

DOCTORAL DISSERTATION

MIGUEL LÁZARO RODRÍGUEZ CASTILLO



# Automation of the Monte Carlo simulation of medical linear accelerators



UNIVERSITAT POLITÈCNICA DE CATALUNYA  
Institut de Tècniques Energètiques

*Barcelona, November 20<sup>th</sup>, 2015*

DOCTORAL DISSERTATION

AUTOMATION OF THE MONTE CARLO SIMULATION OF  
MEDICAL LINEAR ACCELERATORS

MIGUEL LÁZARO RODRÍGUEZ CASTILLO



Institut de Tècniques Energètiques  
Universitat Politècnica de Catalunya

Barcelona - November 20<sup>th</sup>, 2015

Co-supervisors:

Priv.-Doz. Dr. Lorenzo Brualla  
Klinik- und Poliklinik für Strahlentherapie  
Universitätsklinikum Essen  
Universität Duisburg-Essen

Dr. Josep Sempau  
Institut de Tècniques Energètiques  
Universitat Politècnica de Catalunya

*Para mis padres, mi esposa y mis hijos*





*The enjoyment of one's tools is an essential ingredient of successful work*

— Donald E. Knuth

## PUBLICATIONS

---

Publications in scientific journals related to this thesis:

- **M. Rodriguez**, J. Sempau, A. Fogliata, L. Cozzi, W. Sauerwein, and L. Brualla. A geometrical model for the Monte Carlo simulation of the TrueBeam linac. *Phys. Med. Biol.*, 60:N219–N229, 2015.
- **M. Rodriguez**, J. Sempau, and L. Brualla. Study of the electron transport parameters used in PENELOPE for the Monte Carlo simulation of linac targets. *Med. Phys.*, 42:2877–2881, 2015.
- M. F. Belosi, **M. Rodriguez**, A. Fogliata, L. Cozzi, J. Sempau, A. Clivio, G. Nicolini, E. Vanetti, H. Krauss, C. Khamphan, P. Fenoglietto, J. Puxeu, D. Fedele, P. Mancosu, and L. Brualla. Monte Carlo simulation of TrueBeam flattening-filter-free beams using Varian phase-space files: Comparison with experimental data. *Med. Phys.*, 41:051707, 2014.
- **M. Rodriguez**, J. Sempau, and L. Brualla. PRIMO: A graphical environment for the Monte Carlo simulation of Varian and Elekta linacs. *Strahlenther. Onkol.*, 189:881–886, 2013.
- **M. Rodriguez**, J. Sempau, and L. Brualla. A combined approach of variance-reduction techniques for the efficient Monte Carlo simulation of linacs. *Phys. Med. Biol.*, 57:3013–3024, 2012.

Additional publication in a scientific journal related to this thesis:

- **M. Rodriguez**. PENLINAC: extending the capabilities of the Monte Carlo code PENELOPE for the simulation of therapeutic beams. *Phys. Med. Biol.*, 53:4573–4593, 2008.

Published technical reports:

- L. Brualla, **M. Rodriguez**, and J. Sempau. *PRIMO User's Manual*. <http://www.primoproject.net>, 2015.
- L. Brualla, **M. Rodriguez**, and J. Sempau. *PRIMO Quick Start Guide*. <http://www.primoproject.net>, 2014.

Manuscript submitted to a scientific journal:

- L. Brualla, **M. Rodriguez**, and A. Lallena. Review Article. Monte Carlo systems used for treatment planning and dose verification of linac external beam radiotherapy. Submitted to *Med. Phys.*.

Manuscript in preparation to be submitted to a scientific journal:

- **M. Rodriguez**, L. Brualla, and J. Sempau. Optimization of the efficiency in Monte Carlo simulations of linacs that employ multiple variance-reduction techniques.

Presentations in scientific congresses:

- L. Brualla, **M. Rodriguez**, and J. Sempau. *PRIMO: A graphical user-friendly environment for the automatic Monte Carlo simulation of linacs and dose distributions in patients*. Awarded with the Poster Prize during the annual Congress of the Deutsche Gesellschaft für Radioonkologie (German Society for Radiation Oncology, DEGRO). 2012. Published in *Strahlenther. Onkol.*, 188:151, 2012.
- L. Brualla, J. Sempau, and **M. Rodriguez**. *Towards bridging the gap between Monte Carlo and the clinical environment: Variance-reduction techniques for linac simulation with PENELOPE / PRIMO*. Poster presentation in the Congress of the European Society for Radiotherapy and Oncology. ESTRO Congress 31. Barcelona. May, 2012. Published in *Radiother. Oncol.*, 103:S313, 2012.
- L. Brualla, **M. Rodriguez**, and J. Sempau. *PRIMO: A graphical environment for the automatic simulation of Varian and Elekta linacs with PENELOPE*. Oral presentation in the Congress of the European Society for Radiotherapy and Oncology. ESTRO Congress 31. Barcelona. May, 2012. Published in *Radiother. Oncol.*, 103:S246, 2012.
- L. Brualla, **M. Rodriguez**, L. Cozzi, A. Fogliata, W. Sauerwein, and J. Sempau. *An unguine complete description of the TrueBeam linac for Monte Carlo simulation*. Electronic Poster presentation in the ESTRO Congress 33. Vienna. April 4-9, 2014. Published in *Radiother. Oncol.*, 111:S12-S13, 2014.
- **M. Rodriguez**, L. Brualla, and J. Sempau. *PRIMO. A graphical environment for the Monte Carlo simulation of Varian and Elekta linacs*. Pre-congress course in the XV Workshop on Nuclear Physics and IX International Symposium on Nuclear and Related Techniques. Havana. February 9, 2015.

## ACKNOWLEDGMENTS

---

It is a fortunate and joyful occasion when supervisors and student work as a united team. Degrees aside, they all row—sometimes sail—to one goal. I thank Lorenzo Brualla and Josep Sempau for being my team mates. Along this journey I have learned a world from them.

I want to express my deep gratitude to the following people:

Dr. Luca Cozzi, Clinical Research Scientist at the Istituto Clinico Humanitas, Milan, for his help and support in the distribution of PRIMO as well as for his valuable advice.

Prof. Dr. med. Wolfgang Sauerwein, leitender Oberarzt of the Strahlenklinik at the Universitätsklinikum Essen, for support and help in the distribution of PRIMO.

Prof. Dr. Pedro Andreo from the Karolinska Institutet, Sweden, for beta testing the PRIMO system and help in its dissemination.

Ms. Antonella Fogliata, medical physicist at the Istituto Clinico Humanitas, Milan, for beta testing the PRIMO system and experimental measurements on the TrueBeam linear accelerator.

Ms. Francesca Belosi, medical physicist at the Paul Scherrer Institut in Villigen, Switzerland, for extensively beta testing the PRIMO system.

Mr. Marcelino Hermida, medical physicist at the Hospital Universitari Vall d'Hebron in Barcelona, for exhaustive testing the code and valuable recommendations.

Prof. Dr. Antonio M. Lallena of the Universidad de Granada, for his valuable advice in reviewing other Monte Carlo systems.

Ms. Nuria Escobar, medical physicist at the Universitätsklinikum Aachen for providing experimental measurements on the Elekta linear accelerator.

Dipl.- Phys. Andrea Flühs, medical physicist at the Universitätsklinikum Essen for measurements on Varian linac machines and valuable comments about the PRIMO system.

Mr. Josep Pujal, system administrator of the Argos computer cluster at the Universitat Politècnica de Catalunya.

Mr. Michael Nieporte, system administrator of the Contessa computer cluster at the Universitätsklinikum Essen.

Ms. Ina Grübel, documentalist of the Strahlenklinik at the Universitätsklinikum Essen.

To the many PRIMO users, for their comments and suggestions that continuously help us to improve the code.

I am also grateful to the following institutions that have supported my research:

The Deutsche Forschungsgemeinschaft (projects BR 4043/1-1 and BR 4043/3-1).

The Spanish Ministerio de Economía y Competitividad (project FIS2012-38480).

The Zentrale Informationstechnik at the Universitätsklinikum Essen.

The program of Becas de la Cátedra de Seguridad Nuclear ARGOS supported by the Consejo de Seguridad Nuclear and the Escola Tècnica Superior d'Enginyeria Industrial de Barcelona for granting me travel stipends in the years 2009, 2010, 2012, 2013 and 2014.

The IFORES program of the Medizinische Fakultät at the Universität Duisburg-Essen.

The Grup Consolidat 2014SGR846 from the Spanish Generalitat de Catalunya.

## CONTENTS

List of Figures	xiii
Listings	xv
Acronyms	xvi
1 INTRODUCTION	1
1.1 Background	1
1.2 Objectives	2
1.3 Radiation therapy	2
1.4 The medical linear accelerator (linac)	3
1.5 Modern treatment modalities	7
1.6 Treatment planning	8
1.7 The Monte Carlo method for radiation transport	10
1.8 Variance-reduction techniques	13
1.8.1 The Monte Carlo code PENELOPE	14
1.8.2 Simulation of linear accelerators	15
1.8.3 Phase-space files	17
1.8.4 Dose assessment in the patient	19
1.9 Outline of the thesis	21
2 MONTE CARLO SYSTEMS USED FOR TREATMENT PLAN- NING AND DOSE VERIFICATION OF LINAC EXTERNAL BEAM RADIOTHERAPY	25
3 PRIMO: A GRAPHICAL ENVIRONMENT FOR THE SIMULA- TION OF VARIAN AND ELEKTA LINACS	67
4 A COMBINED APPROACH OF VARIANCE-REDUCTION TECH- NIQUES FOR THE EFFICIENT MONTE CARLO SIMULATION OF LINACS	75
5 OPTIMIZATION OF THE EFFICIENCY IN MONTE CARLO SIMULATIONS OF LINACS THAT EMPLOY MULTIPLE VARIANCE- REDUCTION TECHNIQUES	89
6 STUDY OF THE ELECTRON TRANSPORT PARAMETERS USED IN PENELOPE FOR THE MONTE CARLO SIMULATION OF LINAC TARGETS	101
7 MONTE CARLO SIMULATION OF TRUEBEAM FLATTENING- FILTER-FREE BEAMS USING VARIAN PHASE-SPACE FILES: COMPARISON WITH EXPERIMENTAL DATA.	107
8 A GEOMETRICAL MODEL FOR THE MONTE CARLO SIMU- LATION OF THE TRUEBEAM LINAC	119
9 SUMMARY AND CONCLUSIONS	131
A PRIMO DISTRIBUTION	133
B GENERATION OF LINAC GEOMETRIES	135
C GLASS CODE	157
C.1 Software tools and techniques	157

C.2 Algorithms . . . . .	158
D PRIMO: USER'S MANUAL AND QUICK START GUIDE	177
REFERENCES	286



## LIST OF FIGURES

---

Figure 1	Schematic diagram of a typical medical linear accelerator with the accelerating waveguide mounted parallel to the gantry rotation axis. Four major systems can be identified, namely, the <i>injection system</i> , the <i>RF system</i> , the <i>beam transport system</i> and the <i>beam collimation and monitoring system</i> which is represented twice, in its typical configuration for a photon beam (a) and an electron beam (b). . . . .	3
Figure 2	Degrees of freedom of the treatment head and table of a linac. . . . .	5
Figure 3	A 80-leaf multileaf collimator. Adapted from <a href="http://newsroom.varian.com">http://newsroom.varian.com</a> . . . . .	6
Figure 4	A radiation field from a BEV perspective. The field shape is adjusted to the PTV contour (in red) with the MLC. The projection of the patient in this perspective is obtained with a digitally reconstructed radiograph (DRR). . . . .	8
Figure 5	Examples of Varian and Elekta linear accelerator geometries generated with the tools developed in this thesis. . . . .	17
Figure 6	Above: Views of the coded geometry of the Elekta MLCi. The upper view is a transversal section of the MLC showing the fan-shaped arrangement of the leaves. The lower view is a lateral section of a leaf (in a different scale). Colors differentiate materials (at left) and geometry elements (at right). Below: a section of the coded geometry of a Varian Millennium 120-leaf MLC with a similar leaf arrangement to the MLC shown in figure 3. . . . .	18
Figure 7	Data flow among the layered components of PRIMO. . . . .	158
Figure 8	The <i>TProject object</i> and its related objects. Discontinuous lines represent an ownership relation, continuous lines represent inheritance . . .	160
Figure 9	Objects representing a volume (3D matrix) of data. . . . .	161
Figure 10	Trilinear interpolation. See text for details. . .	165

Figure 11	A set of regions of interests are represented in an array of 64-bits unsigned integers elements. Each bit of the data element indicates whether or not the corresponding array element in the CT volume is inside the ROI. . . . .	166
Figure 12	Geometry for the construction of a DRR. The focus point $p_f$ , representing the source of X-rays, is at 100 cm from the isocenter which is the origin of coordinates. The CAX of the linac coincides with the z-axis. A ray (in red) is cast from the point $p_f$ to the point $p_i$ and its total attenuation is calculated and stored in the DRR's pixel. . . . .	169
Figure 13	Calculation of the gamma index for a reference point p. See text for details. . . . .	172
Figure 14	Particle processing pipeline. The particle is filtered at each processing step according to its nature, the material where it was created, its energy, flight direction and position. . . . .	173

Additional figures are included in embedded documents.

## LISTINGS

---

Listing 1	A simplified version of the <i>TProject.Textify</i> method. . . . .	159
Listing 2	Polymorphic pointer used in the <i>TVolume</i> object to address a data element of the 3D matrix. . . . .	161
Listing 3	Fast bilinear interpolation of a 2D matrix . . . . .	163
Listing 4	listing 3 continued . . . . .	164
Listing 5	A function to identify that a pixel is in the contour of a ROI. . . . .	168
Listing 6	A simplified version of the method <i>Tpipeline.pipeall</i> that implements the processing of the particles stored in a phase-space file. . . . .	175

## ACRONYMS

---

3DCRT	tridimensional conformal radiation therapy
API	application programming interface
BEV	beam's eye view
CAD	computed-aided design
CAX	central axis
CPU	central processing unit
CSDA	continuous-slowing-down approximation
CT	computed tomography
DICOM	digital image communication in medicine
DLL	dynamic-link library
DPM	Dose Planning Method
DRR	digitally reconstructed radiograph
DTA	distance-to-agreement
DVH	dose-volume histogram
EGS	Electron-Gamma Shower
FFF	flattening-filter-free
FLUKA	FLUktuierende KAskade
Geant	GEometry ANd Tracking
GGPB	generalized Gaussian pencil-beam
GLASS	graphical layer for the automation of the simulation system
GPU	graphic processing unit
GUI	graphical user interface
IAEA	International Atomic Energy Agency
ICRP	International Commission on Radiological Protection
ICRU	International Commission on Radiation Units and Measurements
IDE	integrated development environment

IMAT	intensity-modulated arc therapy
IMRT	intensity-modulated radiation therapy
linac	linear accelerator
MCNP	Monte Carlo N-particle
MIC	many integrated core
MLC	multileaf collimator
MRI	magnetic resonance imaging
MU	monitor unit
OAR	organ-at-risk
OOL	object-oriented language
OS	operating system
PBC	pencil-beam convolution
PENELOPE	PENetration and Energy LOSS of Positrons and Electrons
PET	positron-emission computed tomography
pixel	picture element
PSF	phase-space file
PTV	planning target volume
RF	radiofrequency
ROI	region of interest
SPECT	single-photon emission computed tomography
TPS	treatment planning system
US	ultrasound
VMAT	volumetric-modulated arc therapy
voxel	volume element
VRT	variance-reduction technique

## INTRODUCTION

---

### 1.1 BACKGROUND

Radiotherapy is the process of treating a disease, especially malignant tumors, by exposing the affected tissues to ionizing radiation. A common technique employed in radiotherapy consists of irradiating the patient externally using the electron or photon beam produced by a medical linear accelerator (linac) [123]. Prior to irradiation, the absorbed dose distribution in the patient is assessed to maximize the dose delivered to the tumor and to minimize the dose received by the nearby normal tissues [55]. In the routine clinical practice this task is performed by treatment planning systems (TPSs). Currently, most of these systems are based on analytical algorithms which may give rise to relevant inaccuracies when used to compute small irradiation fields or in the presence of large mass density gradients (see *e. g.*, [89]).

An alternative to analytical algorithms is the use of the Monte Carlo method. This stochastic approach can simulate radiation beams produced in a linac and estimate the absorbed dose distribution with high accuracy. A drawback that has hindered its use in the routine clinical practice is the long computation time required for obtaining results with acceptable statistical uncertainty. Additionally, the setup of a linac simulation and the determination of the dose with the Monte Carlo method is an elaborate and time consuming problem which requires specific knowledge. For these reasons Monte Carlo simulation of linacs has been extensively used only in research studies which do not have the strict time limits associated to the clinical practice.

Setting up a Monte Carlo simulation of a linac, and the corresponding dose distribution in a patient, involves the following tasks: (i) obtaining a detailed geometrical description of the linac head from the manufacturer; (ii) coding this description into the Monte Carlo system; (iii) configuring the simulation to reproduce reference experimental data from the linac; (iv) adapting the description of the patient from a computerized tomography to the requirements of the simulation system; (v) optimizing the simulation time of the Monte Carlo run, *e. g.*, with parallelization methods or other software techniques; (vi) analyzing the results. Most existing simulation systems require that users manually perform tasks (i) and (ii); others require additional external tools to perform tasks (iv) and (vi); finally, some systems are not specifically designed to cope with task (v). To the

*necessary tasks for  
the Monte Carlo  
simulation of linacs  
and dose  
distributions*

best of our knowledge, at the time of starting this thesis there existed no publicly available system, either commercial or free, capable of performing all these tasks with minimal intervention from the end user.

The present introductory chapter provides an overview of the basic concepts related with the Monte Carlo simulation of linacs and dose distributions.

## 1.2 OBJECTIVES

The purpose of this thesis is to develop a publicly available, freely distributable linac and dose simulation system based on an extensively tested general-purpose Monte Carlo code, that automates tasks (i)-(vi) above under an intuitive and user-friendly graphical interface.

For this purpose the following goals are defined:

1. To generate, without user intervention, linac geometries of the most common linac manufacturers by means of a software package.
2. To facilitate the management of computerized tomographies and their use in a software environment for Monte Carlo simulations.
3. To speed up Monte Carlo linac simulation by means of new optimization algorithms and automatic parallelization strategies.
4. To furnish users with a practical and easy-to-use interface to analyze the features of radiation fields and dose distributions with techniques commonly used in medical physics.
5. To integrate the aforementioned tasks (i)-(vi) into a single, self-contained software system with an intuitive graphical user interface. The new system shall be conceived as a tool for medical physics applications in research, education, quality assurance and clinical practice.

## 1.3 RADIATION THERAPY

Radiation therapy plays a crucial role in the treatment of cancer. It promotes tumor control, decreases the risk of cancer recurrence and is effective in palliative care. Approximately 3.2 million new cancers are diagnosed in Europe per year and roughly 55% of those require radiotherapy treatment. In 2012 Europe had 1286 radiotherapy centers with approximately 5.3 external beam radiotherapy machines per million inhabitants [97].

In external beam radiotherapy the source of radiation is placed externally to the patient and the beam is collimated to the tumor region. The most commonly used external beams are gamma rays emit-

*external beam  
radiation therapy*

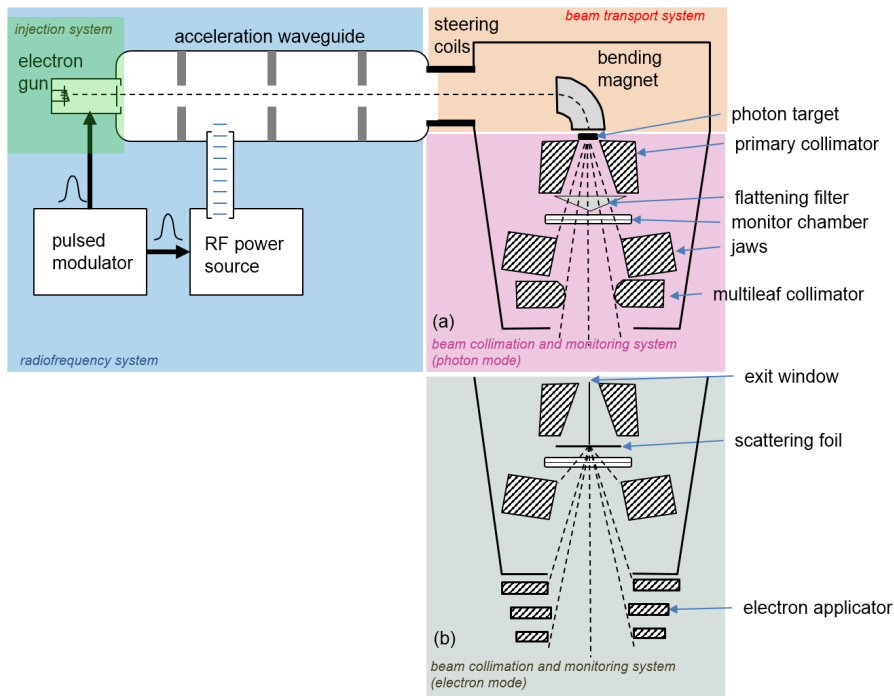


Figure 1: Schematic diagram of a typical medical linear accelerator with the accelerating waveguide mounted parallel to the gantry rotation axis. Four major systems can be identified, namely, the *injection system*, the *RF system*, the *beam transport system* and the *beam collimation and monitoring system* which is represented twice, in its typical configuration for a photon beam (a) and an electron beam (b).

ted from  $^{60}\text{Co}$  sources, X rays, high-energy electrons and photons from medical linacs and protons accelerated in cyclotrons. Heavy-ion beams accelerated to high energies in synchrotrons [58, 59] and epithermal neutron beams used in boron neutron capture therapy [50] are, comparatively, much less employed. Linacs, which are relevant for the scope of this thesis, are the most widely used external beam radiotherapy machines due to their versatility, compact design and efficiency.

#### 1.4 THE MEDICAL LINEAR ACCELERATOR (LINAC)

Medical linacs accelerate electrons to kinetic energies between 4 and 25 MeV using microwave fields. The patient can be irradiated either with these electrons or with photons produced via bremsstrahlung emission, situations that are referred to as electron and photon irradiation modes, respectively. The main four systems of a linac are (see figure 1):

*four main systems of a linac*



- The *electron injection system* is an electrostatic accelerator, also referred to as the electron gun. It consists of a heated cathode and a perforated grounded anode. Electrons are thermionically emitted from the cathode, focused into a pencil beam and accelerated to the anode.
- The *radiofrequency (RF) system* is used to further accelerate electrons. It is mainly composed of a RF power generator, a modulator, and an accelerating waveguide.
- The *electron beam transport system* is used to transport the accelerated electrons from the waveguide to the vacuum exit window (figure 1). It comprises mainly the drift tubes and the bending magnets. Additionally, steering and focusing coils are used to keep the electron beam centered and to reduce its divergence. The bending magnets are used to change the direction of the electron beam in those systems where the waveguide is parallel to the gantry rotation axis (see figure 2). In low energy linacs, in which the short waveguide can be mounted in the same direction of the treatment head central axis (CAX), bending magnets are unnecessary.
- The *beam collimation and monitoring system*, or treatment head, is used to produce the clinical beam. It consists of a series of components to shape the beam and to modulate its intensity. This is the region of the linac that is modeled in order to perform the transport of particles with the Monte Carlo method. Due to its importance for this thesis this system shall be described in more detail.

When the linac operates in photon mode (figure 1a) the electron beam impinges on a small cylindrical block, called target, made of a high atomic number material. In the target photons are produced by bremsstrahlung emission.

*linac photon target*

Targets can be classified as thin, medium and thick according to the continuous-slowing-down approximation (CSDA) electron range,  $R$ , in the target material. The thickness  $\phi$  of thick targets is  $\sim 110\% R$ ; for thin targets  $\phi \sim 2\% R$  [92]. Thin targets produce photon spectra with higher mean energy, but transmit more incident electrons. In the most widely used linac the target is usually constructed as a two layered block: one thin layer made of a high density material like tungsten on top of, or embedded in, a much thicker layer of a heat dissipating and electron absorber material such as copper.

