of concentric sources
		nngp	Number of energy bins for neutron source
		ngmgp	Number of energy bins for gamma source
		nspec	Number of energy spectra (each source is assigned a spectrum)
		nmu	Number of angular current bins
4	i12, 4e12.5	ivspec(is)	Spectrum index for each source (is = 1, nsorcs)
		xtnsn(is)	Average neutron intensity (current) for each source
		xtnsg(is)	Average gamma intensity for each source
		dim(is)	Outer radius (disk) or x-dimension (rectangle) (cm) of each source
		dim2(is)	Outer y-dimension (cm) of each rectangular source
5	6e12.5	sengy(ig)	Neutron energy breakpoints (ev) for each group (ig = 1, nngp + 1)
6	6e12.5	sengy(ig)	Gamma energy breakpoints (eV) for each group (ig = nngp + 2, nngp + ngmgp + 2)
7a	6e12.5	fspec(ig, ispec)	Un-normalized neutron current intensities for each group and each spectrum (ig = 1, nngp)
7b	6e12.5	fspec(ig, ispec)	Un-normalized gamma current intensities for each group and spectrum (ig = nngp + 1, nngp + ngmgp)
7c	6e12.5	xmu(i, ig, ispec)	Lower cosine cutoffs for each group and each spectrum (i = 1, nmu)
Records 7a, 7b, and 7c are repeated for each spectrum (ispec = 1, nspec)			
8	i5	lreg	Number of regions used to define the beam delimiter for this source
	(/, (1x, a40, a20))	mat(ir), reg(ir)	Material name and CG body name for each region (ir = 1, lreg)
9	See Sect. B.5		Combinatorial geometry description for the bodies named above. All records and formats for the CG file description apply here.

C.3.4. dim

This gives the outer dimension of each concentric source region. The values give the radius of a disk, or the half-length of the side of a square.

C.3.5. fspec

These are the unnormalized group currents for the neutron and gamma source components.

C.3.6. lreg, mat, and reg

These entries define the regions used to describe the beam delimiter and collimator for the source. The variable reg provides the region names for each of the lreg regions used in the CG description of the delimiter, which are carried through the entire seraMC calculation. The variable mat gives the seraMC material name used for each region. These materials are chosen from the same set used in seraModel to define the bodies in the patient model. More details on these are found in Section A.2.16.

Appendix D. seraMC Edit Directives

D.1. Introduction

The seraMC edit directives provide user control over the quantities calculated from the transport calculation results. Various integral quantities for dose and flux may be computed and printed, without any influence on the transport calculation. This capability is a result of the implicit assumption in seraMC that the boron level in BNCT is not sufficient to cause significant depressions in the local thermal flux. Thus, a single approximate value for the boron concentration is included in the materials for the transport calculation, and the tally results are calculated per ppm ^{10}B . This characterization allows the edit doses to be computed, by post-processing, for nearly any boron concentration observed thus far in clinical application.

There are four different levels of edit directives in the seraMC input. The passive directives have no active role in the seraMC edits, but merely provide a means for the user to provide information to be used by seraPlot to label plots. The reference dose parameters influence the determination of the reference dose values and thus they must precede any active edit directives. The general parameters provide information that is used by individual active edits. These directives may make multiple appearances, with subsequent values overriding earlier specifications. These directives may appear anywhere in the edit input block. The fourth level is the active edits themselves, which perform the actual edit calculations. Most of the active edits print directly to the standard output file, but some create additional files for their results.

The edit input block must appear after all other data in the seraMC input file. The beginning of the edit block is denoted by the string `edit_dir`, which must start in column 1. All subsequent edit directives must start in column 2, with the values beginning after column 10. Edit directives are specified in the following general keyword-value format:

keyword value₁ value₂ value_n

where the values are space-delimited. Note that the keyword is case-sensitive, and it must be specified exactly as shown in the subsection headings below. Comment lines are allowed and must begin with a # character in column 1, with the comment text starting after column 3.

The edit capability has been shifted to the seraPlan edit module, which is much simpler and more user friendly. The edit capability in seraMC is no longer being supported for univel geometry applications. This section of the manual is retained for historical and reference purposes only.