At megavoltage energies the polar angular distribution of bremsstrahlung photons is forward peaked. This gives rise to centrally-peaked dose profiles not suitable for most conventional radiotherapy treatments. Flattening filters are used to equalize the beam radial intensity by augmenting the beam attenuation from the periphery to

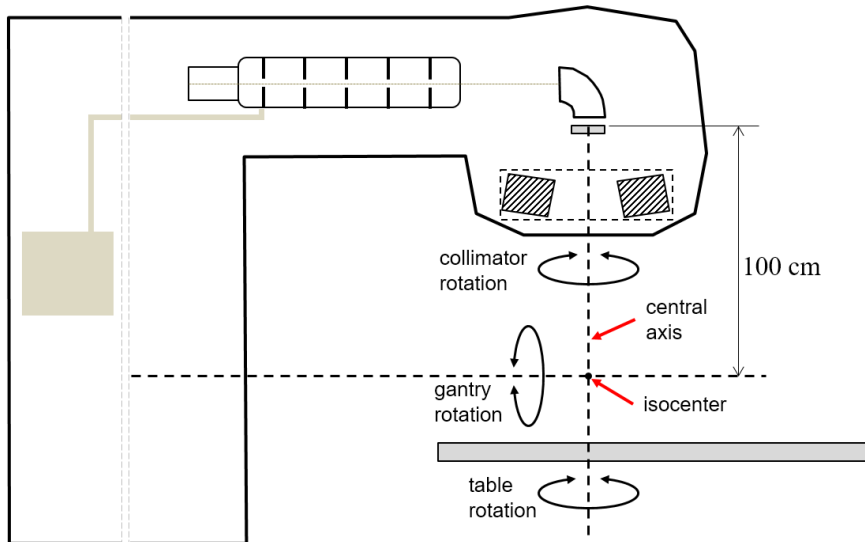


Figure 2: Degrees of freedom of the treatment head and table of a linac.

the center<sup>1</sup>. They have an approximately conical shape and are usually made of midrange atomic number ( $Z$ ) materials such as copper and steel.

An ionization chamber, located downstream the flattening filter, is used to monitor the beam. These chambers are built with thin layers of low  $Z$  materials to reduce the perturbation of the beam. Absorbed dose in water depends on the chamber reading, specified in the so-called monitor units (MU), on the field size, the nominal beam energy, and on other configuration parameters of the linac.

*linac ionization chamber*

Beam collimation is used to shape the radiation field. It is achieved with two groups of structures named according to their position downstream the beam. The primary collimator is situated near the beam entry. Its conical aperture is used to define a large circular photon field on the flattening filter. Secondary collimators comprise the jaws and the multileaf collimator (MLC) (see below). The jaws are generally two upper (Y-jaws) and two lower (X-jaws) tungsten blocks used to produce a rectangular field. In some models the purpose of the upper Y-jaws is performed by combining the MLC with additional shielding elements. The size of the rectangular field defined by the jaws is specified at a plane perpendicular to the linac central axis located at a distance of 100 cm from the target (see figure 2). The maximum field size thus defined is of  $40 \times 40 \text{ cm}^2$  with penumbras<sup>2</sup> of a few millimeters.

*linac beam collimation structures*

The structure called MLC (figure 3) is formed by two banks of packed metal sheets known as leaves. There are in the range of 26 – 80

<sup>1</sup> Nowadays, flattening-filter-free (FFF) beams are used to treat small regions and maximize the dose rate.

<sup>2</sup> The region where the intensity of the radiation field falls drastically.

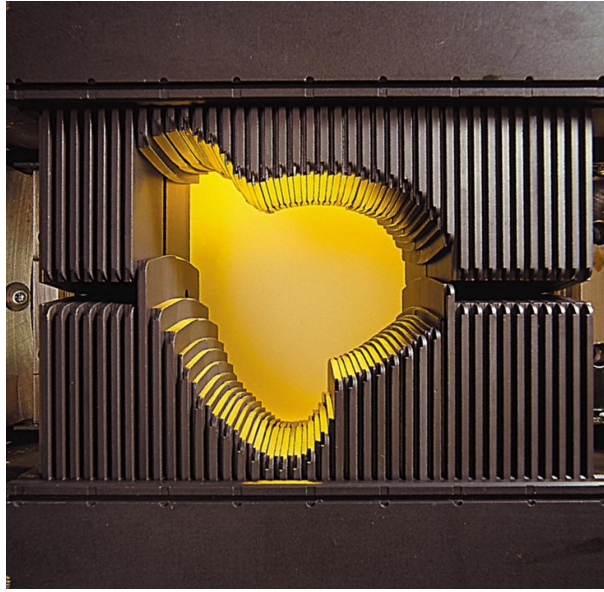


Figure 3: A 80-leaf multileaf collimator. Adapted from <http://newsroom.varian.com>.

opposed pairs of leaves depending on the MLC model. Each leaf is driven by a computer-controlled motor and can move independently from the others in and out of the radiation field. Leaves have a tongue-and-groove design to reduce the interleave radiation leakage. Originally the MLC was conceived to conform the shape of the field to the tumor contour. Later, it was found to be a convenient device for modulating the intensity of the beam [31].

*linac electron mode*

In electron mode linacs produce beams in the 6 – 25 MeV energy range. When operating in this mode the target and the flattening filter are removed from the beam path and, in lieu, a scattering foil is introduced to broaden the electron beam to allow for large treatment field sizes (figure 1b). Frequently, a combination of two scattering foils separated by a few centimeters is used to improve the flatness of the beam profile. In general, the MLC is left wide open to prevent interference with the electron beam. Rather, a removable structure called electron applicator is inserted at the downstream end of the linac head. Its purpose is to conform the beam to the tumor contour using dedicated metal cuts.

The treatment head can rotate  $360^\circ$  around an horizontal axis (see figure 2). This degree of freedom is referred to as gantry rotation. The secondary collimation structures can rotate around the central axis, defined along the beam path, a degree of freedom referred to as collimator rotation. The intersection of both axes is called the isocenter. It is located at 100 cm from the upstream surface of the target.

The patient is positioned on a treatment couch, also referred to as table, displayed in figure 2. The couch can also rotate around a

vertical axis that coincides with the central axis of the linac. In some models the couch can be displaced in the three spatial directions.

Current main vendors of medical linacs are Varian Medical Systems, Inc. (Palo Alto, CA, USA, <http://www.varian.com>) which holds approximately 50% of the world-wide market and Elekta AB (Stockholm, Sweden, <http://www.elekta.com>) which has about 30% of it.

*main linac vendors*

### 1.5 MODERN TREATMENT MODALITIES

Tridimensional conformal radiation therapy (3DCRT) is a treatment modality in which the shape of the radiation field is fitted to the contour defined by the projection of the planning target volume (PTV) [55] in the plane of the field<sup>3</sup>. In other words, it is adjusted to the contour of the PTV as seen by an observer located at the radiation source, a perspective known as the beam's eye view (BEV). The field shape is set with the MLC<sup>4</sup>. This is illustrated in figure 4.

*tridimensional conformal radiation therapy*

The intensity-modulated radiation therapy (IMRT) technique represents a step ahead in terms of achievable plan quality with respect to 3DCRT. Modulation can be seen as subdividing the beam in small beamlets, each with an arbitrary intensity. This can be accomplished by fabricating custom-designed tridimensional compensators, a method little used in practice due to its many limitations, and by computed-controlled positioning of MLC leaves [31]. Brahme et al. [16] showed that if the intensity is modulated across the radiation field a better dose distribution is achieved. Two alternative approaches to the delivery of IMRT using linac and MLC are commonly employed, namely, the step and shoot or the sliding-windows techniques. The former utilizes a sequence of multiple fields (segments) of varying shape. The beam is turned on to irradiate each field when the leaves are not moving. The number of segments can be from a few to several hundred. In the sliding-windows technique leaves are continuously moving while the beam is active. A different approach of intensity modulation delivery is achieved using a rotating fan beam and a binary leaf collimator, a treatment modality known as tomotherapy [81, 25].

*intensity-modulated radiation therapy*

Intensity-modulated arc therapy was first proposed by Yu [132] as an alternative to tomotherapy. It consists in delivering the intensity-modulated treatment during the continuous rotation of the linac gantry, as in tomotherapy, but in this case with the linac cone-shaped beam collimated by the MLC. In his seminal work, Yu predicted that by increasing the number of gantry angles, the number of intensity levels at each gantry angle can be reduced without degrading plan

<sup>3</sup> The PTV is a volume defined in the patient that contains the tumor, together with a margin for microscopic disease spread, plus an additional safety margin for uncertainties in planning or treatment delivery.

<sup>4</sup> Custom blocks made of high density metal are also used for this purpose.

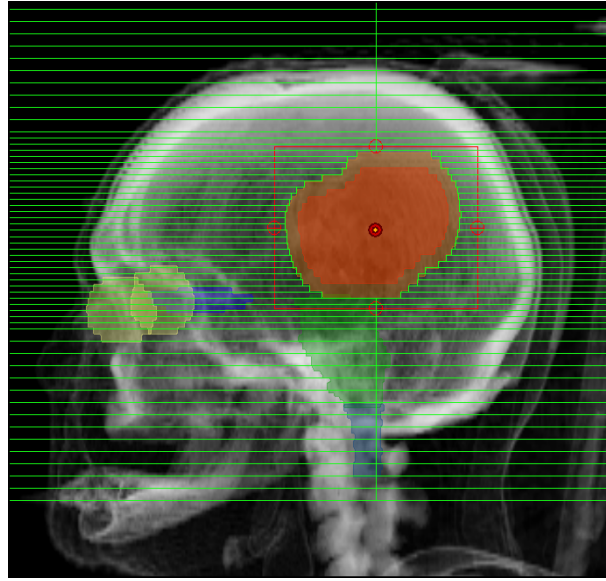


Figure 4: A radiation field from a BEV perspective. The field shape is adjusted to the PTV contour (in red) with the MLC. The projection of the patient in this perspective is obtained with a digitally reconstructed radiograph (DRR).

quality. It was not until 2008 that the first commercial implementation of intensity-modulated arc therapy (IMAT) was available. It appeared with the trade name of RapidArc and was marketed by Varian Medical Systems. RapidArc implements the algorithm developed by Otto [88] which includes more degrees of freedom to the dose delivery, *i. e.*, the rotation of the gantry with variable speed and also a variable dose-rate. In addition, it uses progressive beam angle sampling to optimize a large number of apertures. The term VMAT, introduced by Otto, is widely used to identify the single-arc IMAT technique that uses variable dose-rate.

#### 1.6 TREATMENT PLANNING

Prior to delivering a radiation treatment a comprehensive planning process is performed. The definition of aspects concerning dose prescription, fractionation, dose calculation, patient positioning and immobilization, treatment verification and linac setting is a requirement previous to treatment delivery. The ultimate goal of radiotherapy planning is to deliver the maximum dose to the tumor and the minimal dose to the rest of the body. In practice it is always a trade-off between achieving the prescribed dose in a large percentage of the tumor volume while keeping the dose to the surrounding normal tissues under certain tolerance levels. The process of identifying and delineating the contours of the relevant anatomical structures

in a patient image is crucial to the accuracy of the plan. The International Commission on Radiation Units and Measurements (ICRU) has issued guidelines to standardize dose prescription and report [55, 56, 57]. These guidelines also define the volumes that must be considered, including the PTV and the organs-at-risk (OARs).

Imaging plays a major role in radiotherapy treatment planning. The advent of computed tomography (CT) initiated an era of tridimensional radiotherapy [86]. Tridimensional planning systems started to be developed four decades ago and the effect of the presence of tissues with material composition different from water was incorporated in dose computation. Today, complementary techniques such as ultrasound (US), magnetic resonance imaging (MRI), single-photon emission computed tomography (SPECT) and positron-emission tomography (PET) are used in conjunction with the CT to estimate the extension of the disease. Radiographic projection remains an important tool in patient positioning and treatment verification.

*imaging in radiation therapy*

Forward planning was the only method used in radiotherapy until the nineties. In forward planning, the starting point for dose calculation is the definition of the beam energy and the direction, shape, modifiers and relative intensity of each field. Those parameters are established by the planner based on his training and experience. The obtained dose distribution is compared to the so-called plan objectives to determine whether or not adjustments in the plan parameters are necessary. This process is repeated until an optimized dose distribution is obtained. More recently an alternative approach, called inverse planning, was introduced. With this method the planner specifies the plan objectives and the TPS performs an optimization process to determine the parameters of the fields that better fulfill the objectives. Early ideas on inverse planning were proposed by Brahme and Bortfeld [16, 15].

*forward versus inverse planning*

Most modern treatment planning systems rely on deterministic (non-stochastic) algorithms for the determination of the dose distribution from electron or photon beams. For electron beams the generalized Gaussian pencil beam (GGPB) algorithm has been widely used for several decades. Briefly, a collimated electron beam is described mathematically as a grid of narrow beams. The lateral broadening of the absorbed dose from a narrow beam is approximated by a Gaussian distribution. Broad irregular beams are obtained by a summation over the narrow beams in the grid. Relatively large deviations of the dose of 5 to 10% are expected with this algorithm in irregular surfaces and in the presence of tissue inhomogeneities [100]. GGPB was developed at the Karolinska Institutet in Stockholm by Lax and co-workers for calculation of the dose produced by high-energy electron beams [74, 17]. Subsequently, Hyödynmaa developed a tridimensional version of the algorithm capable of handling arbitrary field

*deterministic dose calculation algorithm for electrons*



shapes and three-dimensional inhomogeneity corrections [52, 117, 100, 8].

*deterministic dose  
calculation  
algorithms for  
photons*

Early methods for dose determination in photon beams were empirical [9, 63, 114]. In those methods irregular fields were traditionally handled by the integration method of Clarkson [29]. In later times analytical algorithms, mostly based on the convolution and convolution/superposition ideas, were developed [6]. The general principle of these methods is the convolution of the energy fluence distribution with a kernel describing the spatial distribution of the energy deposition caused by a narrow monoenergetic beam in water. The kernel can be generated by means of Monte Carlo simulations [80] or by other methods such as deconvolution of measured dose curves [13]. The ideas introduced by these methods have evolved into several well-established algorithms, namely, pencil-beam convolution (PBC) [85, 121], analytical anisotropic algorithm (AAA) [122], and collapsed cone convolution (CCC) [5]. These analytical algorithms might produce inaccurate results in the presence of small radiation fields or large mass density gradients [124, 83, 7, 72, 89]. An alternative approach, fully developed in more recent times, is the numerical solution of the linear Boltzmann transport equation [130, 118, 44].

## 1.7 THE MONTE CARLO METHOD FOR RADIATION TRANSPORT

The Monte Carlo method uses random numbers to solve problems [98]. Propagation of radiation through matter is a process in which primary particles with a given energy penetrate a material medium suffering a series of interactions in which energy is transferred to the medium and secondary particles are produced. Although the principles of the interaction of particles with matter are well known, the mathematical description of the successive interactions undergone by an ensemble of particles is a complex problem that can be easily simulated with the Monte Carlo method.

*radiation transport  
process*

In brief, the application of the Monte Carlo method for radiation transport consists of the following steps: (i) A primary particle is generated in an initial state determined by its position, flight direction and kinetic energy. (ii) The particle is moved to a new position along a straight line following the initial direction of flight<sup>5</sup>, where the next interaction event is assumed to take place; the traveled distance is sampled from a decaying exponential probability distribution characterized by the mean free path of the particle in the material medium. (iii) The type of interaction is randomly selected according to the point probabilities associated to the associated cross sections of the considered interaction mechanisms. (iv) The interaction is simulated by changing the dynamical state of the particle and, possi-

<sup>5</sup> This assumption is valid in the absence of electromagnetic fields and when coherent effects are negligible.

bly, generating secondary radiation. (v) The process starts over at step (ii) and it is repeated until the particle is locally absorbed or when it escapes the material system; absorption occurs when the energy of the particle falls below some user-defined cutoff. Secondary particles are subsequently simulated in the same manner. The simulation of a primary particle and its descendants is called a 'shower' or a 'history'. The simulation stops when the number of simulated histories reaches a user-defined value. The relevant quantities of the problem under study, *e.g.*, the energy deposited in a detector, are tallied at the proper steps of the simulation of each particle and their average are obtained after all histories have been completed. Owing to the stochastic nature of the Monte Carlo simulation the tallied quantities have an inherent statistical uncertainty.

The transport geometry is a computer model of a physical environment. The algorithm determines intersections of the particle trajectory with the surfaces limiting the elements of the geometry. Most Monte Carlo codes include a software library that allows users to model geometries formed by elementary homogeneous bodies limited by quadric surfaces. In complex geometries the time devoted to this task can represent a significant fraction of the total simulation time. An alternative geometric model consists in dividing the space in a grid of parallelepipeds each containing a homogeneous material. Usually, the grid is regular and it is formed by a large number of parallelepipeds, called voxels, of size in the order of millimeters. This geometric model is, in general, more adequate than quadric surfaces for simulating anatomical structures. Voxel material composition and mass density can be obtained from a computerized tomography of the patient by means of a suitable mapping [125]. Other geometry models can also be used, such as triangle meshes employed in standard computer-aided design (CAD) tools (see *e.g.*, [116]).

The purpose of the simulation is the estimation of the expected values of some quantities of interest. More precisely, the estimate of a quantity  $Q$  is given by the arithmetic mean

$$\bar{Q} = \frac{1}{N} \sum_{i=1}^N q_i, \quad (1)$$

where  $q_i$  is the contribution of the  $i$ -th history to  $Q$ , and  $N$  is the total number of independent histories simulated. For a sufficiently large  $N$  this estimate converges to the expected value  $\langle Q \rangle$  and is normally distributed [62] with a standard deviation that can be estimated by

$$\sigma(\bar{Q}) \simeq \sqrt{\frac{1}{N} \left[ \frac{1}{N} \sum_{i=1}^N q_i^2 - \bar{Q}^2 \right]}. \quad (2)$$

Notice that  $\sigma(\bar{Q})$ , which measures the standard statistical uncertainty associated to the Monte Carlo estimate, decreases approximately as  $1/\sqrt{N}$  for large  $N$ .

Monte Carlo  
transport geometry

Monte Carlo  
estimator



The simulation algorithm described above in steps (i) to (v) is known as detailed simulation. When the mean free path between two consecutive interactions is small compared to the total distance traveled by a particle before it comes to a halt, the number of interactions to be simulated is extremely large and, therefore, detailed simulation becomes very slow. This is the situation found for charged particles, especially at high energies as those found in radiotherapy. For photons the number of interactions undergone is relatively low and detailed simulation is affordable.

*condensed transport  
of charged particles*

The problem for charged particles can be overcome by using the so-called condensed transport schemes, in which the effect of multiple interactions is described collectively in a single artificial step [40]. The total energy loss and angular deflection of the particle occurring along a step length is sampled from probability distributions obtained from multiple-scattering theories (see *e. g.*, [47, 48, 75, 73]).

In his seminal work Berger classified condensed-history methods in two classes [12]. In class I algorithms the length of the step is predefined. This scheme may present problems at the interfaces of two different materials because some steps may not be fully contained within a single material medium, a necessary requirement for the underlying multiple-scattering theories to be applicable. These limitations motivate the introduction of refinements in the algorithms such as progressively reducing the step length as the particle approaches an interface.

In contradistinction, class II algorithms<sup>6</sup> select step lengths stochastically. The scheme is based on classifying interactions into soft and hard events. Soft events involve energy losses and angular deflections below certain user-defined cutoffs. The remaining interactions are classified as hard and they are simulated detailedly, which implies that the distance between two consecutive hard events, the step length, is randomly sampled according to the usual decaying exponential distribution. The effect of the soft interactions occurring along the said step length is described by means of a single artificial event, thus speeding up the computation compared to the fully detailed case. The accuracy of the transport algorithm may depend on the selected step length, that is, on the selected cutoffs to distinguish hard from soft events. Usually, shorter steps imply better accuracy at the expense of a lower simulation speed.

<sup>6</sup> Also referred to as mixed simulation schemes.

## 1.8 VARIANCE-REDUCTION TECHNIQUES

A useful measure of the calculation performance is the simulation efficiency, which can be defined as

$$\epsilon = \frac{1}{T} \left( \frac{\bar{Q}}{\sigma(\bar{Q})} \right)^2, \quad (3)$$

where  $T$  is the computing time needed to obtain  $\bar{Q}$  with a given  $\sigma(\bar{Q})$ . For a given simulation the efficiency does not change significantly with the number of histories  $N$ , since  $N\sigma^2$  and the average computation time per history  $T/N$  are almost constant (see equation 2).

*simulation efficiency*

For some problems the efficiency can be exceedingly low, which might entail an unfordable long computation time. This has led to the development of the so-called variance-reduction techniques (VRT) [62, 60] aimed at increasing the efficiency  $\epsilon$ . In general, VRT consists in artificially altering the transport process to favor those events that produce larger contributions to the estimator of the quantity of interest. To keep results unbiased, the scored contributions are weighted by a factor, known as the statistical weight, associated to each simulated particle. Primary particles have an initial statistical weight set to unity. Secondary particles inherit the statistical weight of their parent. In analog simulations, those in which VRT are not applied, the statistical weight of the particles is not modified and remains equal to unity.

A wealth of VRT exists. In this section a selection of them, relevant for the purpose of linac simulation, will be briefly presented.

#### *Range rejection*

Range rejection is applied to charged particles. It consists of discarding the particle when its residual range is too small to reach the detector. A further requirement is that the radiation yield of the discarded particle should be below some predefined value.

Movable-skins [19] is a particular implementation of range rejection in which external layers of material structures are geometrically differentiated from internal regions. These external layers are called 'skins'. Charged particles in the internal non-skin regions are discarded by setting an arbitrarily high absorption energy.

*movable-skins*

#### *Interaction forcing*

Interaction forcing consists in artificially increasing the probability of some selected interaction mechanisms. This technique decreases the variance of the quantities directly related to the forced interaction mechanisms, but as more time is devoted to simulate the forced events other quantities may exhibit larger uncertainties for a given

simulation time. A detailed description of a particular implementation of this technique is given in reference [99].

#### *Russian roulette*

In Russian roulette a survival probability  $\kappa$  is assigned to a certain region of the space of states of the particle (*e.g.*, a volume in space or a kinetic energy interval). When a particle enters this region it is discarded with probability  $\kappa$ . If it survives its statistical weight is increased by a factor  $1/(1 - \kappa)$ . Russian roulette decreases the average simulation time per history at the cost of increasing the variance of non-negative tallies. The value of  $\kappa$  and the selected phase space region must be adequately chosen to improve the efficiency.

#### *Particle splitting*

Splitting consists in replicating a particle in a selected phase space region  $\tau$ -times. The replicas and the original particle are identical and have a statistical weight reduced by a factor  $\tau$ . The rationale is to increase the flux of particles approaching the scoring regions. Contrarily to Russian roulette, splitting increases the simulation time and reduces the variance. Russian roulette and particle splitting are frequently used in conjunction [14].

*rotational splitting*

In geometries with cylindrical symmetry the positions of the split particles can be distributed uniformly spaced over a circumference [19, 23]. More sophisticated splitting and Russian roulette techniques have also been developed [70, 95, 67, 103, 45].

#### 1.8.1 *The Monte Carlo code PENELOPE*

Several general-purpose Monte Carlo codes for radiation transport such as EGS [87, 64], FLUKA [10], Geant4 [1], MCNP [46] and PENELOPE [99] have been publicly available for several decades. They have been used in numerous fields, including medical physics. General-purpose codes can be used in a wide energy range and for complex geometries.

In this thesis the PENELOPE code is used for the simulation of linacs and the estimation of the absorbed dose distribution in phantoms and CT geometries. PENELOPE simulates the coupled transport of electrons, photons and positrons in the energy range from 50 eV to 1 GeV. It has been extensively validated and applied to many fields, including radiotherapy [105, 39, 82, 11, 22]. A mixed simulation scheme (class II) is used for electrons and positrons, while photons are simulated in detail. The user-defined transport parameters that control the behavior of the mixed algorithm are

*transport parameters*

- C1 determines the average angular deflection between consecutive hard events.

- C2 limits the maximum average fractional energy loss between consecutive hard events.
- WCC is the energy cutoff that separates hard from soft interactions for inelastic collisions with atomic electrons.
- WCR is the energy cutoff for bremsstrahlung emission.
- DSMAX is the maximum allowed step length for charged particles.
- EABS are absorption energies at which the transport of the corresponding particle (electron, photon or positron) is terminated and the remaining energy is assumed to be locally deposited.

The software package includes a set of geometry subroutines, named `PENGEOM`, capable of handling objects limited by quadric surfaces. The geometry is coded in a text file according to a series of syntax rules. Surfaces are defined by declaring the parameters of a quadric equation and bodies are defined by declaring their limiting surfaces. Objects can be translated and rotated arbitrarily.

`PENELOPE`  
*distribution*

The `PENELOPE` code is open-source and freely distributed by the Nuclear Energy Agency Data Bank (<http://www.oecd-nea.org>) and, in North America, by the Oak Ridge National Laboratory (<http://rsicc.ornl.gov>).

`PENELOPE` requires a main program to steer the simulation and to define sources and tallies. Several main programs are distributed with the code. `PENEASY` [106] is an independent main program that implements a wide variety of configurable sources, tallies and variance-reduction techniques. Apart from taking advantage of the capabilities of the standard `PENGEOM` subroutines to simulate quadric geometries, `PENEASY` additionally includes a package named `PENVOX` which handles the transport in voxelized geometries. It is open-source and freely distributed through the web page <http://www.upc.es/inte/downloads/penEasy.htm>. For this thesis the combined system `PENELOPE/PENEASY` is used.

`PENEASY`  
*distribution*

### 1.8.2 Simulation of linear accelerators

The description of the radiation beam exiting a medical linear accelerator can be obtained by several approaches. One approach is to rely on the so-called virtual source models, which approximate the particle fluence downstream the head assembly by considering the major linac components as separate particle sources. Virtual source models can be classified in three groups: (i) those that solely use pre-calculated data from Monte Carlo simulations [78, 101, 26, 61]; (ii) hybrid models in which the planar fluences and energy distributions derived from Monte Carlo simulations are adapted to match measured dose profiles [34, 27, 131, 113], and (iii) those based on measurements [38, 35, 43]. Another approach consists in performing a

*virtual source models*

full Monte Carlo simulation of radiation transport through a detailed model of the linac head. In principle, virtual source models are less accurate than the Monte Carlo approach.

Early works on linac simulation employing the Monte Carlo method were done by Petti *et al.* [91], Mohan *et al.* [84], and Udale [120]. In the Monte Carlo approach the simulation of a linac starts by modeling the primary source of particles as a narrow beam of electrons exiting the acceleration structures and entering the head assembly. Knowledge of the characteristics of the initial particle states (*i.e.*, energy distribution, spatial distribution and angular divergence) is necessary for accurately reproducing the actual treatment beam. The simplest model assumes that a monoenergetic and collimated electron point source is located at the top surface of the target and directed downstream along the treatment head central axis. A more elaborate model assumes that the energy and particle density of the electron beam are Gaussian distributed and with an angular divergence.

A suitable configuration of the primary electron source can be found in an iterative trial-and-error procedure in which dose profiles estimated in simulations are compared to measurements in air or in a water phantom [102, 119, 2, 107]. Primary beam information provided by manufacturers might be used as a starting guess in this iterative procedure. Some Monte Carlo treatment planning systems incorporate algorithms that avoid the application of the trial-and-error procedure [2, 90, 3, 30].

To compute the dose distribution in the patient, particles must be transported through the upper structures of the linac, the patient-dependent beam modifiers and, finally, through the computerized tomography (CT) or phantom. However, this may be an inefficient method. In principle, a more efficient approach is to simulate the upper part of the linac, from the primary beam downstream to a plane situated just upstream the jaws, and to save the state of particles reaching that plane in a file. This simulation has to be performed only once per beam. The file tallied at the aforementioned plane, which is known as the phase-space file (PSF), is used as a source of particles for subsequent simulations. It is not infrequent that a second PSF is tallied at the treatment head exit and used for dose estimation in the patient.

The information on the head components used to code the transport geometry is usually provided by the linac manufacturer<sup>7</sup>. The accuracy of these data is crucial for reproducing the beams and estimating the dose [108, 107, 28, 32, 94]. In figure 5 four examples of linac head geometries coded with the tools developed in this thesis are provided. A detail of the coded MLC geometry is depicted in figure 6.

<sup>7</sup> Supply of this information is usually subjected to a non-disclosure agreement.

*linac primary  
electron source  
configuration*

*phase-space file*

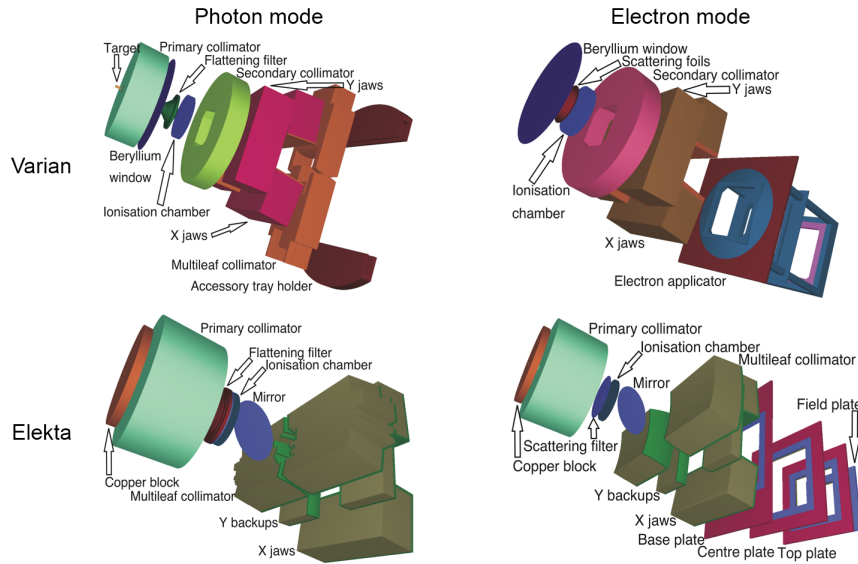


Figure 5: Examples of Varian and Elekta linear accelerator geometries generated with the tools developed in this thesis.

### 1.8.3 Phase-space files

There are differences among Monte Carlo codes with respect to the file format and the variables saved in a PSF. The variables of a particle that must be saved in a PSF are the type (*i. e.*, electron, photon or positron), the energy, the position given by the Cartesian coordinates  $x$ ,  $y$  and  $z$  and the direction cosines ( $u$ ,  $v$ ,  $w$ ) with respect to the Cartesian axes. The inclusion of a variable to relate the particle with its history is necessary to accurately estimate the statistical uncertainty of the dose, or any other quantity of interest, subsequently tallied in simulations in which a PSF is used as radiation source. Thus, the minimum space occupied by one particle in a binary file, considering data saved as single-precision floating-point variables (four bytes per datum), is about 25 bytes. This number can increase considerably depending on the information saved, the precision of the floating-point variables used and how data are packed.

To relate a particle with the history number in which it was generated an incremental shower number is included in the PSF. This number indicates how many histories (showers) were simulated between each tallied particle and the previous one in the sequence of particles in the PSF. For example, if two consecutive particles in the PSF belong to the same history, the second particle will have its incremental shower number equal to zero. The summation of all incremental shower numbers in a PSF yields the number of simulated histories. Another approach to relate each particle with its history number is to tag each particle in the PSF with a flag that indicates when a change of history has occurred, and to provide with the PSF the number of

*history number in a phase space*

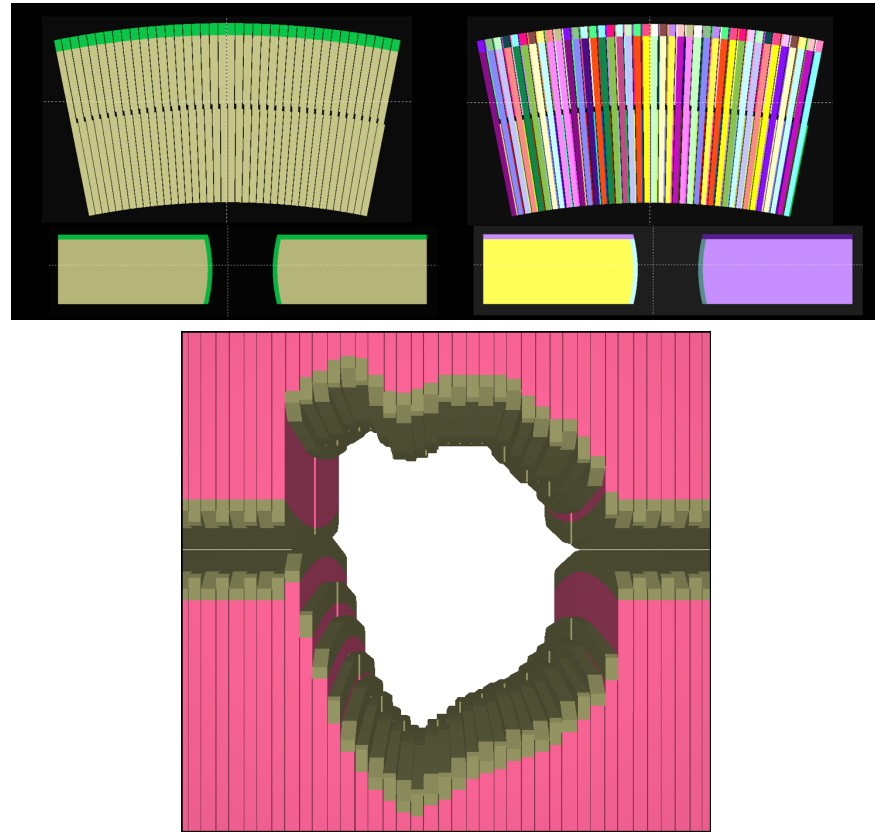


Figure 6: Above: Views of the coded geometry of the Elekta MLCi. The upper view is a transversal section of the MLC showing the fan-shaped arrangement of the leaves. The lower view is a lateral section of a leaf (in a different scale). Colors differentiate materials (at left) and geometry elements (at right). Below: a section of the coded geometry of a Varian Millennium 120-leaf MLC with a similar leaf arrangement to the MLC shown in figure 3.



simulated histories that was used for its computation. This approach yields the same accuracy than the incremental shower number in the estimation of the statistical uncertainty of quantities tallied using the PSF as radiation source, provided the whole PSF is used. If only a fraction of the PSF is used an approximation to the statistical uncertainty can be obtained by assuming that all histories have contributed with the same number of particles to the PSF.

The finite size of a PSF imposes a lower bound to the statistical uncertainty of observables tallied with it. The minimum statistical uncertainty that can be asymptotically reached using a given PSF is called the latent variance [105]. Reusing particles in a PSF is a way to approximate the uncertainty of the estimated quantities to the latent variance of the PSF. However, when recycling a PSF, it is critical to maintain the correlation between particles pertaining to the same history, otherwise uncertainties are erroneously underestimated [128]. A convenient way to reuse  $\tau$  times a PSF without losing the correlation of the particles of one history is to split  $\tau$  times the particle entering the simulation and to treat the original particle and the clones as pertaining to the same history.

The International Atomic Energy Agency (IAEA) has defined a PSF format used in their PSF database [24]. The IAEA specification includes a header file that contains general information, such as the number of histories that were simulated to produce the PSF and the total number of particles stored. The IAEA has made available a library of subroutines written in the C/C++ language to handle its PSF format.

*statistical  
uncertainty and  
phase space  
recycling*

*IAEA format for  
phase-space files*

#### 1.8.4 Dose assessment in the patient

The radiation field resulting from a linac simulation can be used to estimate the absorbed dose distribution in a patient CT or a phantom. Radiation transport in Monte Carlo algorithms relies on the knowledge of interaction cross sections for the material media traversed by particles. To perform the simulation in a CT geometry it is necessary to infer both the material composition and the mass density from the information provided by the CT scanner in terms of the so-called Hounsfield units  $H$ , defined as

$$H = 1000 \frac{\mu - \mu_{\text{water}}}{\mu_{\text{water}}}, \quad (4)$$

where  $\mu$  and  $\mu_{\text{water}}$  are the average linear attenuation coefficients of the material of interest and water, respectively, both determined for the radiation quality of the scanner.

A simple method for inferring the material composition and mass density for each voxel is the following: (i) a piece-wise linear continuous mapping between  $H$  and the density is derived from an experimental calibration curve; (ii) materials are selected from a predefined

*conversion of  
Hounsfield numbers  
to material  
composition and  
mass density*



set (e.g., those defined in ICRU Report No. 46 [54] and ICRP Publication 23 [53]) according to a mapping between them and mass density intervals [34, 37]. Mis-assignment of material composition resulting from a poor CT-to-material conversion process can have an impact on the dose. Verhaegen and Devic [126] evaluated this effect using the codes EGSnrc/DOSXYZnrc [68, 127] and found errors in the dose up to 10% for 6 and 15 MV photon beams and up to 30% for a 18 MeV electron beam. The selection of materials representing human tissues may also influence dose accuracy.

From the 3D dose distribution matrix relevant quantities can be extracted. The most common quantities reported by dose calculation systems include 1D dose profiles, 2D dose maps, isodose lines and dose-volume histograms (DVHs). Dose distributions (1D, 2D or 3D) can be compared by means of relative differences and the gamma index [77].

The increasing clinical application of the Monte Carlo method has raised the question of whether the dose-to-medium  $D_m$ , which is the dose estimated by Monte Carlo codes, or the dose-to-water  $D_w$ , the one historically calculated by treatment planning systems, should be specified [4]. At present, there is no consensus [76]. Relative differences between  $D_m$  and  $D_w$  are given by the unrestricted mass collision stopping power ratio of water to medium  $(\bar{S}/\rho)_m^w$ . They are in the range of 1%-2%, hence considered not clinically relevant for tissues with media composition not differing much from water, with the exception of bone, where differences can be as high as 15% [112, 36, 41]. In order to make comparisons of clinical cases, a conversion from  $D_m$  to  $D_w$  may be required.  $D_w$  can be estimated directly in the Monte Carlo simulation by multiplying the deposited energy by the ratio of the restricted mass collision stopping power of water to the medium  $(\bar{L}/\rho)_m^w$ , determined for the energy of the particle [129]. Post-processing methods to convert  $D_m$  to  $D_w$  [112, 41] are more practical and can be applied to photon beams. For photon beams  $(\bar{S}/\rho)_m^w$  is nearly constant for materials present in the human body (with the exception of air), with a maximum deviation of 1.1% for cortical bone.

Statistical uncertainties are inherent to dose distributions estimated with the Monte Carlo method. The impact of the dose uncertainty on the outcome of the radiation therapy has been addressed by several authors [33, 71, 21, 79]. Due to statistical uncertainties isodose lines exhibit a jagged appearance that may confound visual evaluation of the dose distribution. Integral dose quantities such as DVHs are less affected by the uncertainty. However, steep DVHs, such as those of PTVs, are significantly smoothed (as if resulting from a convolution with a Gaussian curve) when calculated from dose distributions with a large uncertainty. This effect is less pronounced in DVHs of critical structures due to their shallower slopes [71].

*dose-to-medium  
versus dose-to-water*

*absorbed dose  
distribution  
uncertainty*

It has been suggested that a relative statistical uncertainty of 2% of the maximum dose ( $D_{\max}$ ) is acceptable in evaluating a Monte Carlo based treatment plan [71]. However, the uncertainty of a single voxel is a poor measure for the uncertainty of a treatment plan. When analyzing dose limits to critical organs, for instance, the level of acceptable uncertainty is relative to the biological effect produced by the dose on a particular organ. The biological effect on serial organs, such as the spinal cord, is exclusively dependent on maximum dose values, therefore demanding a low statistical uncertainty for their estimation. In contradistinction, biological effects on parallel organs, such as the lungs, depend on mean doses. For estimating the mean statistical uncertainty  $\bar{\sigma}_V$  of a region  $V$  the summation in quadrature of the relative uncertainties of all voxels in  $V$  is done,

$$\bar{\sigma}_V = \sqrt{\frac{1}{K} \sum_{k=1}^K \sigma_k^2}, \quad (5)$$

where  $K$  is the number of voxels in  $V$  and  $\sigma_k$  is the relative statistical uncertainty of the dose in voxel  $k$ . Rogers and Mohan [96] have proposed the mean uncertainty of a volume determined by all voxels accumulating a dose greater than  $D_{\max}/2$  as a convenient indicator of the overall uncertainty of a dose distribution.

General-purpose Monte Carlo codes allow the simulation of a wide range of energy and materials, which makes them computationally slow. There exists, however, another type of Monte Carlo codes which introduce simplifications in the transport routines, and possibly in the range of considered materials, in order to speed up the computation. These codes are generally called fast Monte Carlo and have been extensively used for the simulation of CT geometries. Some of these codes are VMC [69], XVMC [42], VMC++ [66, 65], DPM [104] and PENFAST [20, 49].

*fast Monte Carlo  
codes*

## 1.9 OUTLINE OF THE THESIS

The objectives listed in section 1.2 concretize into a software system we have named PRIMO. This thesis, which describes its development and structure, is organized in nine chapters. Chapters from 2 to 8 correspond to the scientific papers listed on page ix.

Chapter 1 is the current introduction.

Chapter 2 describes the state-of-the-art in the field of Monte Carlo systems for treatment planning and dose verification. All systems that are self-contained and currently available, including PRIMO, are compared in terms of features and simulation speed benchmarks.

PRIMO is introduced in chapter 3. The article gives a general overview of the program and describes its structure and work flow. Three important aspects that distinguish PRIMO are remarked in the text:

ease of use, accurate transport physics and improved efficiency by the use of several variance-reduction techniques. These aspects are illustrated through an example of dose calculation in a 3DCRT prostate treatment and also by a summary of simulation times obtained for a variety of linac models, beams and radiation fields.

One of the objectives of the thesis is to speed up Monte Carlo linac simulation by means of new optimization algorithms. This objective materializes in chapters 4 and 5, which are dedicated to VRT specifically developed for PRIMO and their optimization.

Chapter 4 presents the splitting-roulette VRT. This method combines several VRT to boost simulation efficiency of the upper part of the linac. A splitting technique named selective splitting is also introduced. The technique allows to sample a different direction of flight for each replica of a split bremsstrahlung photon.

Chapter 5 describes a method to optimize the simulation efficiency when radiative interactions are forced in the linac target and particle splitting is used in the patient geometry. A functional relation between the simulation efficiency and the forcing and splitting factors is theoretically derived.

Chapter 6 is a study of the effect that the PENELOPE transport parameters have on the bremsstrahlung angular distribution in the target and the subsequent dose in the phantom or patient. The knowledge derived from this study is used in PRIMO for the selection of the transport parameters that restricts the charged particle step length to produce accurate results without compromising the simulation speed.

The objective of applying the newly developed tool in medical physics research is exemplified in chapter 7. Therein, a comparative study of measured dose profiles with those obtained from PRIMO using Varian's phase-space files for their TrueBeam linac was conducted. Thus, this chapter serves to put the objective of providing a practical analysis tool to the test.

Results from the article appearing in the previous chapter motivated the derivation of an experimentally-based geometrical model of the TrueBeam linac, which is described in chapter 8. This further illustrates the use of PRIMO for a research application. The implementation of this model in PRIMO extends its capabilities to this novel linac and contributes to the fulfillment of the first objective related to linac geometries.

The conclusions are presented in chapter 9. The thesis is further complemented with the following appendices: Appendix A contains distribution information about PRIMO, including current scientific collaborations.

Appendix B describes the modifications done for this thesis to a previously existing code for automatic generation of linac geometries. These modifications were necessary to include Elekta linacs. They

were conducted using the PENLINAC code described in the article included in the appendix. Since this article was published prior to enrollment into the Ph.D. program it does not belong to this thesis, and it is included here only for completeness.

Appendix C contains an overview of the programming techniques and algorithms employed in coding the graphical layer for the automation of the simulation system (GLASS), which is the highest layer on the software structure of PRIMO. The GLASS includes the graphical user interface (GUI) as well as all the functions for the configuration of simulations, data import/export, visualization, processing and data flow control.

Appendix D reproduces the PRIMO User's Manual and the Quick Start Guide. The manual includes examples that illustrate the use of the system in a variety of situations.



# 2

## MONTE CARLO SYSTEMS USED FOR TREATMENT PLANNING AND DOSE VERIFICATION OF LINAC EXTERNAL BEAM RADIOTHERAPY

---

# Monte Carlo systems used for treatment planning and dose verification of linac external beam radiotherapy

Lorenzo Brualla\*

*NCTeam, Strahlenklinik, Universitätsklinikum Essen,  
Hufelandstraße 55, D-45122 Essen, Germany.*

Miguel Rodriguez†

*Institut de Tècniques Energètiques,  
Universitat Politècnica de Catalunya,  
Diagonal 647, E-08028 Barcelona, Spain.*

Antonio M. Lallena‡

*Departamento de Física Atómica, Molecular y Nuclear,  
Universidad de Granada, E-18071 Granada, Spain.*

## Abstract

General-purpose radiation transport Monte Carlo codes have been used for the estimation of the absorbed dose distribution in external beam radiotherapy patients since several decades. Results obtained with these codes are usually more accurate than those provided by treatment planning systems based on non-stochastic methods. Absorbed dose computations based on general-purpose Monte Carlo codes have been traditionally used only for research owing to the difficulties associated to setting up a simulation and the long computation time required. In an effort to extend the application of Monte Carlo codes to the routine clinical practice researchers and private companies have developed treatment planning and dose verification systems that are partly or fully based on fast Monte Carlo algorithms.

This review presents a comprehensive list of the currently existing Monte Carlo systems that can be used to calculate a treatment plan or to verify it. Particular attention is given to those systems that are distributed, either freely or commercially, and that do not require programming tasks from the end-user. These systems are compared in terms of features and simulation time required to compute a set of benchmarks.

Keywords: linac, radiation transport, Monte Carlo, treatment planning, dose verification

---

\*Electronic address: [lorenzo.brualia@uni-due.de](mailto:lorenzo.brualia@uni-due.de)

†Electronic address: [miguel.lazaro.rodriguez@upc.edu](mailto:miguel.lazaro.rodriguez@upc.edu)

‡Electronic address: [lallena@ugr.es](mailto:lallena@ugr.es)



## I. INTRODUCTION

It is known that the Monte Carlo method applied to the simulation of external beam radiotherapy is capable of computing accurate absorbed dose distributions [1]. In general, the accuracy obtained from a Monte Carlo simulation in the dosimetry of external radiotherapy plans of photon or electron beams is better than that yielded by treatment planning systems based on non-stochastic algorithms. Non-negligible differences between the results obtained with both types of algorithms have been reported [2–5]. General-purpose Monte Carlo codes excel in the presence of small radiation fields and regions with large mass density gradients [6–17]. There are, however, two severe drawbacks associated with these codes. First, Monte Carlo simulation of radiotherapy problems usually entails an exceedingly long computation time which is not acceptable in the routine clinical practice. To palliate this problem several approaches have been followed: simplification of the physics and geometry models of the general-purpose Monte Carlo radiation transport codes [18–24], massive parallel computing [25–27] and the implementation of variance-reduction techniques [28]. The second drawback is the effort required to set up a Monte Carlo simulation from scratch.

In the last two decades, both, researchers and private companies have endeavored in producing Monte Carlo codes that can become an alternative to non-stochastic treatment planning systems [3, 18, 19, 29–33]. The scope has been to maintain the virtues of general-purpose Monte Carlo codes while reducing simulation time and facilitating its use. Owing to the interest in Monte Carlo treatment planning much literature has been published and excellent review articles have appeared. In particular, those of Andreo [34], Verhaegen and Seuntjens [35], Rogers [36], Reynaert *et al.* [1], Spezi *et al.* [37], the Task Group article of Chetty *et al.* [9], and the book edited by Seco and Verhaegen [38] constitute fine documents for learning about the Monte Carlo method applied to radiotherapy. These are valuable publications for researchers willing to start working in the field of Monte Carlo treatment planning.

The present review is an update on the Monte Carlo dose planning situation. Since the publication of the aforementioned reviews new Monte Carlo codes have appeared and others which were planned have been abandoned. To date, this review offers the most comprehensive and updated list of Monte Carlo dose planning systems for external photon and electron beam radiotherapy. Emphasis is given on key aspects that can help researchers,

medical physicists and clinicians on deciding which system is more adequate for their needs. Particular attention is paid to those systems that are distributed, either freely or commercially, and that do not require programming tasks from the end-user. These systems are extensively compared in terms of features and simulation time required for computing a set of benchmarks.

## II. CLASSIFICATION CRITERIA FOR MONTE CARLO TREATMENT PLANNING AND DOSE VERIFICATION SYSTEMS

To the extent of this review we call *Monte Carlo treatment planning system* those programs allowing to fully elaborate the treatment plan, estimate the dose distribution in the patient—using the Monte Carlo method—and render any parameter (*e. g.*, the monitor units) necessary for the actual treatment of the patient in a linear accelerator. Here, we do not make distinction between inverse or direct planning techniques nor between 3D CRT and IMRT treatment modalities. Conversely, we call *Monte Carlo dose verification system* those programs aimed at the verification of the dose distribution and/or the monitor units calculated with a treatment planning system. Monte Carlo dose verification systems use as input the configuration of the radiation fields, and other data of the plan, obtained by a treatment planning system. They perform the estimation of the dose distribution using a Monte Carlo based algorithm that is independent from the one implemented in the treatment planning system.