D.2. Passive Directives

D.2.1. patient

The patient directive allows the user to specify an identifying name for the edit run, which is usually the patient's name or initials. This directive begins with a # character in column 1, and the keyword begins in column 2.

D.2.2. date

The date directive allows the user to specify an identifying date for the edit run, which is usually the treatment date. The format is the same as that for the patient directive.

D.2.3. beam

The beam directive allows the user to identify the particular beam source used for the calculation. The format is the same as that for the patient directive.

D.3. Reference Dose Parameters

The first three directives are used to specify the reference volume (refvol), the reference point (refpoint), or the reference depth (refdepth).

Note that only one of refvol/refpoint/refdepth may be used in any one problem, and the chosen option may only be used once in any one problem.

D.3.1. refvol

The refvol directive specifies the volume over which the reference dose is defined and also provides a flag defining the search method used to find the reference voxel(s). The reference dose will be defined as the volume-averaged dose from nvox voxels, where nvox is the ratio of refvol to the voxel volume. If refvol is not a multiple of the voxel volume, then the calculation of nvox serves to truncate the value of refvol specified. The default value for refvol is 1.0.

The flag, in its default value of 0, forces seraMC to define the reference voxel as the voxel with maximum thermal neutron flux. A flag value of 1 defines the reference voxel as the voxel with the maximum total boron- and RBE-weighted dose. The boron concentration value is defined with the ref_b10 directive and the RBE values are defined with the ref_rbe directive.

Example: refvol 15.0 1
Default: refvol 1.0 0

D.3.2. refpoint

The refpoint directive specifies the x, y, and z coordinates, in centimeters, of a spatial point in patient geometry where the reference dose value is to be defined. The reference dose will then be the dose in the voxel that contains the defined point. The default value for refpoint is the center of the reference voxel.

Example: refpoint 2.0 1.3 4.5

D.3.3. refdepth

The refdepth directive specifies a depth, in centimeters, along the beam centerline from the skin surface to the reference point. The reference voxel will then be the voxel containing this point.

Example: refdepth 2.5

D.3.4. ref_reg and more_ref

These directives specify a list of acceptance regions for determination of the reference dose. Only regions on the defined list will be considered for inclusion in the reference voxel search procedure. The ref_reg directive provides the beginning of the list, and it may only be used once, but multiple more_ref directives, which provide continuation of the region list, may be used. For both directives, regions may be specified by either their region names from the geometry files or by their region indices.

Examples: ref_reg brain
ref_reg 7 8
ref_reg brain left-eye right-eye thalamus
more_ref optical-chiasm

The default is to use all regions.

These directives are active only with the refvol directive.

D.3.5. ref_b10

This directive provides the boron (^{10}B) concentration to use for the reference dose determination when the flag on the refvol directive is 1. The boron concentration is given in parts per million (ppm) by weight.

Example: ref_b10 15.1

This directive is active only with the refvol directive.

D.3.6. ref_rbe

This directive provides the set of RBE values to use for the reference dose determination when the flag on the refvol directive is 1. There are six RBE values in use for conventional energies (below 16.9MeV), and eight for ultrafast applications. The eight values modify, respectively, the boron, gamma, nitrogen, hydrogen, other, gamma production, proton recoil ($E > 16.9\text{MeV}$), and ultrafast dose components. Default values for all are 1.0.

Example: ref_rbe 1.3 1.0 3.2 3.2

This directive is active only with the refvol directive.

D.4. General Parameters

All of the directives in this section apply until overridden by a subsequent occurrence of the directive.

D.4.1. mw_min

This provides a multiplication factor for the output edits, and is meant to allow the user to specify the total energy delivered by the reactor in MW-minutes.

Example: mw_min 100.0
Default: mw_min 1.0

It is advised that mw_min be left at its default value to avoid confusion, and that the scaling factor be applied in seraPlot.

D.4.2. mw

This provides a multiplication factor for the output edits, and it is meant to allow the user to specify a reactor power in MW.

Example: mw 10.0
Default: mw 1.0

It is advised that mw be left at its default value to avoid confusion and the scaling factor be applied in seraPlot.