A further classification of the codes reviewed is established according to how the simulation of the linac is approached, the Monte Carlo algorithms for radiation transport employed and how those algorithms are linked to the part of the code dedicated to configure the plan and to analyze the results. In more detail, the criteria for this classification are the following:

1. The simulation of a linac can be divided in two parts: the simulation of the linac components located upstream the movable collimators (upper part), and the simulation of the movable collimators themselves and all other linac components downstream them (lower part). The upper part only depends on the irradiation mode (photon or electron) and the nominal energy chosen, while the lower part changes with every radiation field. For, it is said that the upper part is patient-independent, while the lower part is patient-dependent. As a result, the simulation of a treatment can be

divided into three segments: (i) linac upper part; (ii) linac lower part, and (iii) the computerized tomography (CT) representing the patient. We call *full Monte Carlo systems* those that use Monte Carlo codes for the simulation of the three parts. There are, however, systems that use a virtual source model to represent the beam at the exit of the upper or lower part of the linac and perform the Monte Carlo simulation of radiation transport only through the CT representing the patient. We call them *virtual source model Monte Carlo systems*.

2. Three types of Monte Carlo algorithms are used in radiotherapy to simulate the radiation transport. *General-purpose Monte Carlo codes* simulate radiation transport in a wide energy range, from a few hundreds of eV up to GeV, in geometrical structures of arbitrary complexity, involving materials that can be composed from most of the elements of the periodic table. Examples of these codes are EGSnrc [39], PENLOPE [40, 41], Geant4 [42] and MCNP [43, 44]. The review of Verhaegen and Seuntjens [35] offers a table in which these Monte Carlo codes are compared based on the implemented physical models. General-purpose Monte Carlo codes usually produce the most accurate results [45–48] and offer the largest flexibility, at the expense of slower simulation speed. All full Monte Carlo systems developed so far use general-purpose Monte Carlo codes. *Fast Monte Carlo codes* [49] are those that impose certain limitations on the generality of general-purpose codes in order to attain higher simulation speed. Commonly used simplifications are to limit the physical models to the range of energies found in radiotherapy problems and to restrict the material composition supported to the low  $Z$  elements found in the human body. Fast Monte Carlo codes might also reduce the range of geometries supported, *e. g.*, to include only those necessary for dose scoring. Examples of these codes are VMC [18], DPM [19], XVMC [20], PENFAST [23, 24] and VMC++ [50, 51]. *Pre-calculated Monte Carlo codes* use Monte Carlo pre-calculated data within codes that are either analytical or that use probability distributions. Pre-calculated Monte Carlo codes do not sample interaction cross sections during the simulation but depend on look-up probability tables that have been previously calculated for a given set of conditions. They tend to be faster than all previous types at the expense of introducing further simplifications and limitations in their applicability. Examples of these codes are Macro Monte Carlo (MMC) [52]

and Super-Monte Carlo (SMC) [53].

3. There exist programs that only provide a graphical-user-interface with a set of tools for radiotherapy but that do not include a Monte Carlo algorithm for radiation transport. Instead they provide a form of linking to a specific third party Monte Carlo code in order to let the user perform the simulation of the linac and the estimation of the dose. Usually, the user has to code the geometrical description of the linac. We classify those programs as *not self-contained* to differentiate them from *self-contained* programs that do not require any coding or linking tasks from the end-user.

### III. MONTE CARLO TREATMENT PLANNING AND DOSE VERIFICATION SYSTEMS REVIEWED

The currently existing Monte Carlo treatment planning and Monte Carlo dose verification systems are reviewed in this section. They appear in alphabetical order. Those systems that are self-contained and distributed, either commercially or freely, are treated with more detail. Table I shows a comparison of the reviewed systems in terms of the classification criteria used in this review.

#### A. CARMEN

The system CARMEN has been developed at the Universidad de Sevilla [54–57]. It is based on the codes BEAMnrc [58] for the simulation of the whole linac and a modified version of DOSXYZnrc [69] for tallying the dose distribution in the CT or water phantom. Both, BEAMnrc and DOSXYZnrc, rely on EGSnrc as the general-purpose Monte Carlo code [39]. It bases the simulation of the dose on the implementation of beamlets. CARMEN contains the geometrical description of most Varian, Elekta and Siemens linacs. The graphical user interface is coded in Matlab (MathWorks, Massachusetts, USA) while for the remaining parts C++ is used. Thanks to the employment of Matlab the graphical computations on CT images can be done in parallel using graphics processing units (GPU). CARMEN can be used as a verification tool for evaluating plans proposed by other systems or as planning tool capable of proposing plans. The system has a complete set of graphical and numerical tools

TABLE I: Comparison of the Monte Carlo systems according to the classification criteria used in this review. The following abbreviations are used in the table: treatment planning (TP) and dose verification (DV) systems; full Monte Carlo (full) and virtual source model Monte Carlo (vsm) systems; general-purpose Monte Carlo (gp), fast Monte Carlo (fast) and pre-calculated Monte Carlo (pc) codes.

System	Use	Simulation			Self-contained	Distributed
		Type	Linac (MC code)	Patient (MC code)		
CARMEN	TP	full	gp (EGSnrc)	gp (EGSnrc)	no	no
CERR	DV	–	–	fast (VMC++)	no	free
Corvus	TP	vsm	–	fast (PEREGRINE)	yes	no
Eclipse	TP	vsm	–	pc (MMC)	yes	pay
eIMRT	DV	full	gp (EGSnrc)	gp (EGSnrc)	yes	free
iPlan	TP	vsm	–	fast (XVMC)	yes	pay
ISOgray	TP	full	gp (PENELOPE)	fast (PENFAST)	yes	no
MCDOSE	TP	vsm	–	gp (EGS4)	no	no
MCDE	TP	full	gp (EGSnrc)	gp (EGSnrc)	no	no
MCVS	DV	full	gp (EGSnrc)	gp (EGSnrc)	no	no
MMCTP	DV	full	gp (EGSnrc)	gp (EGSnrc)	no	free
Monaco	TP	vsm	–	fast (XVMC)	yes	pay
Oncentra	TP	vsm	–	fast (VMC++)	yes	pay
Pinnacle	TP	vsm	–	fast (DPM)	yes	no
PLanUNC	DV	full	gp (EGSnrc)	gp (EGSnrc)	no	free
PRIMO	DV	full	gp (PENELOPE)	gp (PENELOPE)	yes	free
RTGrid	TP	full	gp (EGSnrc)	gp (EGSnrc)	no	no
SMCP	TP	vsm	–	gp (EGSnrc)/fast (VMC++)	no	no
VIMC	DV	full	gp (EGSnrc)	gp (EGSnrc)/fast (VMC++)	no	no
XiO	TP	vsm	–	fast (XVMC)	yes	pay

for evaluating the simulated results. CARMEN is not distributed. It may be installed only in the computers of those groups that have established a collaboration with the authors.

## **B. CERR**

The Computational Environment for Radiation Therapy (CERR) [59] is described in its hosting web site <http://www.cerr.info> as a set of tools for the analysis of research results in radiation therapy. CERR is written in Matlab. The system can import CT scans, delineate structures and visualize CT slices, dose distributions and dose-volume histograms. It can also import PET, SPECT and MRI images. It allows the visualization of 3D images of the scans superimposed to the computed dose distributions and the contoured structures. The system includes tools for analyzing the dose distribution matrix. The last update of the user guide is dated on November 2003 and the last update of the code was on March 2007. The code can be downloaded from the site referred above, including a 2002 version of VMC++ [21] that can be used for fast Monte Carlo simulation. Mukumoto and co-workers [60] linked CERR with the general-purpose Monte Carlo code EGSnrc. This work is the basis of the dose verification system MCVS (see below). There is an ongoing work which pursues to use the fast Monte Carlo code DPM [19] as a computation engine of CERR.

## **C. Corvus**

This treatment planning system marketed by Nomos Corporation intended to include the Monte Carlo code PEREGRINE [61]. PEREGRINE was developed at the Lawrence Livermore National Laboratory. It uses the correlated histogram source model [62]. This is a multi-source virtual model that includes the contribution to the beam of particles produced on the linac components located upstream the jaws. The source is completely derived from a precalculated phase-space file (PSF). Thus, PEREGRINE is classified as a virtual source model Monte Carlo system. The particles are transported through the movable components of the linac as well as through the patient geometry. PEREGRINE employs a different transport method for each region. Nevertheless, in both regions photons are transported in a detailed manner and charged particles are transported by employing a class II condensed history method which is a variation of the algorithm implemented in EGS4 [63] and includes

the random-hinge approach developed for the PENELOPE Monte Carlo code [41]. One particular characteristic of PEREGRINE is that the dose is scored in a grid that is independent from the CT-derived voxel grid. The scoring regions are spheres of variable radius. Several variance-reduction techniques are employed in the transport including source particle reuse, range rejection, Russian roulette and splitting [28]. PEREGRINE is implemented in a multiprocessor architecture and operates in UNIX. On year 2000 Nomos obtained the Food and Drug Administration (FDA) clearance for marketing a version of Corvus with the Monte Carlo algorithm. The Monte Carlo module of Corvus is not distributed.

#### D. Eclipse

This treatment planning system from Varian Medical Systems includes a Monte Carlo module, called electron Monte Carlo (eMC), for the calculation of dose distributions due to high-energy electron beams. The Monte Carlo module consists of the Initial Phase Space (IPS) model, in which the dynamical variables of the electrons and photons emerging from the linac head are included, and the Macro Monte Carlo (MMC) transport algorithm [52], which performs the dose calculations.

The IPS model is an implementation of the multiple source model developed by Janssen *et al.* [64]. It involves several particle sources that are based on data calculated for each linac model and tuned using measured data. The sources are: i) a main diverging beam, which describes the particles (electrons and photons) emerging from the scattering foil; ii) an edge electron source, which takes into account the electrons scattered at the upper rim of the inner side of the cutout of the applicator or insert; iii) a transmission photon source, which includes those main photons that go through the insert material without suffering interactions; iv) a second diverging beam, which is a virtual point source of both electrons and photons. IPS only supports Varian linacs (see table II). Fix and co-workers have recently produced an improved version of MMC which allows the simulation of electron beams of Elekta and Siemens linacs [65].

MMC performs first a conventional Monte Carlo simulation of electrons with energies of clinical interest (from 0.2 to 25 MeV) impinging on spheres of various radii (from 0.5 to 3 mm) filled with given materials (air, lung, water, lucite, solid bone). Probability distribution functions for the different particles emerging from the spheres are calculated. Specifically,

in these distributions the position, direction and energy of the emitted primary electrons (those with the higher energy if more than one electron exits the sphere) are scored for each combination of material, sphere radius and electron incident energy, while for the secondary electrons and photons only their average energy per primary electron is stored as a function of the energy of the initial electrons.

In a second step, the particle beams generated in the IPS sources are transported on the CT representing the patient, using the pre-calculated probability distribution functions. This part of the simulation is done by means of macroscopic steps according to the defined material spheres. To do that the absorber CT volume is pre-processed to establish the adequate sphere sizes and mean densities, at each position of the CT, according to its specific material distribution. The output dose distributions can be smoothed to reduce the statistical noise of the Monte Carlo results. Both Gaussian and median dose smoothing are supported. Statistical uncertainties are evaluated with a batch method in which simulations are divided into at least 10 batches of  $10^4$  particles each. The average statistical uncertainty in the dose maximum region, the spatial resolution for the dose calculation, the accuracy limit, the random seeds, the maximum number of histories and the smoothing method and level can be selected by the user.

The result of this approach is a faster simulation at the expense of a reduced accuracy of the results. In general, a 3% accuracy of the dose maximum is expected from eMC. However, it is known to exceed this value in different situations such as, *e. g.*, in outer regions of large applicators, extended source-to-surface distances, heterogeneities or in the presence of small field sizes [12].

Beam configuration requires measurements done for each combination of applicator size and available energy. Each of these measurements involves a depth dose curve in water and a lateral profile in air, as well as an absolute dose measurement at the calibration point. The measured data are loaded afterwards in the Eclipse Beam Configuration program distributed by Varian which performs the fitting of the virtual source model parameters.

#### **E. eIMRT**

eIMRT [66, 67] simulates the radiation transport through the whole linac head and the CT of a patient. The program relies on BEAMnrc [58, 68] for the simulation of the linac



and DOSXYZnrc [68, 69] for the simulation of dose distributions in CTs, both based on the general-purpose Monte Carlo code EGSnrc [39]. eIMRT cannot handle PSFs and, therefore, the simulation of each radiation field must always include the entire linac head. The implemented linacs are the Siemens Primus, Siemens Onco, and the Varian Clinac C-series (18, 21, 23, iX, etc.) (see table II). The code only simulates photon beams from these linacs. eIMRT can compute the absorbed dose distribution from 3DCRT and IMRT treatments. Dynamic wedges can also be simulated. A necessary input for the code is a treatment plan previously configured with a treatment planning system. eIMRT recalculates the provided plan using EGSnrc and then compares the obtained absorbed dose distribution with that one provided by the treatment planning system. It is a full general-purpose Monte Carlo dose verification self-contained system.

eIMRT is based on the cloud Software as a Service (SaaS) paradigm. The end-user has a client program written in Java that allows to enter the desired configuration of the linac, the CT of the patient and other necessary information such as the structures (using DICOM-RT STRUCT) and the treatment plan (DICOM-RT PLAN). Optionally, calculated doses from another treatment planning system can be also uploaded using DICOM-RT DOSE files. This client connects to the Supercomputing Centre of Galicia (CESGA, Spain) and sends the anonymized data over the Internet in order to launch the computation. Once the simulation has finished it alerts the user and it gives the possibility to analyze the results. This service can generate 2D and 3D gamma index [70] maps to compare the dose distributions. The code decides the number of primary histories to be simulated to reach an adequate uncertainty for the treatment. The user has no access to the statistical uncertainty information. The dose is given in terms of dose to medium. Finally, the dose distribution can be downloaded in DICOM-RT DOSE format or a proprietary one.

When importing the CT of a patient into eIMRT the code converts Hounsfield units into medium composition and mass density for each voxel. There is a database of 45 materials and a given CT can contain up to 7 materials. The database contains all clinically relevant materials such as bone, soft tissue, lung, titanium, etc. The dose computation matrix of eIMRT uses slices of CTs with a voxel size of  $3 \times 3 \text{ mm}^2$ . Therefore, independently of the number of voxels per slice, or the voxel size of the original CT, all tomographies are rescaled to match this voxel size.

To start using eIMRT with a given linac and nominal energy the user must provide dose

profiles in water for the  $20 \times 20$ ,  $10 \times 10$  and  $2 \times 2$  cm<sup>2</sup> reference fields. A notable feature of this system is an algorithm that automatically finds the adequate initial beam parameters for a given nominal energy of a linac based on the uploaded experimental data [71]. The beam parameters used for defining the primary electron source are the primary energy and the full width at half maximum (FWHM) of the focal spot size. The user has no access to these parameters and they cannot be modified.

For the simulation of the linac head eIMRT implements the variance-reduction techniques of directional bremsstrahlung splitting [72] and range rejection [28]. The optimal parameters used for these techniques are selected by the system and the user cannot modify them.

Currently, eIMRT is freely available to anyone working at a hospital or research institution. Interested users can request access to the server by filling out the application form located at <http://eimrt.cesga.es/signup.html>. In order to guarantee that simulations are computed in a given scheduled time, the authors have developed load-balancing algorithms. eIMRT uses computational resources located at CESGA as well as external facilities. The rationale of the code is to provide a fully Monte Carlo based treatment plan verification system that automatically decides most of the parameters required for running a Monte Carlo simulation, thus making it simple to use.

## F. iPlan

The treatment planning system of Brainlab, called iPlan RT Dose [73], permits to calculate dose distributions either with a pencil beam convolution algorithm or a Monte Carlo simulation. iPlan is a fast Monte Carlo virtual source self-contained treatment planning system.

The Monte Carlo module of iPlan consists of three components. The first one relies on a virtual source model of the upper part of the linac head (above the jaws and the multileaf collimator) which is modeled in terms of various particle sources that are able to reproduce the PSFs corresponding to the actual geometry. This virtual source model is based on the virtual energy fluence model of Fippel *et al.* [74]. It includes two or three photon sources for describing the photons produced in the target by bremsstrahlung and those suffering Compton scattering in the primary collimator and the flattening filter. There is source for modeling the beam electron contamination at the patient entrance. Sources

are conveniently situated according to each particular linac. The parameters characterizing the spatial distribution of the photon sources are the Gaussian width, the relative weight, and the horn corrections that take care of the differences between ideal flat profiles and actual beam profiles. These parameters are determined by fitting the theoretical fluence distribution to experimental profiles measured in air. The corresponding energy spectra, as well as, the spatial and energetic characteristics of the contaminant electron source are determined from a depth dose curve measured in water for the reference field size. Most of the Elekta, Varian and Siemens linacs are supported (see table II).

In the second component of the Brainlab Monte Carlo algorithm, the simulation of the particle transport through the collimation system, it is first assumed that jaws fully block the beam. Two different approaches, that can be selected by the user, are implemented for the simulation of the multileaf collimators (MLC). If the ‘accuracy optimized’ option is selected, the collimator is modeled on the base of the design of the actual leaves and the leaf leakage is simulated. If the ‘speed optimized’ option is chosen, the air gaps between neighboring leaves and the tongue/groove design are ignored. The transport algorithm is based on the work of Fippel [75] in which photoelectric absorption, Compton scattering and pair production are considered in case of photons, while electrons (primary and secondary) are simulated within the continuous slowing down approximation. The geometry definition is simplified by using plane and cylindrical surfaces to define the different material regions. The photon cross sections and electron stopping powers and ranges for the simulation of tungsten and air are pre-calculated using the XCOM [76] and ESTAR [77] codes, respectively. MLCs from Elekta (beam modulator, MLCi), MHI MLC 60, Novalis/Brainlab m3 (for Varian, Elekta, Siemens Mevatron, Primus, Oncor 82 and 160 and Artiste), Varian (HD120, 52, 80 and 120) and Siemens (3-D 58 and 82 Leaves, 160 and ModuLeaf) are supported.

The particles surviving the collimation system are transferred to the patient dose calculation engine, the third component of the Monte Carlo module, where the absorbed dose distribution in the CT of the patient is calculated using the X-ray Voxel Monte Carlo (XVMC) code, developed by Kawrakow and Fippel [18, 20, 22]. XVMC is a fast Monte Carlo algorithm that introduces simplifications in the physical transport and geometry models with respect to general-purpose algorithms. Electrons are transported within a condensed history algorithm, and both delta electrons and bremsstrahlung photons are taken into account. Photoelectric absorption, Compton interactions and pair production are considered. The code implements

several variance-reduction techniques, such as electron history repetition [22, 38], multiple photon transport [38, 50, 51] and Russian roulette. Inaccurate doses may be obtained in case implants or, in general, non-human tissues are present in the patient geometry.

In addition to the degree of accuracy with which the MLC is simulated, the user can select three characteristics of the last step of the simulation:

- The spatial resolution of the dose grid. It can be chosen between 2 and 10 mm.
- The mean statistical variance desired for the final dose results. Actually, this value is used to estimate the number of histories that are required to reach dose values with such a variance at the maximum dose of the corresponding beam.
- The dose type result, that is either dose to the medium or actual tissue (which is the default option) or dose to water.

Configuration of the pencil beam algorithm is a prerequisite for using the two Monte Carlo dose calculation engines that are almost independent and whose results can be cross checked by the user. The series of measurements needed for the beam configuration include measurements in air and in water. Specifically, in air, one must provide the  $z$  profile in the linac central axis from  $z \sim 85$  to  $\sim 115$  cm, the  $x$  and  $y$  transverse profiles at  $z = 85, 100$  and  $115$  cm and the output factor at  $z = 100$  cm for all fields and photon energies (these  $z$  values are distances with respect to the nominal focus of the photon source). In water, the depth dose and  $x$  and  $y$  transverse profiles at the depth of the maximum dose and at depths of 10 and 20 cm, for source-to-surface distances of 90 and 100 cm. In this last case, also the output factors at a depth of 10 cm are needed. Both square and rectangular radiation fields with lateral dimensions ranging between 0.8 cm and 40 cm are required depending on the specific MLC of the simulated linac. The whole beam configuration procedure is carried out by the company.

### **G. ISOgray**

DOSIsoft (France) developed a Monte Carlo module that simulates both electron and photon beams. This module was coded to work within the ISOgray treatment planning system [24, 78, 79]. The Monte Carlo module relies on the general-purpose Monte Carlo

code PENELOPE [41] for the simulation of the whole linac. The module contains a library of PSFs, generated with PENELOPE, of the upper part of the linac. The lower part is simulated with geometries generated by AUTOLINAC [78–80], a code that automatically generates geometry files for most Varian linacs to be simulated with PENELOPE. The simulation of the absorbed dose distribution in the CT of the patient is performed with the fast Monte Carlo code PENFAST [23, 24], a class I algorithm in the classification of Berger [81]. A beta version of the Monte Carlo module was distributed to reference centers for pre-clinical validation. A release version is not being distributed.

## H. MCDE

The Monte Carlo Dose Engine (MCDE) was developed at Ghent University [82]. It relies on EGSnrc/BEAMnrc [58] and EGSnrc/DOSXYZnrc [69] for the simulation of the patient-dependent part of the linac and the patient, respectively. DOSXYZnrc is reprogrammed and incorporated into BEAMnrc as one additional component of the linac, so that, the linac and the patient can both be configured in the BEAMnrc’s input file. The source of particles is a PSF obtained from a simulation with BEAMnrc and tallied at a plane situated below the mirror of the linac. The dose is scored in a grid of spherical voxels. MCDE can be interfaced to the GRATIS planning system and its Virtual Simulator<sup>TM</sup> [83] for the configuration of the treatment plan and the analysis of dose distributions. MCDE can read the DICOM-RT and CT files generated with GRATIS and can convert dose files resulting from the simulation to DICOM format [84]. MCDE is not distributed.

## I. MCDOSE

The MCDOSE [85] code developed at Stanford University and the Fox Chase Cancer Center uses EGS4 [63] for the simulation of the patient-dependent part of the linac and the CT. For the patient-independent part of the linac a source model is employed. The code is capable of simulating photon and electron beams. Dynamic wedges and IMRT plans can be simulated. MCDOSE relies on external programs, also coded in-house, for the analysis and plot of results, such as, absorbed dose distributions and dose-volume histograms. An external program is used for obtaining medium composition and material density from the

Hounsfield units of the input CT. MCDOSE can be linked to the treatment planning system FOCUS (Computerized Medical Systems, Inc. St. Louis, USA) for reading the patient CT and the associated structures. The code is not distributed.

## **J. MCVS**

The Monte Carlo Verification System (MCVS) [60, 86] has been developed at the University of Osaka. The code is based on the general-purpose Monte Carlo code EGSnrc [39]. It uses BEAMnrc [58, 68] to simulate the linac head and DOSXYZnrc [68, 69] to compute the dose in the CT of the patient. For the graphical user interface the code relies on CERR [59]. The code is not distributed.

## **K. MMCTP**

McGill Monte Carlo Treatment Planning (MMCTP) [87] is a multiplatform interface for Monte Carlo treatment planning. The system can run in Mac OS X, Windows and Linux. The system uses EGSnrc [39] as Monte Carlo engine and BEAMnrc [58, 68]. In order to run a simulation the user must obtain and install those programs and code the linac geometry. MMCTP falls is a not self-contained Monte Carlo dose verification system. MMCTP can import the DICOM files of the CT and the treatment plan parameters from a treatment planning system. The linac must be simulated using BEAMnrc/EGSnrc. It can import delineated structures and dose distributions calculated by a treatment planning system. It can generate dose-volume histograms. The source code is distributed by personal request to the authors. Users must compile their own executable files using a multiplatform development environment named Xojo which must be downloaded and installed independently.

## **L. Monaco**

Monaco is a fast Monte Carlo virtual source self-contained treatment planning system. It is distributed by Elekta and incorporates a pencil beam convolution model. The pencil beam model is used for optimization in inverse treatment planning, because of its faster speed, while the Monte Carlo code may be used to compute the final absorbed dose distribution

with higher accuracy. Major linac models are supported, with the exception of TrueBeam (see table II).

As in the case of iPlan, the Monaco Monte Carlo engine is formed by three components: a virtual source model, which is used to describe the radiation transport through the linac head components, a set of transmission filters that model the primary collimators, the jaws and the MLC, and, finally, the dose calculation algorithm, which is based on XVMC [18, 20, 22].

The virtual source model currently available in Monaco is VSM1.6 which is based on the work by Sikora and Alber [88–90]. There are still some Monaco systems working with the previous version, VSM1.5. In VSM1.6 three sources, primary and secondary photon, and electron contamination, are defined. Their geometrical characteristics, as well as their energy spectra, are chosen based on specific measurements performed in water. The primary source is situated at the target and the secondary one at the base of the primary collimator. Both sources have Gaussian fluences with energy dependent sizes. Fluences can be fine tuned by means of horn correction functions (tabulated or fitted) that take care of the effects of flattening filters having shapes more elaborated than the simple conical shape. The electron source is located at the base of the flattening filter. It has an exponentially decreasing spectrum, with a maximum energy that is determined during the commissioning process and an energy dependent Gaussian fluence.

With respect to VSM1.5, VSM1.6 includes a better description of the electron source spectrum that avoids the situations in which electron energies above the maximum photon energy occurred. Also, the model of the horn corrections can be done with tabulated values instead of using the function proposed by Fippel *et al.* [74] allowing a better tuning of the fluence in those cases in which the flattening filter shows complex shapes.

To simulate the lower part of the linac, that is the movable jaws and the MLC, the second component of Monaco relies on transmission filters. Once the characteristics (emitting source, initial position, direction of movement and energy) have been determined according to the virtual source model, the algorithm finds the probability of going through these collimators by means of an analytical model. Jaws can be defined as full absorbers to reduce simulation time. The MLC is modeled in such a way that the variation of the transmission value between its geometrical components can be established permitting, for example, to take into account interleaf leakage.

Monaco uses XVMC [18, 20, 22] for the estimation of the absorbed dose distribution in

the CT of the patient. The main differences with general-purpose Monte Carlo codes are that XVMC supports only low  $Z$  materials, whose cross sections are scaled from those of water using the corresponding electron and mass densities. Only voxelized geometries can be simulated. The allowed energy range is between 1 and 25 MeV and the mass density of materials must be below  $3 \text{ g/cm}^3$ . The variance-reduction technique of electron history repetition [22, 38] is applied, even in heterogeneous media, to speed up the simulation. As in the case of iPlan, both dose to medium (default) and dose to water can be obtained. The number of histories to be simulated is determined from the uncertainty selected by the user, which is considered for each single beam. The user is informed of the achieved statistical uncertainty of the entire plan and of each segment.

To carry out the beam configuration several open field dose profiles and output factors are required. For VSM1.6 only in-water measurements must be provided while VSM1.5 needs additional in-air profiles. Specifically, central depth dose and transverse profiles, output factors, diagonal profiles and absolute doses for several source-to-detector distances (in case of in-air measurements) and various source-to-surface distances and depths (for in-water measurements) are necessary. Measurements must be done for both square and rectangular fields with lateral dimensions ranging between 1 cm and the maximum available for the particular linac. Also dose profiles for wedged fields with various wedge angles and the corresponding wedge transmission factors are required, although Monaco does not support the simulation of blocks or wedges. A series of measurements performed for several segment combinations must to be supplied for the fine tuning of the MLC. Beam configuration of the virtual sources is performed by the company.

### M. Oncentra

Oncentra external beam is a self-contained fast Monte Carlo virtual source model treatment planning system distributed by Elekta. It implements non-stochastic algorithms for treatment planning, as well as a fast Monte Carlo algorithm, described herein, that can be used for the simulation of electron beams. The system was originally marketed by Nucletron as Oncentra MasterPlan and it can simulate most linacs from major vendors (see table III). The patient-independent part of the linac is simulated using a coupled multi-source beam model [91] to create a phase space. This phase space is propagated through the patient-



dependent parts of the linac including the electron applicator, and eventually the Cerrobend insert, for creating an exit phase space downstream of the whole linac. The multiple source model takes into account both electrons that do not suffer any interaction with the collimation system, and particles (electrons or contaminant photons) that interact with it. The exit phase space is used as radiation source by the dose estimation engine that is based on VMC++ [21].

The user determines the number of simulated histories per unit area. The voxel size is automatically set by the system remapping the original CT to fit within the limit of a maximum of 800,000 voxels. Thus, CT scans that contain a small anatomical volume result in voxel sizes also small. A database of 21 materials is used for the assignment of Hounsfield units into medium composition. The estimated dose is referred to water.

Beam configuration requires fluence measurements in air done without electron applicator, and dose measurements in water with the different applicators installed. The configuration of the beam is done by the manufacturer with the measurements provided by the user.

#### **N. Pinnacle**

A Monte Carlo module based on the fast code DPM [19] was written for this treatment planning system from Philips. The Monte Carlo module has never been distributed.

#### **O. PPlanUNC**

The planning system coded at the University of North Carolina at Chapel Hill (PPlanUNC) is a set of software tools for treatment planning in radiotherapy. It includes functions for outlining structures, virtual simulation, dose calculation and analysis, and IMRT planning. The dose calculation algorithms in PPlanUNC are convolution/superposition and pencil beam. However, the code provides a Monte Carlo interface package for EGSnrc which allows to translate the patient and beam information such that it can be used as input in Monte Carlo simulations performed with EGSnrc. This package also provides the tools to import the simulation results into PPlanUNC. Abella and co-workers have developed a similar interface package to link the MCNP5 Monte Carlo code with PPlanUNC [92]. PLa-

nUNC is available for Windows, Linux and Mac OS X. The compiled version for Windows can be downloaded from the site <https://sites.google.com/site/planunc/home>. It does not include the EGSnrc interface package. The source code can be obtained under request through the web site.

## P. PRIMO

PRIMO [93] simulates the radiation transport through the whole linac head and a binned water phantom or a CT of a patient using the general-purpose Monte Carlo code PENELOPE 2011 [41]. The system supports all the range of Varian Clinac C-series (18, 21, 23, iX, etc.), 600, Unique, TrueBeam, and Elekta SLi and MLCi series (see table III). TrueBeam can be simulated by importing PSFs distributed by Varian [94, 95] or by using an approximate empirical geometry called FakeBeam [96]. The linac geometries are part of the package, therefore, the user is not required to enter any geometrical information. PRIMO can produce PSFs at the downstream end of the upper and lower parts of the linac and can import external PSFs provided they are compliant with the IAEA specification [97].

For each simulation users can assign values to the initial beam parameters which are: primary electron beam energy, full width at half maximum (FWHM) of the primary energy distribution, FWHM of the focal spot size, and beam divergence. The code provides suitable default initial parameters for most available linacs and nominal energies. Users can fine-tune these parameters to obtain a better match between the simulated and the experimentally measured depth dose and lateral profiles of reference fields. The code does not provide a beam configuration algorithm. Therefore, the user must perform several simulations varying the primary beam energy until finding the most adequate one that reproduces the experimental depth dose profiles and then vary the FWHM of the focal spot size until reproducing the experimental lateral dose profiles.

To reduce simulation time the code incorporates a number of specifically developed variance-reduction techniques, namely, the movable skins technique [80], splitting roulette [98], rotational splitting [99] and fan splitting [100]. Other variance-reduction techniques are also available such as interaction forcing in the linac target to maximize the bremsstrahlung production and standard particle splitting in the patient. Additionally, the simulation can be distributed among the cores in a computer.

The patient representation is a binned matrix created from the CT scan which is imported from a DICOM file. In the same way, phantoms of slabs and homogeneous phantoms can be created directly in the program. PRIMO includes 40 clinically relevant materials in its database which are used to convert the Hounsfield units into material composition. The system allows to import anatomical structures from DICOM-RT STRUCT files or to delineate them.

The analysis tools include the creation of probability distributions from PSFs, the production of dose-volume histograms, dose profiles and the comparison of experimental dose measurements with the Monte Carlo estimated dose distributions using the gamma index. All these features are wrapped under a graphical user interface and runs on Windows. PRIMO is a full Monte Carlo and a self-contained system which can be used for dose verification and research purposes. It is freely distributed from the website <http://www.primoproject.net>.

#### **Q. RTGrid**

RTGrid [101] is a parallel computational environment for Monte Carlo simulation of conformal radiotherapy problems. It was developed at Cardiff University and the Velindre Cancer Centre. It uses Globe Toolkit for enabling the distributed management of computing resources. The RTGrid system is accessed through a web portal; although in order to run simulations, it is necessary to install a series of packages in the client system. Simulations are called “experiments” and are specified by templates known as “profiles”. Profiles are archives that contain simulation data and executable scripts to be run at different phases of the simulation and can be edited online in the portal. RTGrid uses EGSnrc/BEAMnrc [39, 58, 68] for the Monte Carlo simulations. The project is currently closed and access to the portal has been discontinued.

#### **R. SMCP**

The Swiss Monte Carlo Plan (SMCP) [102, 103] is a planning framework interfaced to the treatment planning system Eclipse (Varian Medical Systems). SMCP can use either the fast Monte Carlo code VMC++ [21] or the general-purpose Monte Carlo code EGSnrc [39] for the transport in the collimating structures and the patient geometry. Several levels of

complexity for the transport through the beam modifiers can be selected in order to speed up the simulation. The source of particles can be either a PSF tallied above the movable components of the linac or a virtual source. The particles in the phase space can be reused employing a technique that rotates the initial position of the particle by a random azimuthal angle [104], thus taking advantage of the cylindrical symmetry of the beam at the plane of the phase space. The system provides a calibration procedure to express the dose in Gy/MU that takes into account the fraction of radiation backscattered to the monitor ionization chamber. The code is not distributed and it is only used at the Inselspital of the University of Bern.

## S. VIMC

The Vancouver Island Monte Carlo (VIMC) system [105–108] is a Monte Carlo treatment plan verification system. The program is web-based. Users connect to a web server through one of the common browsers and use a web-page interface to configure submission jobs that are then sent to a cluster, where calculations are performed. VIMC relies on BEAMnrc [58, 68] for simulating treatment machines. The Varian Clinac 21EX series and TrueBeam models are configured. The latter is simulated by means of the PSFs distributed by Varian [95]. Subsequently, VIMC uses DOSXYZnrc [68, 69] or VMC++ [21] for the estimation of the absorbed dose distribution. VIMC has a dose visualization tool. VIMC relies on the treatment planning system Eclipse (Varian Medical Systems) for all aspects related to field configuration and dose analysis tasks. The code implements IMRT and VMAT with jaw-tracking as well as conformal treatments. For quality assurance purposes the system performs Monte Carlo simulations of the patient dose distribution from the fluence obtained with electronic portal imaging devices (EPID). The system also allows reconstructing the particle fluences from 21EX dynalog or TrueBeam trajectory files that record time-dependent characteristics of linac beam-limiting components during the treatment delivery. VIMC is only available to British Columbia Cancer Agency staff.

## T. XiO

XiO is a self-contained fast Monte Carlo virtual source model treatment planning system distributed by Elekta. It implements non-stochastic algorithms for treatment planning, as well as a fast Monte Carlo algorithm, described herein, that can be used for the simulation of electron beams. A virtual source model is used to create particles in the linac head which are subsequently transported through the linac, the electron applicator, and the Cerrobend insert by means of pre-calculated Monte Carlo kernels [109]. Those kernels were computed with the EGSnrc code [39] and the algorithm was developed by Elekta. The computation engine used for the Monte Carlo simulation of the absorbed dose distribution is based on an early version of XVMC [20].

XiO allows the user to define the voxel dimensions and to determine the statistical uncertainty that will be reached. Absorbed dose distributions can either be referred to medium or water. Most linacs manufactured by major vendors are supported (see table III).

Beam configuration requires experimental depth dose and lateral profiles in water, at least for seven depths and two source-to-surface distances, for each combination of energy and electron applicator. The system also requires a full set of measurements using the  $15 \times 15 \text{ cm}^2$  electron applicator with a Cerrobend insert that has an aperture of  $5 \times 5 \text{ cm}^2$ . Experimental measurements with an open field in the absence of electron applicator are needed too. The beam model parameters are tuned by the manufacturer.

## IV. BENCHMARKS

A set of benchmarks for measuring computation speed of the systems that are self-contained and distributed is presented. The purpose of these tests is to give the reader an order of magnitude of how much time is required to perform a simulation. The time required to reach a given statistical uncertainty of the dose distribution in the region of interest (planning target volume) is nearly independent from the number of simulated fields. Thus, in all benchmarks the simplest case of one irradiation field was considered. The benchmarks were designed to satisfy the requirements and capabilities of most of the tested systems, so to produce meaningful comparisons. The following fields were simulated:

- Field 1 (photon beam):

TABLE II: Comparison of self-contained and distributed systems. Abbreviations: pencil beam convolution (pbc), inverse treatment planning (ITP), beam quality assurance (BQA), reference fields (rf), done by manufacturer (dbm), not applicable (n/a), plus those in table I.

Features	Systems			
	Eclipse	eIMRT	iPlan	Monaco
linac upper part	vsm	gp (BEAMnrc)	vsm	vsm
linac lower part	vsm	gp (BEAMnrc)	fast	trans. filter
dose distribution	pc (MMC)	gp (DOSXYZnrc)	fast (XVMC)	fast (XVMC)
Beams				
electron ( $e^-$ )	yes	no	no	no
photon ( $\gamma$ )	no	yes	yes	yes
BQA tools in water	yes	no	no	yes (optional)
3DCRT	n/a	yes	yes	yes
IMRT	n/a	yes	yes	yes
VMAT	n/a	no	no	yes
Varian	C-Series	C-Series	600, C-Series Novalis	600, C-Series
Elekta	-	-	MLCi, MLCi2 Series	MLCi, MLCi2 Series
Siemens	-	Primus, Oncor	Mevatron, Primus, Oncor, Artiste	-
ITP optimization	n/a	no	pbc	pbc
Dose referred to	water	medium	medium or water	medium or water
Beam configuration	required measurements	rf in water	rf in water, air	rf in water
tuning algorithm	automatic	automatic	dbm	dbm
Operating system	Windows	Linux	Windows	Windows
FDA approval	yes	no	yes	yes

TABLE III: Continuation of table II.

Features	Systems		
	Oncentra	PRIMO	XiO
linac upper part	vsm	gp (PENELOPE)	vsm
linac lower part	vsm	gp (PENELOPE)	pc (Elekta)
dose distribution	fast (VMC++)	gp (PENELOPE)	fast (XVMC)
Beams			
electron ( $e^-$ )	yes	yes	yes
photon ( $\gamma$ )	no	yes	no
BQA tools in water	yes	yes	yes
3DCRT	n/a	yes	n/a
Modalities			
IMRT	n/a	no	n/a
VMAT	n/a	no	n/a
Varian	C-Series	600, Unique, C-Series, TrueBeam	C-Series
Linacs			
Elekta	SLi, MLCi, MLCi2 Series	SLi, MLCi Series	SLi, MLCi, MLCi2 Series
Siemens	Mevatron, Primus, Oncor	Mevatron, -	Mevatron, Primus, Oncor
ITP optimization	n/a	no	n/a
Dose referred to	water	medium	medium or water
Beam configuration			
required measurements	rf in water, air	rf in water	rf in water
tuning algorithm	dbm	manual	dbm
Operating system	Windows	Windows	Linux
FDA approval	yes	no	yes

- Energy: 6 MV
  - Field size:  $10 \times 10 \text{ cm}^2$ , centered
  - Gantry angle and collimator angle both equal 0
  - Source-to-surface distance: 100 cm
  - Dose-scoring region: homogeneous water phantom ( $\rho = 1 \text{ g/cm}^3$ )
  - Voxel size:  $3 \times 3 \times 3 \text{ mm}^3$
  - Average standard statistical uncertainty of the dose reached: 2% (with a coverage factor  $k = 1$ )
- Field 2 (photon beam):  
Same simulation conditions as field 1 but with a diamond-shaped aperture conformed with the MLC. The distance across opposite vertexes of the diamond is 5 cm. Collimator angle is kept to 0. Jaws circumscribe the field defined by the MLC (figure 1).
  - Field 3 (electron beam):  
Same simulation conditions as field 1 but using a 6 MeV electron beam. The  $10 \times 10 \text{ cm}^2$  electron applicator is used.
  - Field 4 (electron beam):  
Same simulation conditions as field 3 but using a 16 MeV electron beam.

The simulation conditions particular to each system were as follows:

- Eclipse: Voxel size in eMC was set to  $2.5 \times 2.5 \times 2.5 \text{ mm}^3$ , which is the closest value possible to the requested size.
- eIMRT: The user has no control on the reached statistical uncertainty. The simulation stops when a low uncertainty is reached. eIMRT does not support PSFs, therefore, the whole linac head was simulated for both fields.
- iPlan: Simulations were run with the option ‘accuracy optimized’ activated, thus leakage was simulated.
- Monaco: The reported uncertainty is the one called by the system as ‘per control point’, that is, the average statistical uncertainty on a region at the center of the field.



- Oncentra: The nominal energy used for the simulation of field 4 was 15 MeV. In Oncentra the desired statistical uncertainty cannot be set and it is not accessible to the user. Instead, the user must decide the number of simulated histories per  $\text{cm}^2$ . For both simulations the number was set to 150000 histories/ $\text{cm}^2$ . Based on the obtained dose profiles we estimated the standard statistical uncertainty reached around 2% (with a coverage factor  $k = 1$ ).
- PRIMO: The system supports tallying a PSF upstream the jaws. The simulation time for creating the PSF used for photon beams was 73800 s. If a specific PSF were tallied for the smaller  $5 \times 5 \text{ cm}^2$  field the required time would be 28800 s. The time required for scoring the PSFs for the 6 and 16 MeV electron beams was 3050 and 1440 s, respectively. The time devoted to tally these PSFs is not counted for benchmark purposes since it is patient-independent, so the PSFs can be computed once and for all. The time reported in table IV is that employed for simulating the patient-dependent part of the linac (jaws, MLC and electron applicator) and the radiation transport in the water phantom. The variance-reduction techniques used were movable skins in the jaws and the MLC. Splitting factors of 15 and 20 were used for the simulation of the photon and electron beams, respectively.
- XiO: The nominal energy used for the simulation of field 4 was 15 MeV. The simulations of XiO were run on an old computer manufactured in 2004. There is no official information from the manufacturer of the processor on its base speed in GFLOPS. Based on third-party processor benchmarks we estimated the base speed to be around 5 GFLOPS.

Table IV shows the simulation times required by each system to reach a standard statistical uncertainty of 2% (with a coverage factor  $k = 1$ ) under the aforementioned conditions. Two figures are given for each combination of system and field considered. The upper figure is the simulation time required in seconds ( $t$ ). The lower figure,  $n$ , is an estimation of the number, divided by  $10^{12}$ , of floating point operations (FLOP) done to reach that uncertainty. This estimation assumes that the whole simulation time has been dedicated to floating point operations. The purpose of  $n$  is to homogenize the simulation times obtained in different computers. For its calculation the base speed  $s$  of each processor, quoted by the



TABLE IV: Benchmarks. The columns from ‘Field 1’ to ‘Field 4’ contain for each system an upper and a lower figure. The former is the simulation time  $t$  in seconds. The latter is the number of tera-floating-point-operations. ‘# CPU’ indicates the number  $m$  of processors installed in the computer. ‘# core’ indicates the number of processing cores of each processor. All cores of all processors were used for each simulation. The model of the Intel Xeon processor installed on each computer is shown in the column ‘CPU’ together with its base computation speed  $s$  in ‘GFLOP’.

System	Version	Linac	# CPU	CPU	$\gamma$ beams		$e^-$ beams	
		MLC	# core	GFLOPS	Field 1	Field 2	Field 3	Field 4
Eclipse	10.0.28	Varian 2100	2	E5620			21	41
		n/a	4	34.8			1.5	2.9
eIMRT	2.1	Varian 2100	5	E5-2650	10260	9960		
		120	10	128	6566	6374		
iPlan	4.5.2	Varian 2100	2	E5-2667	28	9		
		120	6	139.2	7.8	2.5		
Monaco	5.0	Elekta Axesse	2	X5650	50	69		
		160	6	63.984	6.4	8.8		
Oncentra	4.0	Siemens Artiste	2	E5620			124	229
		n/a	4	34.8			8.6	16
PRIMO	0.1.5.1307	Varian 2100	2	E5-2470	1770	1410	1260	1860
		120	10	147.2	521	415	371	548
XiO	4.60	Siemens Primus	2	Xeon 2.8			200	390
		n/a	1	$\sim 5$			2	4

There are two reasons why the general-purpose Monte Carlo-based system eIMRT requires a much longer computer time with respect to PRIMO to simulate the benchmark fields. First, eIMRT does not support PSFs, therefore, the simulation time includes the whole linac head. Most of the time devoted to the linac head simulation is spent on the patient-independent part, which PRIMO simulated using a previously tallied PSF. The time that eIMRT devoted to the simulation of the linac head was 3960 s and 2520 s for fields 1 and 2,

respectively. Secondly, the system always reaches a low statistical uncertainty. The average standard statistical uncertainty of voxels scoring more than 50 % of the maximum dose was 1 % and 0.5 % for fields 1 and 2, respectively. After discounting the simulation time of the linac head and rescaling the simulation time to that required for reaching a 2 % standard statistical uncertainty, and assuming that 2 E5-2470 CPUs were used (instead of 5 E5-2650 CPUs) the resulting time and number of floating point operations that would be needed are 3424 s and 1000 TFLOP for field 1, and 1011 s and 300 TFLOP for field 2. These results are of the same order of magnitude than those obtained with PRIMO, namely, 1770 s and 521 TFLOP for field 1, and 1410 s and 415 TFLOP for field 2. Thus, although the simulation speed and number of required operations of eIMRT and PRIMO are similar, the structure of eIMRT may render it slow, unless a standard statistical uncertainty below 1 % is required, in which case the fact of simulating the whole linac can be compensated by the fact of running the simulation on a supercomputer with a large number of processors available.

## V. CONCLUSIONS

Although Monte Carlo algorithms for radiation transport have been incorporated for many years in radiotherapy applications for treatment planning and dose verification, only a few of such systems have reached the level of development of the non-stochastic alternatives. The most successful have been those commercial systems based on fast or pre-calculated Monte Carlo algorithms and virtual sources. Current systems based on fast Monte Carlo codes are between 50 to 100 times faster than the systems based on general-purpose codes.

Several of the systems reviewed herein have been abandoned, have disappeared or were never released. Others have their use limited to the development institutions. The relatively low presence of Monte Carlo systems in modern radiotherapy implies that its adaptation to the requirements of the clinical practice is still a challenge.

The increasing usage of smaller and highly modulated radiation fields aimed at improving the tumor-normal tissue dose ratio, will be accompanied by a growing interest on Monte Carlo treatment planning systems. As the demand grows and the interest of research groups and private companies increase accordingly, more resources will be placed on the development of clinically oriented self-contained and distributed systems. Of particular importance in the pursue of this objective will be the drastic reduction of the current, still prolonged,

simulation times of general-purpose Monte Carlo codes. It is foreseen that the advent of low-cost massive parallel computing will play a significant role in this regard.

### Acknowledgements

We thank the following persons for their help provided for preparing this review: Julio Almansa, Luca Cozzi, Nuria Escobar-Corral, Lluís Escudé, Matthias Fippel, Michael Fix, Andrea Flühs, Eric Franchisseur, Andrés Gómez, Diego González-Castaño, Damián Guirado, Marcelino Hermida-López, Gisela Hürtgen, Antonio Leal, Rodolfo del Moral, Carlos Mouriño, David Navarro Giménez, Josep Puxeu, José Manuel Reinoso, Carlos Sandín, Wolfgang Sauerwein, Josep Sempau, Emiliano Spezi, Hideki Takegawa and Sergei Zavgorodni. We are thankful to the Deutsche Forschungsgemeinschaft (project BR 4043/3-1), the Spanish Ministerio de Economía y Competitividad (projects FIS2012-38480 and FPA2012-31993), the European Regional Development Fund (ERDF), and the Junta de Andalucía (FQM0220).