D.4.3. b10_blood

This directive sets the blood boron concentration in parts per million (ppm) by weight for the set of edits following it.

Example: b10_blood 15.1
Default: b10_blood 1.0

This directive is active until superseded by another b10_blood directive and it applies to all edits that follow it.

D.4.4. b10_ratio

This directive sets the ratio of the tissue boron concentration to the blood boron concentration for specified regions.

Examples: b10_ratio 1 1.0 2 1.5 4 3.5
 b10_ratio brain 1.0 tumor 3.5

Either region names or region indices may be used to specify the regions. The default is 1.0 for all regions.

D.4.5. delta

The delta directive defines the distance between edit points for line edits. For the dose-volume integrations, delta provides a starting spacing between points, which is halved until convergence on the volume is achieved.

Example: delta 0.25
Default: delta 1.0

This directive is active only with the line, box, and DVbs edit directives.

D.4.6. eps

This directive specifies the volume convergence criterion for the dose-volume integrations. An approximation to the size of the volume elements for the integrations is delta^3 .

Example: eps 0.05
Default: eps 0.01

It is often much faster to set delta relatively low, say 0.2 or less, and set eps to 1.0.

This directive is active only with the box and DVbs edit directives and it should only be specified once, before any dose-volume edits are performed.

D.4.7. **in_reg** and **more_reg**

These directives very closely resemble the `ref_reg/more_ref` directives. The `in_reg` and `more_reg` directives specify a restricted set of regions to be used for a subsequent edit. The regions may be specified either by their names or indices. The default is to use all regions.

These directives may be used with the `point`, `line`, `contour`, `box`, and `DVbs` directives and are active until another `in_reg/more_reg` directive is encountered.

D.4.8. **nbin_DV**

This directive divides the dose/volume edit results into a number of equal width bins.

Example: `nbin_DV 20`
Default: `nbin_DV 10`

The default value results in data bins from 0-10%, 10-20%, ... 90-100%, and >100%. This directive is active only with the `DVbs` directive.

D.4.9. **rbe**

This directive provides the set of RBE values to use for scaling the dose results. There are a total of six RBE values in use for conventional energies (below 16.9MeV) and an additional two for ultrafast applications. The eight values modify, respectively, the boron, gamma, nitrogen, hydrogen, other, gamma production, proton recoil ($E > 16.9\text{MeV}$), and ultrafast dose components. Default values for all are 1.0.

Example: `rbe 1.3 1.0 3.2 3.2`

This directive applies to all edit directives that follow and is active until superseded.

D.4.10. **images**

This directive defines the field of view (FOV), coordinate axis mapping (`in_x`, `in_y`, `in_z`), and the number of x- and y-grid points to be written to a contour data file by a contour directive. The field of view, which is the height and width of the image, is specified in centimeters. The axis mapping values map the coordinate axes from model space to image space. Generally, this mapping is not necessary, as `seraModel` automatically makes the default transformation for human images. However, for other image sets, such as dogs, the model coordinates do not map the same as for humans, making the mapping necessary. The number of points in the x- and y-directions should be the same, as the performance of non-square contour data arrays is not guaranteed. An example of the `images` directive is shown in Section D.4.11.

D.4.11. **ap**, **bp**, and **cp**

These three directives, which, when used, must always be used together, specify a geometric transformation from image (patient) space to model space in the `seraMC` edits. These directives are used in conjunction with the `images` and `contour` directives. The `ap`, `bp`, and `cp` directives define the transformation of a vector \mathbf{x}_b in image space to a vector \mathbf{x} in model space, as given by:

$$x(i) = ap(i) + bp(i) * \{xb(in(i)) + cp(i)\}$$

where the vector in is given on the images card as in_x, in_y, and in_z. These directives are rarely used, unless a non-default configuration for a contour data set is desired. Use of the ottocon directive is recommended instead.

```
Example:  images  26.0 2 1 3 40 40
          ap      0.0   0.0   0.0
          bp      0.076923 0.076923 0.076923
          cp      0.0   0.0   0.0
          contour "AX_1.014" -7.25
```

Usually, the ap and cp vectors are zero, and the bp vector provides a scaling factor for the transformation. This scaling factor is 2.0/FOV.