- 
- [1] N. Reynaert, S. C. van der Marck, D. R. Schaart, W. Van der Zee, C. Van Vliet-Vroegindewij, M. Tomsej, J. Jansen, B. Heijmen, M. Coghe, and C. De Wagter, “Monte Carlo treatment planning for photon and electron beams,” *Rad. Phys. Chem.* **76**, 643–686 (2007).
  - [2] J. Cygler, J. J. Battista, J. W. Scrimger, E. Mah, and J. Antolak, “Electron dose distributions in experimental phantoms: a comparison with 2D pencil beam calculations,” *Phys. Med. Biol.* **32**, 1073–1083 (1987).
  - [3] C.-M. Ma, E. Mok, A. Kapur, T. Pawlicki, D. Findley, S. Brain, K. Forster, and A. L. Boyer, “Clinical implementation of a Monte Carlo treatment planning system,” *Med. Phys.* **26**, 2133–2143 (1999).
  - [4] M. R. Arnfield, C. Hartmann-Siantar, J. Siebers, P. Garmon, L. Cox, and R. Mohan, “The impact of electron transport on the accuracy of computed dose,” *Med. Phys.* **27**, 1266–1274 (2000).
  - [5] M. Miften, M. Wiesmeyer, A. Kapur, and C.-M. Ma, “Comparison of RTP dose distributions in heterogeneous phantoms with the BEAM Monte Carlo simulation system,” *J. Appl. Clin.*

- Med. Phys. **2**, 21–31 (2001).
- [6] K. De Vlaminck, H. Palmans, F. Verhaegen, C. De Wagter, W. De Neve, and H. Thierens, “Dose measurements compared with Monte Carlo simulations of narrow 6 MV multileaf collimator shaped photon beams,” Med. Phys. **34**, 4818–4853 (2007).
- [7] M. Rincón, F. Sánchez-Doblado, M. Perucha, A. Leal, R. Arráns, E. Carrasco, J. A. Sánchez-Calzado, and L. Errazquin, “A Monte Carlo approach for small electron beam dosimetry,” Radiother. Oncol. **58**, 179–185 (2001).
- [8] F. Sánchez-Doblado, P. Andreo, R. Capote, A. Leal, M. Perucha, R. Arráns, L. Núñez, E. Mainegra, J. I. Lagares, and E. Carrasco, “Ionization chamber dosimetry of small photon fields: a Monte Carlo study on stopping-power ratios for radiosurgery and IMRT beams,” Phys. Med. Biol. **48**, 2081–2099 (2003).
- [9] I. J. Chetty, B. Curran, J. E. Cygler, J. J. DeMarco, G. Ezzell, B. A. Faddegon, I. Kawrakow, P. J. Keall, H. Liu, C.-M. C. Ma, D. W. O. Rogers, J. Seuntjens, D. Sheikh-Bagheri, and J. V. Siebers, “Report of the AAPM Task Group No. 105: Issues associated with clinical implementation of Monte Carlo-based photon and electron external beam treatment planning,” Med. Phys. **26**, 1874–1882 (1999).
- [10] R. Alfonso, P. Andreo, R. Capote, M. Saiful Huq, W. Kilby, P. Kjäll, T. R. Mackie, H. Palmans, K. Rosser, J. Seuntjens, W. Ullrich, and S. Vatnitsky, “A new formalism for reference dosimetry of small and nonstandard fields,” Med. Phys. **35**, 5179–5186 (2008).
- [11] I. J. Das, G. X. Ding, and A. Ahnesjö, “Small fields: Nonequilibrium radiation dosimetry,” Med. Phys. **35**, 206–215 (2008).
- [12] L. Brualla, R. Palanco-Zamora, A. Wittig, J. Sempau, and W. Sauerwein, “Comparison between PENELOPE and electron Monte Carlo simulations of electron fields used in the treatment of conjunctival lymphoma,” Phys. Med. Biol. **54**, 5469–5481 (2009).
- [13] V. Panettieri, P. Barsoum, M. Westermark, L. Brualla, and I. Lax, “AAA and PBC calculation accuracy in the surface build-up region in tangential beam treatments. Phantom and breast case study with the Monte Carlo code PENELOPE,” Radiother. Oncol. **93**, 94–101 (2009).
- [14] L. Brualla, R. Palanco-Zamora, K.-P. Steuhl, N. Bornfeld, and W. Sauerwein, “Monte Carlo simulations applied to conjunctival lymphoma radiotherapy treatment,” Strahlenther. Onkol. **187**, 492–498 (2011).

- [15] L. Brualla, P. A. Mayorga, A. Flühls, A. M. Lallena, J. Sempau, and W. Sauerwein, “Retinoblastoma external beam photon irradiation with a special “D”-shaped collimator: a comparison between measurements, Monte Carlo simulation and a treatment planning system calculation,” *Phys. Med. Biol.* **57**, 7741–7751 (2012).
- [16] L. Brualla, F. J. Zaragoza, J. Sempau, A. Wittig, and W. Sauerwein, “Electron irradiation of conjunctival lymphoma–Monte Carlo simulation of the minute dose distribution and technique optimization,” *Int. J. Radiat. Oncol. Biol. Phys.* **83**, 1330–1337 (2012).
- [17] P. A. Mayorga, L. Brualla, W. Sauerwein, and A. M. Lallena, “Monte Carlo study for designing a dedicated “D”-shaped collimator used in the external beam radiotherapy of retinoblastoma patients,” *Med. Phys.* **41**, 011714 (2012).
- [18] I. Kawrakow, M. Fippel, and K. Friedrich, “3D electron dose calculation using Voxel based Monte Carlo algorithm (VMC),” *Med. Phys.* **23**, 445–447 (1996).
- [19] J. Sempau, S. Wilderman, and A. Bielajew, “DPM, a fast, accurate Monte Carlo code optimized for photon and electron radiotherapy treatment planning dose calculations,” *Phys. Med. Biol.* **45**, 2263–2291 (2000).
- [20] M. Fippel, “Fast Monte Carlo dose calculation for photon beams based on the VMC electron algorithm,” *Med. Phys.* **26**, 1466–1475 (1999).
- [21] I. Kawrakow and M. Fippel, “VMC++, a fast MC algorithm for radiation treatment planning,” in *The use of computers in radiation therapy*, edited by W. Schlegel, T. Bortfeld (Springer-Verlag, Heidelberg, 2000), pp. 126–128.
- [22] I. Kawrakow and M. Fippel, “Investigation of variance reduction techniques for Monte Carlo photon dose calculation using XVMC,” *Phys. Med. Biol.* **45**, 2163–2183 (2000).
- [23] G. Bueno, O. Déniz, C. B. Carrascosa, J. M. Delgado, and L. Brualla, “Fast Monte Carlo simulation on a voxelized human phantom deformed to a patient,” *Med. Phys.* **36**, 5162–5174 (2009).
- [24] B. Habib, B. Poumarede, F. Tola, and J. Barthe, “Evaluation of PENFAST – A fast Monte Carlo code for dose calculations in photon and electron radiotherapy treatment planning,” *Physica Medica: Eur. J. Med. Phys.* **26**, 17–25 (2010).
- [25] A. Badal and J. Sempau, “A package of Linux scripts for the parallelization of Monte Carlo simulations,” *Comput. Phys. Commun.* **175**, 440–450 (2006).
- [26] X. Jia, X. Gu, J. Sempau, D. Choi, A. Majumdar, and S. B. Jiang, “Development of a GPU-

- based Monte Carlo dose calculation code for coupled electron–photon transport,” *Phys. Med. Biol.* **55**, 3077–3086 (2010).
- [27] X. Jia, X. Gu, Y. J. Graves, M. Folkerts, and S. B. Jiang, “GPU-based fast Monte Carlo simulation for radiotherapy dose calculation,” *Phys. Med. Biol.* **56**, 7017–7031 (2011).
- [28] T. M. Jenkins, W. R. Nelson, and A. Rindi (Eds.), *Monte Carlo transport of electrons and photons*. (Plenum Press, New York, 1988).
- [29] C.-M. Ma, B. A. Faddegon, D. W. O. Rogers, and T. R. Mackie, “Accurate characterization of the Monte Carlo calculated electron beams for radiotherapy,” *Med. Phys.* **24**, 401–417 (1997).
- [30] J. J. DeMarco, T. D. Solberg, and J. B. Smathers, “A CT-based Monte Carlo simulation tool for dosimetry planning and analysis,” *Med. Phys.* **25**, 1–11 (1998).
- [31] A. Kapur, C.-M. Ma, E. Mok, D. Findley, and A. L. Boyer, “Monte Carlo calculations of clinical electron beam output factors,” *Phys. Med. Biol.* **43**, 3479–3494 (1998).
- [32] B. A. Faddegon, J. Balogh, R. Mackenzie, and D. Scora, “Clinical considerations of Monte Carlo for electron radiotherapy treatment planning,” *Radiat. Phys. Chem.* **35**, 217–228 (1998).
- [33] L. Wang, C. Chui, and M. Lovelock, “A patient-specific Monte Carlo dose-calculation method for photon beams,” *Med. Phys.* **25**, 867–878 (1998).
- [34] P. Andreo, “Monte Carlo techniques in medical radiation physics,” *Phys. Med. Biol.* **36**, 861–920 (1991).
- [35] F. Verhaegen and J. Seuntjens, “Monte Carlo modelling of external radiotherapy photon beams,” *Phys. Med. Biol.* **48**, R107–R164 (2003).
- [36] D. W. O. Rogers, “Fifty years of Monte Carlo simulations for medical physics,” *Phys. Med. Biol.* **51**, R287–R301 (2006).
- [37] E. Spezi and G. Lewis, “An overview of Monte Carlo treatment planning for radiotherapy,” *Radiat. Prot. Dosim.* **131**, 123–129 (2008).
- [38] J. Seco and F. Verhaegen, *Monte Carlo techniques in radiation therapy*. (CRC Press, Boca Raton, 2013).
- [39] I. Kawrakow, “Accurate condensed history Monte Carlo simulation of electron transport. I. EGSnrc, the new EGS4 version,” *Med. Phys.* **27**, 485–498 (2000).
- [40] J. Sempau, E. Acosta, J. Baró, J. M. Fernández-Varea, and F. Salvat, “An algorithm for



- Monte Carlo simulation of coupled electron-photon transport,” Nucl. Instrum. Meth. Phys. Res. B **132**, 377–390 (1997).
- [41] F. Salvat, J. M. Fernández-Varea, and J. Sempau, *PENELOPE 2011– A code system for Monte Carlo simulation of electron and photon transport* (OECD Nuclear Energy Agency, Paris, 2011).
- [42] S. Agostinelli, J. Allison, K. Amako, J. Apostolakis, H. Araujo, P. Arce, M. Asai, D. Axen, S. Banerjee, G. Barrand, F. Behner, L. Bellagamba, J. Boudreau, L. Broglia, A. Brunengo, H. Burkhardt, S. Chauvie, J. Chuma, R. Chytracsek, G. Cooperman, G. Cosmo, P. Degtyarenko, A. Dell’Acqua, i, G. Depaola, D. Dietrich, R. Enami, A. Feliciello, C. Ferguson, H. Fesefeldt, G. Folger, F. Foppiano, A. Forti, S. Garelli, S. Giani, R. Giannitrapani, D. Gibin, and J. J. Gómez Cadenas, “Geant4—a simulation toolkit,” Nucl. Instrum. Meth. Phys. Res. A **506**, 250–303 (2003).
- [43] L. S. Waters (Ed.), *MCNPX User’s Manual*, RSICC CCC-715 (Los Alamos National Laboratory, Los Alamos, 2002).
- [44] T. Goorley, M. James, T. Booth, F. Brown, J. Bull, L. J. Cox, J. Durkee, J. Elson, M. Fensin, R. A. Forster, J. Hendricks, H. G. Hughes, R. Johns, B. Kiedrowski, R. Martz, S. Mashnik, G. McKinney, D. Pelowitz, R. Prael, J. Sweezy, L. Waters, T. Wilcox, and T. Zukaitis, *Initial MCNP6 release overview of MCNP6 Version 1.0*, LA-UR-13-22934 (Los Alamos National Laboratory, Los Alamos, 2013).
- [45] M. Vilches, S. García-Pareja, R. Guerrero, M. Anguiano, and A. M. Lallena, “Monte Carlo simulation of the electron transport through thin slabs: A comparative study of PENELOPE GEANT3 GEANT4 EGSnrc and MCNPX,” Nucl. Instrum. Meth. Phys. Res. B **254**, 219–230 (2007).
- [46] M. Vilches, S. García-Pareja, R. Guerrero, M. Anguiano, and A. M. Lallena, “Monte Carlo Simulation of the Electron Transport Through Air Slabs: A comparative study of PENELOPE GEANT3 GEANT4 and EGSnrc,” IEEE Trans. Nucl. Sci. **55**, 710–716 (2008).
- [47] M. Vilches, S. García-Pareja, R. Guerrero, M. Anguiano, and A. M. Lallena, “Multiple scattering of 13 and 20 MeV electrons by thin foils: a Monte Carlo study with GEANT geant4, and PENELOPE,” Med. Phys. **36**, 3964–3970 (2009).
- [48] B. A. Faddegon, I. Kawrakow, Y. Kubyshev, J. Perl, J. Sempau, and L. Urban, “The accuracy of EGSnrc, Geant4 and PENELOPE Monte Carlo systems for the simulation of electron scatter

- in external beam radiotherapy,” *Phys. Med. Biol.* **54**, 6151–6163 (2009).
- [49] K. Jabbari, “Review of fast Monte Carlo codes for dose calculation in radiation therapy treatment planning,” *J. Med. Signals Sens.* **1**, 73–86 (2011).
- [50] I. Kawrakow, “VMC++, electron and photon Monte Carlo calculations optimized for radiation planning,” in *Advanced Monte Carlo for radiation physics, particle transport simulation and applications, Proceedings of the Monte Carlo 2000 Conference*, edited by A. Kling, F. Barao, M. Nakagawa, L. Tavora, P. Vaz (Springer-Verlag, Berlin, 2000), pp. 229–236.
- [51] I. Kawrakow and M. Fippel, “VMC++, a MC algorithm optimized for electron and photon beam dose calculations for RTP,” in *Proceedings of the 22nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Vol. 2* (IEEE, Chicago, 2000).
- [52] H. Neuenschwander, T. Mackie, and P. Reckwerdt, “MMC—a high-performance Monte Carlo code for electron beam treatment planning,” *Phys. Med. Biol.* **40**, 543–574 (1995).
- [53] P. J. Keall and P. W. Hoban, “Super-Monte Carlo: A 3-D electron beam dose calculation algorithm,” *Med. Phys.* **23**, 2023–2034 (1996).
- [54] F. J. Salguero, B. Palma, R. Arráns, J. Roselló, and A. Leal, “Modulated electron radiotherapy treatment planning using a photon multileaf collimator for post-mastectomized chest walls,” *Radiother. Oncol.* **93**, 625–632 (2009).
- [55] F. J. Salguero, R. Arráns, B. A. Palma, and A. Leal, “Intensity- and energy-modulated electron radiotherapy by means of an xMLC for head and neck shallow tumors,” *Phys. Med. Biol.* **55**, 1413–1427 (2010).
- [56] B. A. Palma, A. Ureba Sánchez, F. J. Salguero, R. Arráns, C. Míguez Sánchez, A. Walls Zurita, M. I. Romero Hermida, and A. Leal, “Combined modulated electron and photon beams planned by a Monte-Carlo-based optimization procedure for accelerated partial breast irradiation,” *Phys. Med. Biol.* **57**, 1191–1202 (2012).
- [57] A. Ureba, F. J. Salguero, A. R. Barbeiro, E. Jiménez-Ortega, J. A. Baeza, H. Miras, R. Linares, M. Perucha, and A. Leal, “MCTP system model based on linear programming optimization of apertures obtained from sequencing patient image data maps,” *Med. Phys.* **41**, 081719 (2014).
- [58] D. W. O. Rogers, B. R. Walters, and I. Kawrakow, *BEAMnrc Users Manual*, NRCC Report PIRS-0509(A)revL (National Research Council of Canada, Ottawa, 2011).
- [59] J. O. Deasy, A. I. Blanco, and V. H. Clark, “CERR: A computational environment for

- radiotherapy research,” *Med. Phys.* **30**, 979–985 (2003).
- [60] N. Mukumoto, K. Tsujii, S. Saito, M. Yasunaga, H. Takegawa, T. Yamamoto, H. Numasaki, and T. Teshima, “A preliminary study of in-house Monte Carlo simulations: an integrated Monte Carlo verification system,” *Int. J. Radiat. Oncol. Biol. Phys.* **75**, 571–579 (2009).
- [61] C. L. Hartmann Siantar, R. S. Walling, T. P. Daly, B. Faddegon, N. Albright, P. Bergstrom, A. F. Bielajew, C. Chuang, D. Garrett, R. K. House, D. Knapp, D. J. Wiczorek, and L. J. Verhey, “Description and dosimetric verification of the PEREGRINE Monte Carlo dose calculation system for photon beams incident on a water phantom,” *Med. Phys.* **28**, 1322–1337 (2001).
- [62] A. E. Schach von Wittenau, L. J. Cox, P. M. Bergstrom, J. W. P. Chandler, C. L. Hartmann Siantar, and R. Mohan, “Correlated histogram representation of Monte Carlo derived medical accelerator photon-output phase space,” *Med. Phys.* **26**, 1196–1211 (1999).
- [63] R. Nelson, H. Hirayama, and D. W. O. Rogers, *The EGS4 Code System*, SLAC-265 (SLAC, Stanford, 1985).
- [64] J. J. Janssen, E. W. Korevaar, L. J. van Battum, P. R. Storchi, and H. Huizenga, “A model to determine the initial phase space of a clinical electron beam from measured data,” *Phys. Med. Biol.* **46**, 269–286 (2001).
- [65] M. K. Fix, J. Cygler, D. Frei, W. Volken, H. Neuenschwander, E. J. Born, and P. Manser, “Generalized eMC implementation for Monte Carlo dose calculation of electron beams from different machine types,” *Phys. Med. Biol.* **58**, 2841–2859 (2013).
- [66] J. Pena, D. M. González-Castaño, F. Gómez, A. Gago-Arias, F. J. González-Castaño, D. Rodríguez-Silva, A. Gómez, C. Mouriño, M. Pombar, and M. Sánchez, “eIMRT: a web platform for the verification and optimization of radiation treatment plans,” *J. Appl. Clin. Med. Phys.* **10**, 205–220 (2009).
- [67] J. C. Mouriño Gallego, A. Gómez, L. Carril, Z. Martín Rodríguez, D. Lezzi, R. Rafanell, and R. M. Badía, “Execution of Monte Carlo treatment verification on Cloud using COMPSs platform,” in *Proceedings of the Third European Workshop on Monte Carlo Treatment Planning* (2012), p. 186–189.
- [68] I. Kawrakow and B. R. Walters, “Efficient photon beam dose calculations using DOSXYZnrc with BEAMnrc,” *Med. Phys.* **33**, 3046–3056 (2006).
- [69] B. R. Walters, I. Kawrakow, and D. W. O. Rogers, *DOSXYZnrc Users Manual*, NRCC

Report PIRS-794revB (National Research Council of Canada, Ottawa, 2009).

- [70] D. A. Low, W. B. Harms, S. Mutic, and J. A. Purdy, “A technique for the quantitative evaluation of dose distributions,” *Med. Phys.* **25**, 656–661 (1998).
- [71] J. Pena, D. M. González-Castaño, F. Gómez, F. Sánchez-Doblado, and G. H. Hartmann, “Automatic determination of primary electron beam parameters in Monte Carlo simulation,” *Med. Phys.* **34**, 1076–1084 (2007).
- [72] I. Kawrakow, D. W. O. Rogers, and B. R. Walters, “Large efficiency improvements in BEAM-nrc using directional bremsstrahlung splitting,” *Med. Phys.* **31**, 2883–2898 (2004).
- [73] Brainlab, *iPlan RT version 4.5, Clinical user guide, rev. 1.1* (Brainlab, Feldkirchen, 2011).
- [74] M. Fippel, F. Haryanto, O. Dohm, F. Nüsslin, and S. Kriesen, “A virtual photon energy fluence model for Monte Carlo dose calculation,” *Med. Phys.* **30**, 301–311 (2003).
- [75] M. Fippel, “Efficient particle transport simulation through beam modulating devices for Monte Carlo treatment planning,” *Med. Phys.* **31**, 1235–1242 (2004).
- [76] M. J. Berger and J. H. Hubbell, *XCOM: Photon Cross Sections on a Personal Computer*, NBSIR 87-3597 (NIST, Gaithersburg, 1987).
- [77] M. J. Berger, *ESTAR, PSTAR, and ASTAR: Computer Programs for Calculating Stopping-Power and Range Tables for Electrons, Protons, and Helium Ions*, NISTIR 4999 (NIST, Gaithersburg, 1993).
- [78] A. Isambert, L. Brualla, and D. Lefkopoulos, “Evaluation of the material assignment method used by a Monte Carlo treatment planning system,” *Cancer Radiother.* **13**, 744–746 (2009).
- [79] A. Isambert, L. Brualla, M. Benkebil, and D. Lefkopoulos, “Determination of the optimal statistical uncertainty to perform electron-beam Monte Carlo absorbed dose estimation in the target volume,” *Cancer Radiother.* **14**, 89–95 (2010).
- [80] L. Brualla, F. Salvat, and R. Palanco-Zamora, “Efficient Monte Carlo simulation of multi-leaf collimators using geometry-related variance-reduction techniques,” *Phys. Med. Biol.* **54**, 4131–4149 (2009).
- [81] M. Berger, “Monte Carlo calculation of the penetration and diffusion of fast charged particles,” in *Methods in computational physics, Vol. I*, edited by B. Alder, S. Fernbach, M. Rotenberg (Academic Press, New York, 1963).
- [82] N. Reynaert, B. De Smedt, M. Coghe, L. Paelinck, B. Van Duyse, W. De Gersem, C. De Wagter, W. De Neve, and H. Thierens, “MCDE: a new Monte Carlo dose engine for IMRT,”

- Phys. Med. Biol. **49**, N235–N241 (2004).
- [83] G. W. Sherouse and E. L. Chaney, “The portable virtual simulator,” *Int. J. Radiat. Oncol. Biol. Phys.* **21**, 475–482 (1991).
- [84] B. De Smedt, *Development of a Monte Carlo dose engine for IMRT treatment planning*, Ph. D. Thesis, Universiteit Gent (2006).
- [85] C.-M. Ma, J. S. Li, T. Pawlicki, S. B. Jiang, J. Deng, M. C. Lee, T. Koumrian, M. Luxton, and S. Brain, “A Monte Carlo dose calculation tool for radiotherapy treatment planning,” *Phys. Med. Biol.* **47**, 1671–1689 (2002).
- [86] M. N. Usmani, H. Takegawa, M. Takashina, H. Numasaki, M. Suga, Y. Anetai, K. Kurosu, M. Koizumi, and T. Teshima, “Development and reproducibility evaluation of a Monte Carlo-based standard LINAC model for quality assurance of multi-institutional clinical trials,” *J. Radiat. Res.* **55**, 1131–1140 (2014).
- [87] A. Alexander, F. DeBlois, G. Stroian, K. Al-Yahya, E. Heath, and J. Seuntjens, “MMCTP: a radiotherapy research environment for Monte Carlo and patient-specific treatment planning,” *Phys. Med. Biol.* **52**, N297–N308 (2007).
- [88] M. Sikora, O. Dohm, and M. Alber, “A virtual photon source model of an Elekta linear accelerator with integrated mini MLC for Monte Carlo based IMRT dose calculation,” *Phys. Med. Biol.* **52**, 4449–4463 (2007).
- [89] M. Sikora, *Virtual source modeling of photon beams for Monte Carlo based radiation therapy treatment planning*, Ph. D. Thesis, University of Bergen (2010).
- [90] M. Sikora and M. Alber, “A virtual source model of electron contamination of a therapeutic photon beam,” *Phys. Med. Biol.* **54**, 7329–7344 (2009).
- [91] E. Traneus, A. Ahnesjö, M. Fippel, I. Kawrakow, F. Nusslin, G. Zeng and M. Asell, “Application and verification of a coupled multi-source electron beam source model for Monte Carlo based treatment planning,” *Radiother. Oncol.* **61**, S102 (2001).
- [92] V. Abella, R. Miró, B. Juste, and G. Verdú, “Comparison of MCNP5 dose calculations inside the RANDO phantom irradiated with a MLC LinAc photon beam against treatment planning system PLUNC,” *Prog. Nucl. Sci. Technol.* **2**, 232–236 (2011).
- [93] M. Rodríguez, J. Sempau, and L. Brualla, “PRIMO: A graphical environment for the Monte Carlo simulation of Varian and Elekta linacs,” *Strahlenther. Onkol.* **189**, 881–886 (2013).
- [94] M. F. Belosi, M. Rodriguez, A. Fogliata, L. Cozzi, J. Sempau, A. Clivio, G. Nicolini, E.