D.4.12. skinentry

This directive provides additional information needed by the crosshair edit directive. It gives the distance along the beam centerline from the beam exit plane to the entry point on the skin, in centimeters. This directive may be useful for patient positioning.

```
Example:  skinentry  30.0
Default:  skinentry  0.1
```

D.5. Active Edits

D.5.1. point

This directive generates dose and flux results at the specified spatial point. All dose and flux components are edited. The point is specified by its coordinates in centimeters.

```
Example:  point  1.0 1.0 3.5
```

```
Other required directives:  none
Other useful directives:    b10_blood, rbe
```

D.5.2. line

The line edit directive defines two points, and performs a dose edit along the line between these points. The spacing between edit points is defined on the delta directive. All locations are given in centimeters.

```
Example:  line  1.0 1.0 10.0  1.0 1.0 2.0
```

```
Other required directives:  delta
Other useful directives:    in_reg
```

D.5.3. px, py, and pz

These directives generate 20x20 grids of dose and flux edits for the specified plane. The px directive holds x constant, py holds y constant, and pz holds z constant. The information is written to the file sur.plot.

```
Example:  px  1.0 -10.0 10.0 -20.0 0.0
```

This example would generate data for the plane defined by $x=1.0$, $-10.0 < y < 10.0$, $-20.0 < z < 0.0$. These directives have been superseded by the `ottocon` directive.

Other required directives: `delta`
Other useful directives: `in_reg`

D.5.4. contour

The `contour` edit directive creates a set of files to provide information to `seraDose` and `sera3d` for isodose contour generation. Three files are generated for each contour directive - a contour header file `.chd`, a contour data file `.cdf.sz`, and a mask file `.cmf.sz`. The data and mask files are both compressed after generation. The number of points generated is determined by the `images` directive. An example contour edit directive is shown in Section D.4.11.

The first parameter on the `contour` directive provides the name of the file to which the dose information is written. The second parameter gives the axial position, in cm, for which the dose information is calculated.

Other useful directives: `in_reg`

D.5.5. ottocon

This directive generates contour information for all image slices within a specified region, and writes these to a single set of contour files.

Example: `ottocon 40 40 target`

The first two values define the number of grid points to generate for the contour data file, and the third value specifies the region. Contour information is generated for all image slices that intersect this region. The files created are the same as those from the `contour` directive, and all bear the prefix "ottocon". These files are renamed in the `seraMC` run script to correspond to the requested `univel` filename.

Other required directives: `none`
Other useful directives: `none`

D.5.6. box

This directive performs a dose-volume edit, with the integration restricted to the space enveloped by the `box`, in the regions specified on the `in_reg` acceptance list. The box dimensions are all given in centimeters. The values are respectively the upper and lower boundaries of the `x`, `y`, and `z` coordinates of the box.

Example: `box 1.0 5.0 3.0 6.0 3.5 7.5`

If a dose-volume edit for an entire region is desired, it is simplest to use the `DVbs` edit directive, which finds and defines the bounding box automatically.

Other required directives: `delta, eps`
Other useful directives: `b10_blood, b10_ratio, in_reg, rbe`

D.5.7. DVbs

This directive performs a dose-volume integration over all regions defined in the `in_reg` acceptance list. Dose-volume histogram data are calculated for all dose components and for the total dose, using the previously specified RBE values, isotope densities, and boron concentrations. Bins for the histogram results are determined by the `nbin_DV` directive. Note that, if multiple regions are specified in the acceptance list, the integration is performed over the entire volume defined by these regions and not over each region separately.

Other required directives: `delta, eps`

Other useful directives: `b10_blood, b10_ratio, in_reg, nbin_DV, rbe, N_avg`

D.5.8. fiducial

This directive specifies the location of a fiducial marker in the model space and calculates the three directional offsets (left-right, anterior-posterior, inferior-superior) to the skin entry point. This directive may be useful for patient positioning.

Example: `fiducial -4.5 5.0 9.2`

Other required directives: `none`

Other useful directives: `none`

D.5.9. ras

This directive generates a raster image file on a specified plane, for a uniform grid, with the distance between points determined by the `delta` directive. The data are written to a file `raster.n`, where `n` indicates the number of `ras` directives executed.

Example: `ras -4.0 5.0 3.0 12.0 7.0 7.0`

This example would generate data for the plane $-4.0 < x < 5.0$, $3.0 < y < 12.0$ at $z=7.0$. Note that one of the coordinate pairs must be degenerate.

Other required directives: `delta`

Other useful directives: `none`

D.5.10. beamplt

The `beamplt` directive causes edits to be generated for a plane through the beam line. A set of contour data is created for the plane and two line edits parallel to the beamline are also created. The directive input is given as follows:

Example: `beamplt nx ny maxis fov rbeam imname`

where `nx` and `ny` specify the grid for the contour data, `maxis` defines the axis normal to the plane, `fov` gives the desired field of view in centimeters, `rbeam` specifies the distance in centimeters from the beam line for the line edits, and `imname` is the prefix for the contour files. The `maxis` parameter forces the values normal to the plane to have constant value on the axis specified, with 1, 2, and 3, denoting the x-, y-, and z-directions, respectively, as possible values.

Other required directives: `delta`

Other useful directives: `in_reg`

D.5.11. N_avg

This directive induces the dose-volume integrations to provide additional information. The dose-volume integration will, in addition to the total dose-volume integral, produce volume-averaged dose data for N_avg equal volume intervals within the region. These sub-regions are determined by ordering the dose values within the region, then dividing the list into N_avg equal volume intervals and averaging the dose over each of these sub-volumes. Note that these sub-volumes will not necessarily be contiguous spatial regions. The results are then ordered by average dose.

Example: N_avg 3
Default: N_avg 1

Other required directives: DVbs
Other useful directives: none

D.5.12. crosshair

This directive calculates the distance from the entry point, defined by the skinentry directive, to the targeting laser crosshair location.

Example: crosshair 12.0
Default: crosshair 10.0

The value specified is the distance from the beam exit plane to the laser crosshair location, in centimeters.