- Vanetti, H. Krauss, C. Khamphan, P. Fenoglietto, J. Puxeu, D. Fedele, P. Mancosu, and L. Brualla, “Monte Carlo simulation of TrueBeam flattening-filter-free beams using Varian phase-space files: Comparison with experimental data,” *Med. Phys.* **41**, 051707 (2014).
- [95] M. Constantin, J. Perl, T. LoSasso, A. Salop, D. Whittum, A. Narula, M. Svatos, and P. J. Keall, “Modeling the TrueBeam linac using a CAD to Geant4 geometry implementation: Dose and IAEA-compliant phase space calculations,” *Med. Phys.* **38**, 4018–4024 (2011).
- [96] M. Rodriguez, J. Sempau, A. Fogliata, L. Cozzi, W. Sauerwein, and L Brualla, “A geometrical model for the Monte Carlo simulation of the TrueBeam linac,” *Phys. Med. Biol.* **60**, N219–N229 (2015).
- [97] R. Capote, R. Jeraj, C.-M. Ma, D. W. O. Rogers, F. Sánchez-Doblado, J. Sempau, J. Seuntjens, and J. V. Siebers, *Phase-space database for external beam radiotherapy*, Report INDC(NDS)-0484 (International Atomic Energy Agency, Vienna, 2006).
- [98] M. Rodríguez, J. Sempau, and L. Brualla, “A combined approach of variance-reduction techniques for the efficient Monte Carlo simulation of linacs,” *Phys. Med. Biol.* **57**, 3013–3024 (2012).
- [99] L. Brualla and W. Sauerwein, “On the efficiency of azimuthal and rotational splitting for Monte Carlo simulation of clinical linear accelerators,” *Rad. Phys. Chem.* **79**, 929–932 (2010).
- [100] J. Sempau, A. Badal, and L. Brualla, “A PENELOPE-based system for the automated Monte Carlo simulation of clinacs and voxelized geometries—application to far-from-axis fields,” *Med. Phys.* **38**, 5887–5895 (2011).
- [101] P. Downes, G. Yaikhom, J. P. Giddy, D. W. Walker, E. Spezi, and D. G. Lewis, “High-performance computing for Monte Carlo radiotherapy calculations,” *Phil. Trans. R. Soc. A* **367**, 2607–2617 (2009).
- [102] M. K. Fix, P. Manser, D. Frei, W. Volken, R. Mini, and E. J. Born, “An efficient framework for photon Monte Carlo treatment planning,” *Phys. Med. Biol.* **52**, N425–N437 (2007).
- [103] V. Magaddino, P. Manser, D. Frei, W. Volken, D. Schmidhalter, L. Hirschi, and M. K. Fix, “Validation of the Swiss Monte Carlo Plan for a static and dynamic 6 MV photon beam,” *Z. Med. Phys.* **21**, 124–134 (2011).
- [104] K. Bush, S. F. Zavgorodni and W. A. Beckham, “Azimuthal particle redistribution for the reduction of latent phase-space variance in Monte Carlo simulations,” *Phys. Med. Biol.* **52**, 4345–4360 (2007).

- [105] K. Bush, I. A. Popescu, and S. Zavgorodni, “A technique for generating phase-space-based Monte Carlo beamlets in radiotherapy applications,” *Phys. Med. Biol.* **53**, N337–N347 (2008).
- [106] K. Bush, R. Townson, and S. Zavgorodni, “Monte Carlo simulation of RapidArc radiotherapy delivery,” *Phys. Med. Biol.* **53**, N359–N370 (2008).
- [107] S. Zavgorodni, K. Bush, C. Locke, and W. Beckham, “Vancouver Island Monte Carlo (VIMC) system for radiotherapy treatment planning dosimetry and research,” *Radiother. Oncol.* **84 Suppl. 1**, S49 (2007).
- [108] S. Zavgorodni, K. Bush, C. Locke, and W. Beckham, “Vancouver Island Monte Carlo (VIMC) system for accurate radiotherapy dose calculations,” in *Proceedings of the 16th International Conference on Medical Physics* (2008), p. 78.
- [109] E. J. Vandervoort, E. Tchistiakova, D. J. La Russa, and J. E. Cygler, “Evaluation of a new commercial Monte Carlo dose calculation algorithm for electron beams,” *Med. Phys.* **41**, 021711 (2014).

PRIMO: A GRAPHICAL ENVIRONMENT FOR THE  
SIMULATION OF VARIAN AND ELEKTA LINACS

---

Rodriguez, M., Sempau, J. & Brualla, L. Strahlenther Onkol (2013) 189: 881.  
doi:10.1007/s00066-013-0415-1

ATTENTION !

Pages 68 to 74 of the thesis, containing the cited article, are  
available at the editor's web

<http://link.springer.com/article/10.1007/s00066-013-0415-1>



A COMBINED APPROACH OF  
VARIANCE-REDUCTION TECHNIQUES FOR THE  
EFFICIENT MONTE CARLO SIMULATION OF  
LINACS

---

M Rodriguez et al 2012 Phys. Med. Biol. 57 3013  
Doi: [10.1088/0031-9155/57/10/3013](https://doi.org/10.1088/0031-9155/57/10/3013)

ATTENTION !

Pages 76 to 88 of the thesis, containing the cited article, are  
available at the editor's web

<http://iopscience.iop.org/article/10.1088/0031-9155/57/10/3013/pdf>

# 5

## OPTIMIZATION OF THE EFFICIENCY IN MONTE CARLO SIMULATIONS OF LINACS THAT EMPLOY MULTIPLE VARIANCE-REDUCTION TECHNIQUES

---

## **Optimization of the efficiency in Monte Carlo simulations of linacs that employ multiple variance-reduction techniques**

Miguel Rodriguez<sup>a)</sup>

*Institut de Tècniques Energètiques, Universitat Politècnica de Catalunya  
Diagonal 647, E-08028 Barcelona, Spain.*

Lorenzo Brualla

*NCTeam, Strahlenklinik, Universitätsklinikum Essen  
Hufelandstraße 55, D-45122 Essen, Germany.*

Josep Sempau

*Institut de Tècniques Energètiques, Universitat Politècnica de Catalunya  
Diagonal 647, E-08028 Barcelona, Spain.*

The problem of obtaining the maximum simulation efficiency when the variance-reduction techniques of interaction forcing in the linac target and particle splitting in the water phantom are both combined in a Monte Carlo simulation is addressed. A theory is presented to derive the forcing and splitting factors that optimize the efficiency. In this model, a non-linear equation establishing a functional relation between the simulation efficiency and the splitting and forcing factors is derived. The theory was verified with simulations performed in a Varian linear accelerator and the optimal splitting and forcing factors were calculated by employing a non-linear optimization method. Results show that when optimal factors are applied, the efficiency is improved by a factor of 12.

## I. INTRODUCTION

The Monte Carlo simulation of radiation transport is computationally intensive. Attempts to reduce the computation time have been made in several directions, including simplifications in the physics models<sup>1-5</sup>, the use of parallel computing<sup>6,7</sup> and the use of variance-reduction techniques<sup>8</sup>. Variance-reduction techniques are intended to increase the simulation efficiency, defined by

$$\epsilon = \frac{1}{T\sigma^2}, \quad (1)$$

where  $T$  is the time employed and  $\sigma$  is the statistical standard deviation of the mean of the tallied quantity, *e.g.*, the absorbed dose. Two such techniques that are widely employed for the simulation of medical linear accelerators (linacs) are interaction forcing and particle splitting. The former consists in artificially increasing, for a given interaction mechanism, the particle interaction probability per unit path length by a user-defined factor  $F$ . The latter involves replicating the particle a user-defined number of times  $S$ . In both cases, to keep results unbiased, a statistical weight equal to  $1/F$  and  $1/S$  for interaction forcing and splitting, respectively, is associated to each particle. In a linac simulation interaction forcing can be employed to enhance bremsstrahlung production in the photon target, whereas splitting can be applied to increment the number of particles reaching the dose scoring region. A simulation in which no variance-reduction technique is applied is usually referred to as analogue simulation.

When variance-reduction techniques are employed the efficiency, given by equation (1), depends on the values of the involved parameters—*e.g.*,  $F$  or  $S$  in the cases discussed above. For a technique with a single parameter, the value for which the efficiency is maximized can be readily approximated by numerical trial and error. Since each trial requires a Monte Carlo simulation the optimization may be extremely time consuming when more than one parameter come into play, thus making the search for the maximum efficiency pointless.

Here we address the problem of optimizing linac simulation when interaction forcing and splitting are applied simultaneously by having recourse to a framework in which the dependency of  $\epsilon$  on both  $F$  and  $S$  is stated explicitly in analytical form. This allows obtaining the optimal values of these two parameters with a reduced number of trial simulations. In section II we introduce the model that leads to the theoretical framework. In section III

the model is verified by comparing its predictions with simulation results. Finally, some conclusions are drawn in section IV.

## II. THEORY

Consider a Monte Carlo simulation intended to estimate the absorbed dose distribution produced by a linac in a water phantom. As it is usually the case, suppose that this simulation is divided into two parts. Part one starts from a primary electron beam impinging on the linac target and simulates the radiation transport through the linac head. Particles arriving at the downstream end of the head are tallied in a phase-space file. Part two uses the previous phase-space file as its radiation source and simulates the transport downstream into a water phantom where the absorbed dose distribution is tallied. The simulation of one primary electron and of all its descendants will be called a history.

### II.A. Analytical framework

Let us assume that electron bremsstrahlung interactions in the target are forced by a factor  $F$ . Additionally, particles crossing the upstream surface of the water phantom are split into  $S$  identical copies. The goal of our analysis is to find the combination of  $F$  and  $S$  that maximizes the simulation efficiency.

The simulation time per history of part one,  $T_1 N^{-1}$ , is, on average, linearly dependent on the forcing factor  $F$ . Thus, it can be written as

$$\frac{T_1}{N} = t_a + Ft_b, \quad (2)$$

where  $N$  represents the total number of simulated histories and  $t_a$  and  $t_b$  are coefficients. Notice that  $t_b$  and  $t_a + t_b$  can be interpreted as the time per history in the processes involved in forced and analogue interactions, respectively.

In part two  $S$  identical copies of each particle crossing the phantom surface are created. Therefore, the time per history of part two,  $T_2 N^{-1}$ , will be

$$\frac{T_2}{N} = St_2, \quad (3)$$

where  $t_2$  is the proportionality factor. Thus, the total simulation time is

$$T = N(t_a + Ft_b + St_2). \quad (4)$$

According to Sempau and coworkers<sup>9</sup>, when splitting is used in the water phantom the variance  $\sigma^2$  of the mean can be expressed as

$$\sigma^2 = \frac{1}{N}(A + BS^{-1}), \quad (5)$$

where  $A$  and  $B$  are quantities (independent of  $N$  and  $S$ ) that depend on the simulation setup. They can be obtained as the fitting parameters when  $\sigma^2$  is represented versus  $S^{-1}$ . Notice that  $AN^{-1}$  is the so-called latent variance of the quantity of interest.

We shall assume that the quantity  $A$  is linearly dependent on the forcing factor and can be written as

$$A = A_0 + \frac{A_1}{F}, \quad (6)$$

where  $A_0$  and  $A_1$  are the coefficients of the line. Analogously, we shall assume that the quantity  $B$  is inversely proportional to the forcing factor, that is,

$$B = \frac{B_1}{F} \quad (7)$$

where  $B_1$  is the proportionality factor.

Combining equations (5), (6) and (7), the variance of the dose can be expressed as a function of the forcing and splitting factors

$$\sigma^2 = \frac{1}{N} \left[ A_0 + \left( A_1 + \frac{B_1}{S} \right) \frac{1}{F} \right]. \quad (8)$$

Using (4) and (8) the inverse of the simulation efficiency is

$$\epsilon^{-1} = [t_a + Ft_b + St_2] \left[ A_0 + \left( A_1 + \frac{B_1}{S} \right) \frac{1}{F} \right], \quad (9)$$

which can be recast as

$$\begin{aligned} \epsilon^{-1} = & A_0t_a + A_1t_b + (A_1t_a + B_1t_2)\frac{1}{F} + A_0t_bF + B_1t_b\frac{1}{S} + \\ & + A_0t_2S + A_1t_2\frac{S}{F} + B_1t_a\frac{1}{FS}. \end{aligned} \quad (10)$$

The problem of optimizing the efficiency has now been reduced to finding the values of  $F$  and  $S$  for which equation (10) has a minimum.

## II.B. Determination of parameters

Separate consideration of parts one and two described above allow the determination of the parameters in equations (2) and (3), respectively. We first consider equation (2) and obtain  $t_a$  and  $t_b$  by fitting a straight line to the representation of  $T_1 N^{-1}$  versus  $F$ . In the results reported below a series of simulations with  $F = 1, 5, 10, 50, 100$  and  $200$  have been performed. Analogously, the parameter  $t_2$  is obtained by fitting  $T_2 N^{-1}$  versus  $S$  to a straight line passing through the origin according to equation (3). In this case simulations were performed with  $S = 1, 10, 50, 100, 250$  and  $500$ .

The parameters appearing in equation (8) can be estimated from a series of simulations in which one factor, either  $F$  or  $S$ , is varied while the other remains fixed. In the case in which no splitting is performed ( $S = 1$ ), equation (8) becomes

$$N\sigma^2 = A_0 + (A_1 + B_1) \frac{1}{F}. \quad (11)$$

By fitting  $N\sigma^2$  versus  $F^{-1}$  to a straight line the parameters  $A_0$  and  $C \equiv A_1 + B_1$  are calculated. Conversely, when no forcing is performed ( $F = 1$ ), equation (8) becomes

$$N\sigma^2 = A_0 + A_1 + \frac{B_1}{S}. \quad (12)$$

Thus,  $D \equiv A_0 + A_1$  and  $B_1$  are inferred from the fitting of  $N\sigma^2$  versus  $S^{-1}$ . Notice that  $A_1$  can be obtained as either  $C - B_1$  or  $D - A_0$  thus allowing for a cross check on the consistency of equation (8).

Although not strictly necessary for the application of the optimization procedure, we have numerically checked the validity of equation (8). To this end, for each given forcing factor  $F$  simulations with splitting factors  $S = 1, 10, 25$  and  $50$  were performed in order to verify that equation (8) is valid not only for the particular cases defined in equations (11) and (12), but also when both  $F$  and  $S$  take on values different from unity.

## II.C. Simulations

All the simulations in this work were carried out with PRIMO<sup>10</sup> (PRIMO and its documentation can be freely downloaded from <http://www.primoproject.net>), a system based on the Monte Carlo code PENELOPE<sup>11</sup> that simulates linacs and the corresponding dose distribution in phantoms and computerized tomographies. PRIMO includes the coded geometries of most

Elekta and Varian linacs and various variance-reduction techniques, including interaction forcing and splitting. The linac chosen for the simulations conducted in this article was a Clinac series EX (Varian Medical Systems, Palo Alto) configured in photon mode. Without loss of generality, a nominal energy of 18 MV was chosen, since at this energy the photon production efficiency in the target is conveniently high. The initial energy of the monoenergetic primary electron source was 20.75 MeV. The jaws were opened so as to define a  $40 \times 40 \text{ cm}^2$  field at 100 cm from the target. The phase-space plane was situated at 70 cm from the target, just upstream the surface of the water phantom. The water phantom dimensions were  $50 \times 50 \times 20 \text{ cm}^3$  and it was gridded in cubic bins of 0.5 cm of side. Absorption energies and transport parameters for PENELOPE are automatically selected by PRIMO based on the linac configuration and nominal beam energy. For further details on the latter the reader is referred to the software manuals.

### III. RESULTS

All simulations were performed on an eight-core Intel Xeon processor running at 2.67 GHz. The simulations were automatically distributed by PRIMO among the eight cores in all cases. Figure 1 shows the computation time per history of part one and two for varying values of  $F$  and  $S$ , respectively. The parameters  $t_a$ ,  $t_b$  and  $t_2$  resulting from the linear fits are reported, together with their uncertainties, in table I. This table also reports the corresponding correlation coefficients, showing that they are closer to unity than  $\sim 10^{-6}$  and confirming that the linear dependence implied by equations (2) and (3) is indeed an excellent model.

Analogously, the variance as a function of either  $1/F$  or  $1/S$  has been fitted to straight lines as suggested by equations (11) and (12). From these fits the parameters  $A_0$ ,  $A_1$  and  $B_1$  were derived. The resulting values and the corresponding correlation coefficients and uncertainties are reported in table I. Additionally, to check the validity of equation (8) when both  $F$  and  $S$  vary, the slope of the fit of  $\sigma^2$  versus  $1/F$  was represented as a function of  $1/S$ . The results, which are displayed in figure 2, confirm the adequacy of the relation derived theoretically between the variance and the involved variance-reduction parameters.

Introducing the values from table I into equation (10) yields

$$\epsilon^{-1} = 5518.7 + \frac{4298.0}{F} + 33.0F + \frac{39654.2}{S} + 0.136S + 22.6\frac{S}{F} + \frac{29843.3}{FS}. \quad (13)$$



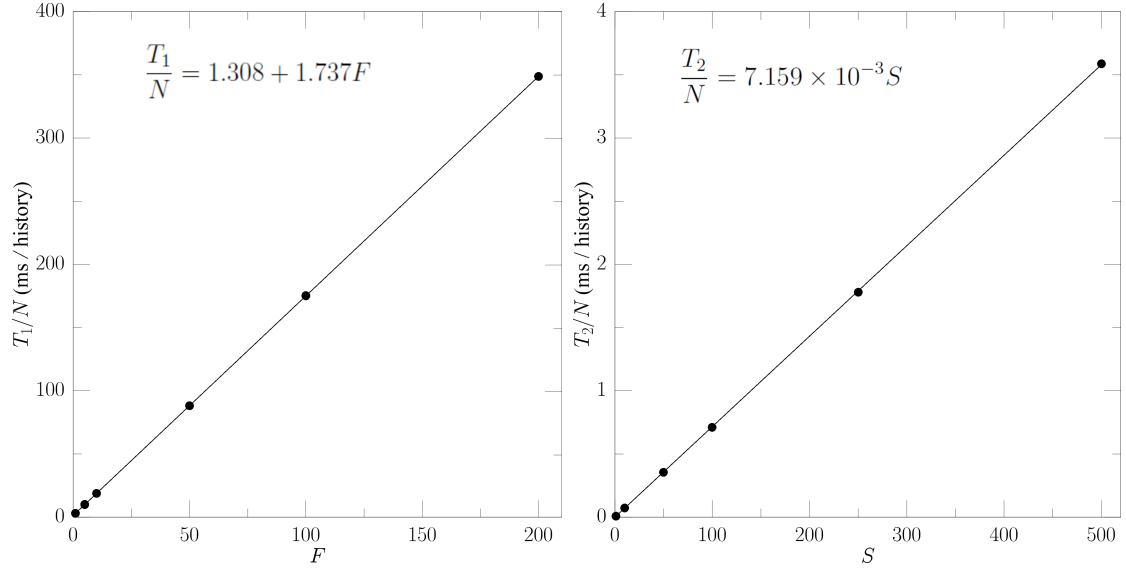


FIG. 1 Computation time per history of the simulation of parts one (left) and two (right) as functions of the forcing and splitting factors, respectively. The resulting linear fits are indicated in the legends.

TABLE I Parameters of equation (10) determined from fitting equations (2), (3), (11) and (12) to results depicted in figures 1 and 2. Discrepancy with unity of the correlation coefficient  $R$  of the fitting associated to the determination of each parameter is also shown.

Parameter	Value $\pm 1\sigma$	$1 - R$
$t_a$ (ms/history)	$1.31 \pm 0.04$	$1.25 \times 10^{-7}$
$t_b$ (ms/history)	$1.7375 \pm 0.0004$	$1.25 \times 10^{-7}$
$t_2$ ( $\mu$ s/history)	$7.15 \pm 0.01$	$1.14 \times 10^{-6}$
$A_0$ (Megahistories)	$0.0186 \pm 0.0002$	$6.00 \times 10^{-9}$
$A_1$ (Megahistories)	$3.161 \pm 0.005$	$1.55 \times 10^{-7}$
$B_1$ (Megahistories)	$22.816 \pm 0.009$	$1.55 \times 10^{-7}$

Projections along the  $F$  and  $S$  axis of the surface described by the inverse of this function,  $\epsilon$ , are shown in figure 3. The values of  $F$  and  $S$  that maximize the efficiency were obtained by a non-linear optimization procedure using the modified Newton's method, yielding  $F = 16$  and  $S = 164$ . The combined application of interaction forcing and splitting with the optimal values of the corresponding parameters enhances the efficiency by a factor of 12 with respect

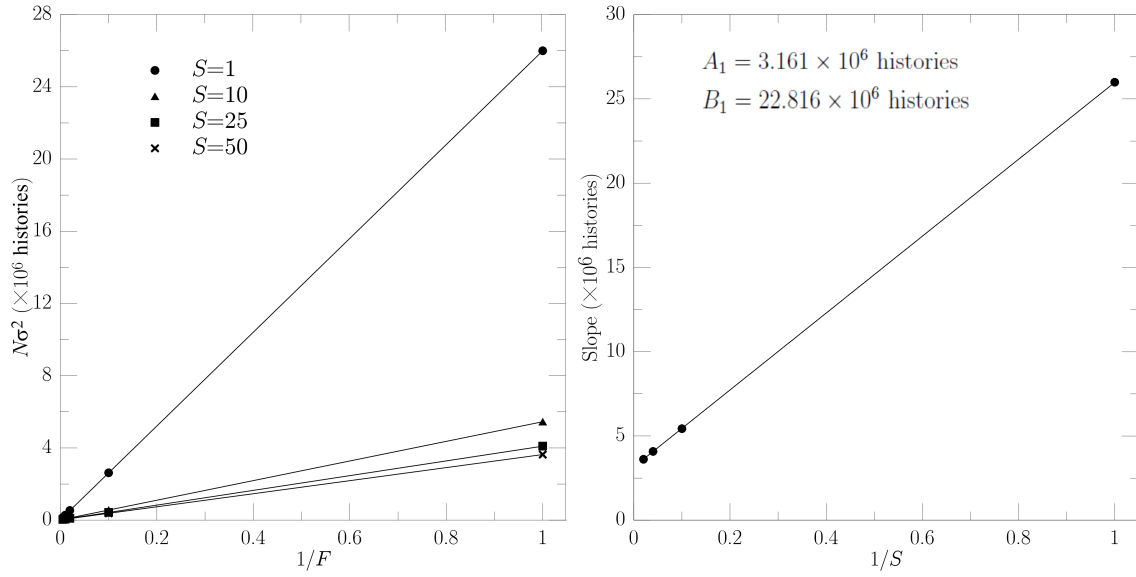


FIG. 2 Verification of equation ((8)) for a set of forcing and splitting factors (left). The dependence of the terms  $A_1 + B_1/S$  (i.e. the slopes of the linear fittings in the graph of the left), with the inverse of the splitting factor is shown in the graph of the right. In this case, the linear fitting renders the quantities  $A_1$  and  $B_1$  (values shown in the graph) as the slope and the independent term of the line, respectively.

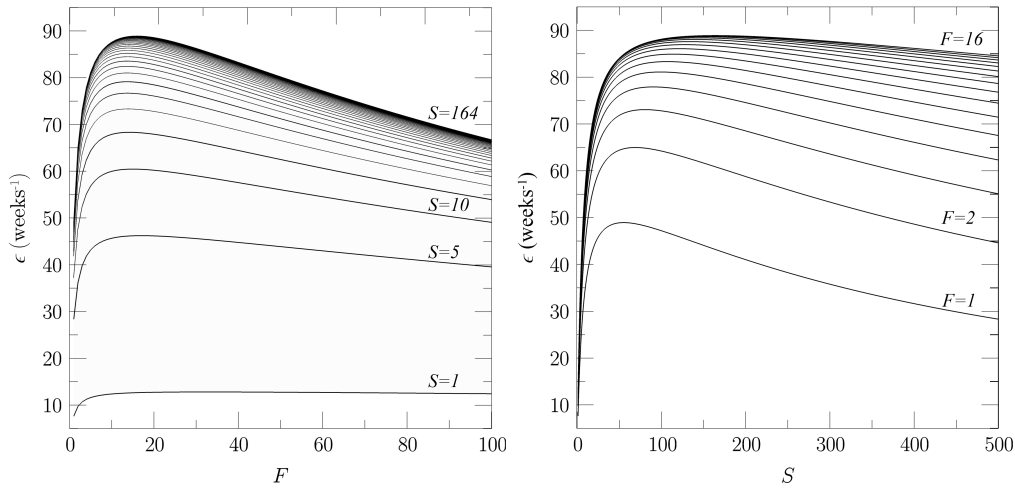


FIG. 3 Projected views of the surface determined by the inverse of equation (10) with the coefficients calculated according to the values of the parameters represented in table I. It can be observed that the efficiency has a maximum in the vicinity of  $F = 16$  and  $S = 164$ .

to an analogue simulation.