Other required directives: skinentry
Other useful directives: none

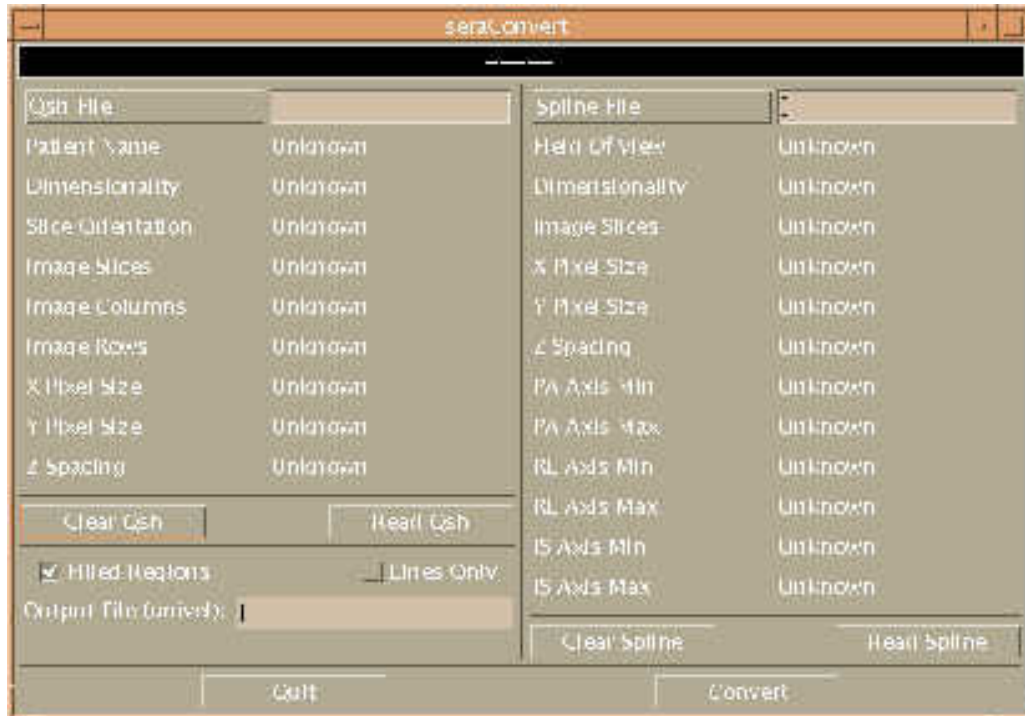
Appendix E. SeraConvert

E.1. Introduction

The seraConvert module provides a method for converting the old b-spline geometry files, used in the BNCT_rtpc codes, to univel files for use in Sera. The following sub-sections provide a step-by-step guide to performing this conversion.

E.2. Invoke seraConvert

Begin seraConvert by typing seraConvert on the command line in the directory of the spline file you would like to convert. The seraConvert interface is shown below.



E.3. Load QSH and Spline Files

You'll notice two buttons for loading files. The "QSH File" button is used to gather information about the images used to define the spline file. If you don't have a QSH file for the patient, you will probably want to build one using seraImage. This way you will be able to view the univel files in seraModel. The other button is the "Spline File" button. This is the button to load the actual spline file.

Note: You can convert a spline file to univels without loading the QSH file, but if the spline file is spaced unevenly, or has a different number of images with defined bodies, the univels may not match up correctly with the image set.

E.4. Making Sure all Bodies Are Known

When you load the spline file, seraConvert may close and give a warning at the command prompt about adding an unknown body to body_order.txt. This file contains a list of bodies that are found in spline files. It is used by seraConvert to determine the order to draw the regions. Roughly, it is sort of a body containment file.

E.5. Build the Univel Files

Once you have loaded the spline and QSH (if desired) files, press Convert. SeraConvert will take a moment to convert the file and should display “Done!” when it is finished.

E.6. Verifying Results

Check the .uv files with seraModel. You should load the images and univel files in seraModel to make sure they look correct. Occasionally, problems have arisen when a region has filled outside instead of inside the spline. If this happens, one 'work-around' provided is to click “Lines Only” before converting. You will then have to manually floodfill inside the body contours in seraModel to get filled regions.

E.7. Viewing Univel Files in SeraModel

SeraModel may not recognize the body names from the univel file created from the spline file. That is because body_data.txt doesn't contain these bodies. You'll have to edit this file using seraModel. Just open seraModel, click “Edit Bodies/Materials” and add the appropriate bodies. You will probably want to save this as a new file, such as body_data_splines.txt, and load it next time you start seraModel for viewing the spline data. Lastly, seraConvert doesn't define colors for the bodies it finds in the spline file. So, seraModel may display them as black. You can adjust the colors to your liking. Once you have done this, you can save the .uv file in seraModel and it should look the same next time you load the file.

Appendix F. dicom2qsh

F.1. Introduction

The dicom2qsh utility code is provided to allow Sera to use DICOM formatted image files. It is a stand-alone program that reads a set of DICOM images and generates a QSH file pair. Note that this code will only read and process image data for DICOM-RT files; other data and the general DICOM image formats will produce errors.

F.2. Installing dicom2qsh

The dicom2qsh source code is provided in the file dicom.tar.gz in the Sera distribution. To build and install the code, the following series of command line directives are necessary:

```
gunzip dicom.tar.gz
tar xf dicom.tar
cd dicom2qsh
make
mv dicom2qsh $SERA_HOME/Target/bin
```

This will build the dicom2qsh executable, called dicom2qsh, and install it in the Target/bin directory in the Sera home directory.

F.3. Converting DICOM to QSH

The dicom2qsh code is executed from the command line by the command:

```
dicom2qsh qshfile dicomfilelist
```

where qshfile is the prefix for the QSH file pair to be generated and dicomfilelist is the list of the DICOM image files to be included in the conversion. The list of DICOM files can contain Unix wildcards, so that each individual filename does not need to be provided. The QSH file pair will be written in the directory in which dicom2qsh was executed, unless a full path is defined as part of the prefix.