#### IV. CONCLUSION

We have elaborated and verified a theory to calculate the factors that optimize the efficiency when the variance-reduction techniques of interaction forcing in the linac target and particle splitting in the water phantom are both combined in a simulation. Using these optimal factors, the simulation efficiency is improved by a factor of 12. The optimized  $F$  and  $S$  values found in this article, as well as the factor by which the simulation efficiency is improved, are only valid for the particular case of linac model, energy, irradiated field and Monte Carlo code chosen. Nevertheless, the presented method is general and valid for any choice of linac, field and Monte Carlo code. The investigated variance-reduction techniques (with optimal factors set as default) are incorporated in the program PRIMO, a free distribution Monte Carlo code for linac simulation.

#### REFERENCES

<sup>a)</sup>miguel.lazaro.rodriguez@upc.edu

- <sup>1</sup>J. Sempau, S. Wilderman, and A. Bielajew, “DPM, a fast, accurate Monte Carlo code optimized for photon and electron radiotherapy treatment planning dose calculations,” *Phys. Med. Biol.* **45**, 2263–2291 (2000).
- <sup>2</sup>P. J. Keall and P. W. Hoban, “Super-Monte Carlo: A 3-D electron beam dose calculation algorithm,” *Med. Phys.* **23**, 2023 – 2034 (1996).
- <sup>3</sup>M. Fippel, “Fast Monte Carlo dose calculation for photon beams based on the VMC electron algorithm,” *Med. Phys.* **26**, 1466 – 1475 (1999).
- <sup>4</sup>C. L. Hartmann-Siantar, W. P. Chandler, J. A. Rathkopf, M. M. Svatos, and R. M. White, “PEREGRINE: An all-particle Monte Carlo code for radiation therapy,” in *Proceedings of the International Conference on Mathematics and Computations, Reactor Physics, and Environmental Analyses* (American Nuclear Society Press, La Grange Park, Illinois, 1995) pp. 857 – 865.
- <sup>5</sup>H. Neuenschwander, T. Mackie, and P. Reckwerdt, “MMC—a high-performance Monte Carlo code for electron beam treatment planning,” *Phys. Med. Biol.* **40**, 543–574 (1995).

- <sup>6</sup>S. Hissoiny, B. Ozell, H. Bouchard, and P. Despres, “GPUMCD: A new GPU-oriented Monte Carlo dose calculation platform,” *Med. Phys.* **38**, 754–758 (2011).
- <sup>7</sup>X. Jia, X. Gu, Y. J. Graves, M. Folkerts, and S. B. Jiang, “GPU-based fast Monte Carlo simulation for radiotherapy dose calculation,” *Phys. Med. Biol.* **56**, 7017–31 (2011).
- <sup>8</sup>T. M. Jenkins, W. R. Nelson, and A. Rindi, *Monte Carlo transport of electrons and photons*, Ettore Majorana International Science Series: Physical Sciences (Plenum Press, 1988).
- <sup>9</sup>J. Sempau, “Monte Carlo simulation of electron beams from an accelerator head using PENELOPE,” *Phys. Med. Biol.* **46**, 1163–1186 (2001).
- <sup>10</sup>M. Rodriguez, J. Sempau, and L. Brualla, “PRIMO: A graphical environment for the Monte Carlo simulation of Varian and Elekta linacs.” *Strahlentherapie und Onkologie* , 1–6 (2013).
- <sup>11</sup>F. Salvat, J. M. Fernández-Varea, and J. Sempau, *PENELOPE 2008—A Code System for Monte Carlo Simulation of Electron and Photon Transport* (OECD Nuclear Energy Agency, Issy-les-Moulineaux, France, 2009).

## STUDY OF THE ELECTRON TRANSPORT

PARAMETERS USED IN PENELOPE FOR THE  
MONTE CARLO SIMULATION OF LINAC TARGETS

---

Rodriguez, M., Sempau, J. & Brualla, L. Med. Phys. 42, 2877 (2015);  
doi: [10.1118/1.4916686](https://doi.org/10.1118/1.4916686)

## ATTENTION ;

Pages 102 to 106 of the thesis, containing the cited article, are  
available at the editor's web

<http://scitation.aip.org/content/aapm/journal/medphys/42/6/10.1118/1.4916686>

MONTE CARLO SIMULATION OF TRUEBEAM  
FLATTENING-FILTER-FREE BEAMS USING VARIAN  
PHASE-SPACE FILES: COMPARISON WITH  
EXPERIMENTAL DATA.

---

Maria F. Belosi et al. Med. Phys. 41, 051707 (2014);  
doi: [10.1118/1.4871041](https://doi.org/10.1118/1.4871041)

ATTENTION ;

Pages 108 to 118 of the thesis, containing the cited article, are  
available at the editor's web

<http://scitation.aip.org/content/aapm/journal/medphys/41/5/10.1118/1.4871041>

A GEOMETRICAL MODEL FOR THE MONTE CARLO  
SIMULATION OF THE TRUEBEAM LINAC

---

M.Rodriguez et al. Physics in Medicine and Biology, Volume 60, Number 11  
doi:10.1088/0031-9155/60/11/N219

ATTENTION !

Pages 120 to 130 of the thesis, containing the cited article, are  
available at the editor's web

<http://iopscience.iop.org/article/10.1088/0031-9155/60/11/N219>

## SUMMARY AND CONCLUSIONS

---

The main result of this thesis is a software system, called PRIMO, which simulates clinical linear accelerators and the subsequent dose distributions using the Monte Carlo method. PRIMO has the following features: (i) it is self-contained, that is, it does not require additional software libraries or coding; (ii) it includes a geometry library with most Varian and Elekta linacs; (iii) it is based on the general-purpose Monte Carlo code PENELOPE; (iv) it provides a suite of VRTs and distributed parallel computing to enhance the simulation efficiency; (v) it is GUI-driven; and (vi) it is freely distributed through the website <http://www.primoproject.net>.

In order to endow PRIMO with these features the following tasks were conducted:

- PRIMO was conceived with a layered structure. The topmost layer, named the GLASS, was developed in this thesis. The GLASS implements the GUI, drives all the functions of the system and performs the analysis of results. Lower layers generate geometry files, provide input data and execute the Monte Carlo simulation.
- The geometry of Elekta linacs from series SLi and MLCi were coded in the PRIMO system.
- A geometrical model of the Varian TrueBeam linear accelerator was developed and validated. This model was created to surmount the limitations of the Varian distributed phase-space files and the absence of released information about the actual geometry of that machine. This geometry model was incorporated into PRIMO.
- Two new variance-reduction techniques, named splitting roulette and selective splitting, were developed and validated. In a test made with an Elekta linac it was found that when both techniques are used in conjunction the simulation efficiency improves by a factor of up to 45.
- A method to automatically distribute the simulation among the available processing cores of a computer was implemented.

The following investigations were done using PRIMO as a research tool:



- The configuration of the condensed history transport algorithm for charged particles in PENELOPE was optimized for linac simulation. Dose distributions in the patient were found to be particularly sensitive to the values of the transport parameters in the linac target. Use of inadequate values of these parameters may lead to an incorrect determination of the initial beam configuration or to biased dose distributions.
- PRIMO was used to simulate phase-space files distributed by Varian for the TrueBeam linac. The results were compared with experimental data provided by five European radiotherapy centers. It was concluded that the latent variance and the accuracy of the phase-space files were adequate for the routine clinical practice. However, for research purposes where low statistical uncertainties are required the phase-space files are not large enough.

To the best of our knowledge PRIMO is the only fully Monte Carlo-based linac and dose simulation system, addressed to research and dose verification, that does not require coding tasks from end users and is publicly available.

PRIMO DISTRIBUTION

---

PRIMO is freely distributed through the site <http://www.primoproject.net>. The first release of the code was on September 2013. The current version is beta 0.1.5.1307. Up until now PRIMO has been downloaded more than 950 times by about 500 different users from nearly 450 institutions around the world. The countries with more access activity to the PRIMO project website are Spain, Unites States and Germany. Currently, there are users from 70 European hospitals (30 in Spain and 20 in Germany), 85 universities and 5 national regulatory commissions.

PRIMO is also being used for education. Currently, the Medical Physics program of the European Master on Nuclear Applications of the Technical University Aachen at Jülich includes the use of PRIMO for teaching Monte Carlo planning in radiotherapy. Master and undergraduate theses have been, or are being, carried out using PRIMO at the Universitätsklinikum Gießen und Marburg, Universitätsmedizin Mainz, McGill University, Universität Duisburg-Essen and Universitat Politècnica de Catalunya. Doctoral theses are in progress at the Universitätsklinikum Essen for clinical validation of PRIMO and at the Universidad de Granada for its application on the absolute dosimetry of small fields for the treatment of retinoblastomas.

We have collaborations on scientific applications of PRIMO with research groups from Istituto Clinico Humanitas, Universitätsklinikum Aachen, the Food and Drug Administration, Universidad de Granada and Karolinska Institutet and we have given support on the usage of the code to some research groups at Université Catholique de Louvain, Universitätsmedizin Mainz, Hospital Universitari Vall d'Hebron and McGill University, to name a few.

In the framework of the project FIS2012-38480 of the Spanish Ministerio de Economía y Competitividad PRIMO is being used for the study of the response of ionization chambers in small fields aimed at the development of a new dosimetry protocol.

Further development of PRIMO, from 2015 until 2018, is funded through the Deutsche Forschungsgemeinschaft project BR 4043/3-1.



GENERATION OF LINAC GEOMETRIES

---

A modified version of the `PENEASYLINAC` code is used in this thesis to create the linac geometries. `PENEASYLINAC` uses a database of linac components coded according to the syntax employed by `PENGEOM`. These components are combined to produce the geometry of a given linac in a given configuration (mode, energy, field size, MLC type and position, electron applicator size). It also generates the `PENEASY` input files specifying the source, tallies, VRT and transport parameters.

The original `PENEASYLINAC` program had support for the low- and high-energy models of Varian linac. These two models encompass the commercial Clinac 600C, 600C/D, Unique, and the C-Series. In this thesis, the geometries of the Elekta linac series SLi and MLCi were generated and incorporated into the `PENEASYLINAC` library. The geometries include elements for tallying phase spaces and for implementing variance-reduction techniques. Elekta geometries were generated using the `PENLINAC` code, which is described in the article included in this appendix.

## PENLINAC: extending the capabilities of the Monte Carlo code PENELOPE for the simulation of therapeutic beams

**M L Rodríguez**

Centro Médico Paitilla. Calle 53 y ave Balboa. Paitilla. Panamá, Republic of Panama E-

mail: [milrocas@gmail.com](mailto:milrocas@gmail.com)

Received 17 March 2008, in final form 24 June 2008

Published 4 August 2008

Online at [stacks.iop.org/PMB/53/4573](http://stacks.iop.org/PMB/53/4573)

### Abstract

In this work we present PENLINAC, a code package developed to facilitate the use of the Monte Carlo code PENELOPE for the simulation of therapeutic beams, including high-energy electrons, photons and  $^{60}\text{Co}$  beams. The code simplifies the creation of the treatment machine geometry, allowing the modeling of their components from elementary geometric bodies and their further conversion to the quadric functions-based structure handled by PENELOPE. The code is implemented in various subroutines that allow the user to handle several models of radiation sources and phase spaces. The phase spaces are not part of the geometry and can store many variables of the particle in a relatively small data space. The set of subroutines does not alter the PENELOPE algorithms; thus, the main program implemented by the user can maintain its kind-of-particle-independent structure. A support program can handle and analyze the phase spaces to generate, among others, last interaction maps and probability distributions that can be used as sources in simulation. Results from simulations of a Clinac linear accelerator head are presented in order to demonstrate the package capabilities. Dose distributions calculated in a water phantom for a variety of beams of this accelerator showed good agreement with measurements.

### ATTENTION ;

Pages 137 to 156 of the thesis, containing the article, are available at the editor's web

<http://iopscience.iop.org/article/10.1088/0031-9155/53/17/008>

## GLASS CODE

PRIMO was conceived with a layered structure. This is a convenient approach since it allows to combine previous codes developed by the authors, namely `PENEASYLINAC` for the creation of linac geometries and `PENEASY`, a main program for steering the `PENELOPE` subroutines. An additional benefit of the layered structure is that it facilitates code maintenance. The topmost layer, named the graphical layer for the automation of the simulation system (`GLASS`), was the one fully developed in this thesis. It controls the data flow and functioning of the lower layers (as illustrated in figure 7) and provides PRIMO with the following functions: setting up and controlling the Monte Carlo simulation of linear accelerator beams; configuring the radiation fields; delimiting and importing contours of structures on medical images; setting up and controlling the Monte Carlo simulation for dose calculation in the patient; importing, visualizing and processing of phase-space files; visualization and rendering of medical images; visualization of dose distributions and dose-volume histograms and processing of dose distributions. `GLASS` contains the graphical user interface (GUI) which was conceived with the spirit of balancing user abstraction from the underlying radiation transport intricacies and wide access to simulation data.

## C.1 SOFTWARE TOOLS AND TECHNIQUES

`GLASS` is programmed entirely in Delphi. Delphi is an object-oriented Pascal language distributed by Embarcadero Technologies Inc. The Delphi Builder Professional, version 2010 IDE is used for visual programming, code editing and debugging. Recently, Delphi Builder was upgraded to the version XE8 and some of its new capabilities were incorporated into the code. Chart graphs are programmed using the Steema Software TeeChart Pro Package. The compiled program is self-contained and does not require for its execution of any other proprietary DLL or third-party library than those provided with the Windows Operating System (OS). The requirements for PRIMO installation are provided in the manual (see appendix D).

Bug tracking and To-Do-List of PRIMO are performed using BugTrac, a web-based bug tracking system installed in the Argos cluster at the the Institut de Tècniques Energètiques, Universitat Politècnica de Catalunya. It is used to centralize records of known issues, including details on how to reproduce them, the programming steps

*Delphi**TeeChart**bug tracking*

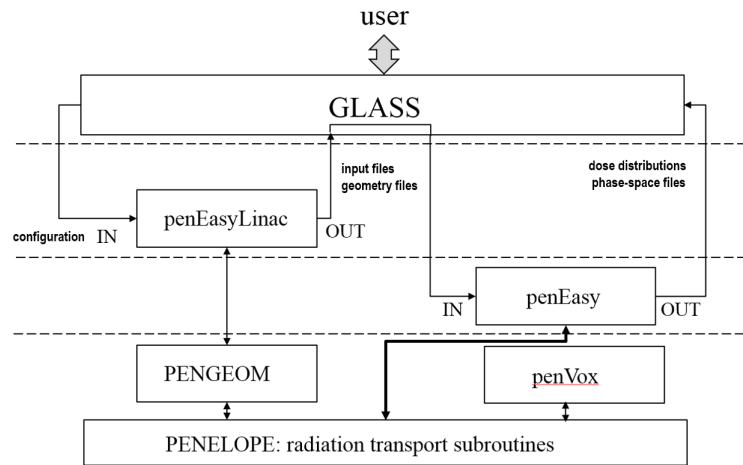


Figure 7: Data flow among the layered components of PRIMO.

to fix them and the person that should do the programming work. It also keeps track of improvements planned for future versions.

*Git*

Versions of the GLASS source code are maintained using Git, a very robust distributed version control system. The functions of Git can be accessed from the Delphi IDE in its last version XE8. A complete PRIMO source repository is maintained at the site <http://bitbucket.org>. Both, the BugTrac system and the PRIMO source repository are not public, access is only granted to PRIMO authors.

## C.2 ALGORITHMS

*object-oriented programming*

Delphi data abstraction involves the definition of classes. Briefly, a class (*e.g.*, a circle) has properties (*e.g.*, a radius) and functions (*e.g.*, to draw the circle in the screen), can change its state (*e.g.*, the circle can change its radius), and can own other classes. In OOL nomenclature, the term object refers to an instance of a class. Delphi supports inheritance and polymorphism. Inheritance is the capability of descendant classes to inherit the properties and functionality of ancestor classes (*e.g.*, a shape is an ancestor class of a circle). Polymorphism allows the substitution of one type of class by other related class in the same or similar role (*e.g.*, a circle can be passed as a parameter to a function that requires a shape). It allows to handle objects of different classes in a uniform manner. In this appendix some of the algorithms implemented in GLASS are explained by describing the classes and their functions in the code.

*The simulation project*

*TProject object*

The *TProject* class represents the simulation project. The project class

Listing 1: A simplified version of the *TProject.Textify* method.

```

1 procedure TProject.Textify(var Lines: TStringList);
  { This method streams the project data to a text file.
    Lines is a list of strings (object)
    Notes is also a list of strings
  }
6 var
  i: integer;
  filename: string;
  begin
    filename := Path + '\'+Id+'.ppj'
11 Lines.Clear;
    {tag the start}
    Lines.Add('PRIMOPRstart');
    { textify project id data}
    Lines.Add('PROJECTid='+Id);
16 Lines.Add('PROJECTname='+Name);
    Lines.Add('PROJECTpath='+Path);
    { textify the project notes (line by line)}
    for i:= 0 to Notes.Count-1 do
      Lines.Add('PROJECTnotes('+IntToStr(I)+'='+Notes.Strings[i])
21 ;
    {each object textify its own data}
    Linac.Textify(Lines);
    Patient.Textify(Lines);
    SimData.Textify(Lines);
    {tag the end}
26 Lines.Add('PRIMOPRend');
    {save the project}
    Lines.SaveToFile(filename);
  end;

```

owns a *Linac* class and a *Patient* class. Their properties include transport parameter tables, parameters for setting the variance-reduction techniques and others. This class is represented graphically in figure 8. The *TProject* class and its owned classes are all descendants of the *TTextStream* class. The main purpose of *TTextStream* is the streaming of text files employing two Pascal procedures, *Textify* and *BuildFromText*. Each descendant of *TTextStream* implements its own version of these two procedures to save/read its own attributes to/from disk. The procedure *TProject.Textify* is shown in listing 1.

#### *Volumetric data*

The *TVolume* class is the common ancestor of classes representing CT volumes, 3D dose distributions and structures. This is illustrated in figure 9. *TVolume* is a 3D array of bins. Each bin has a position and a

*TVolume* object



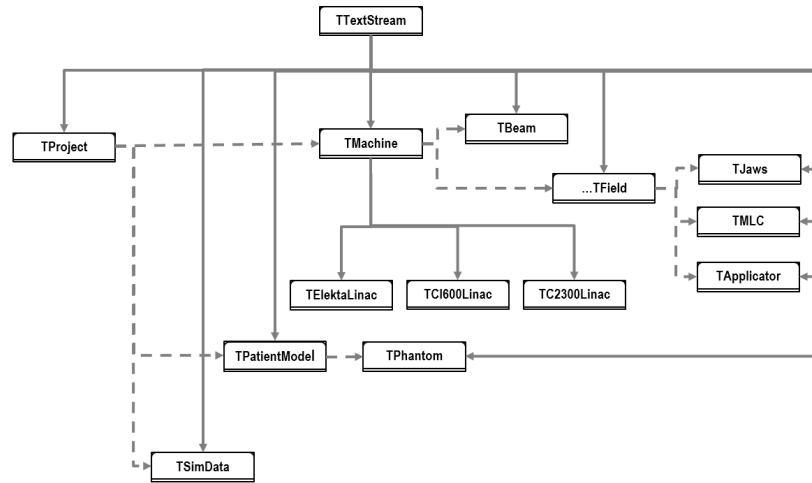


Figure 8: The *TProject* object and its related objects. Discontinuous lines represent an ownership relation, continuous lines represent inheritance

size in a Cartesian coordinate system. The position  $(x_n, y_n, z_n)$  of the  $n$ -th bin in the array is defined by the position of the first bin  $(x_{min}, y_{min}, z_{min})$  as follows,

$$\begin{aligned}
 x_n &= x_{min} + i \delta_x \\
 y_n &= y_{min} + j \delta_y \\
 z_n &= z_{min} + k \delta_z,
 \end{aligned} \tag{6}$$

where  $(i, j, k)$  are the indices (starting at zero) of the array and  $\delta_x$ ,  $\delta_y$  and  $\delta_z$  are the bin sizes in each coordinate direction. The array is organized such that  $x$  runs first, then  $y$  and then  $z$ .

For a dose distribution scored in the geometry of a water phantom, the values of  $(x_{min}, y_{min}, z_{min})$  and  $(\delta_x, \delta_y, \delta_z)$  are defined by the user. For a CT volume they are decoded from the DICOM file when the CT is imported. The format of the datum stored in the array element (bin) is variable. It is a double-precision (8-byte) floating point number for a dose distribution, a 4-byte integer for a CT and an 8-byte integer for an array of structures. Thus, a polymorphic pointer as the one shown in listing 2 is used to access the bin value.

One of the functions of the *TVolume* class is to calculate the value of the dose at an arbitrary point by trilinear interpolation [115]. The

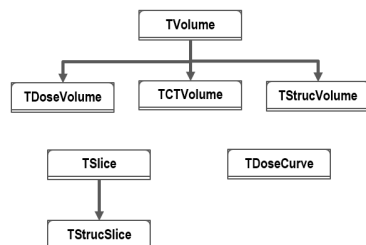


Figure 9: Objects representing a volume (3D matrix) of data.

Listing 2: Polymorphic pointer used in the *TVolume* object to address a data element of the 3D matrix.

```

1  pIntArray=array[0..0] of integer;
   pFloatArray=array[0..0] of double;
   pInt64Array=array[0..0] of Int64;
   PolyPointer=packed record
6   case integer of
     0: ( p : pointer);      {generic pointer}
     1: ( pi : pIntArray);   {pointer to 32 bits integers}
     2: ( pf : pFloatArray); {pointer to double-precision floating
                             -point numbers}
     3: ( pc : pInt64Array); {pointer to 64-bits integers}
   end;

```

interpolated value  $d_{xyz}$  at an arbitrary point  $(x,y,z)$  is calculated (see figure 10) as follows

$$\begin{aligned}
 d_{x00} &= d_{000} + f_x(d_{100} - d_{000}) \\
 d_{x01} &= d_{001} + f_x(d_{101} - d_{001}) \\
 d_{x10} &= d_{010} + f_x(d_{110} - d_{010}) \\
 d_{x11} &= d_{011} + f_x(d_{111} - d_{011}) \\
 d_{xy0} &= d_{x00} + f_y(d_{x10} - d_{x00}) \\
 d_{xy1} &= d_{x01} + f_y(d_{x11} - d_{x01}) \\
 d_{xyz} &= d_{xy0} + f_z(d_{xy1} - d_{xy0}), \tag{7}
 \end{aligned}$$

where the  $d$ 's represent the values of the dose at the considered point, see figure 10,  $f_x$ ,  $f_y$  and  $f_z$  are the relative distances of the point with respect to the starting point  $(x_0, y_0, z_0)$  in the  $x$ ,  $y$  and  $z$  directions, respectively, that is,

$$\begin{aligned}
 f_x &= \frac{x - x_0}{\delta_x} \\
 f_y &= \frac{y - y_0}{\delta_y} \\
 f_z &= \frac{z - z_0}{\delta_z}. \tag{8}
 \end{aligned}$$

*TSlice object*

Another function of the *TVolume* class is to create slices of the volumetric data. These slices, in the case of a CT volume, produce images of the transverse (or axial), sagittal and coronal planes of the human body. Slices are encapsulated in the *TSlice* class. A slice has many functions, most of them with visualization purposes. A slice can be enlarged or shrunk employing bilinear interpolation to accommodate its size to the size of the displaying window or to zoom a region. The efficiency of the algorithm that performs this function is crucial for a fast response when a CT volume or a dose distribution is displayed interactively. The method *TSlice.Grow* implements the fast interpolation algorithm in listing 3. The visualization of slices has been implemented more efficiently with a series of classes developed using the functions of the OpenGL library. OpenGL is an API that exploits the capabilities of the GPUs included in most graphic cards [110].

#### *Dose distributions*

*TDoseVolume object*

The *TDoseVolume* class is a descendant of *TVolume*. It represents either a dose distribution or a distribution of dose uncertainties. Among the functions of this class there is one to combine several dose distributions. In multi-field simulations one dose distribution is created per field. The total dose is obtained for analysis and visualization. User-defined weight factors can be assigned to fields and used to combine

Listing 3: Fast bilinear interpolation of a 2D matrix

```

procedure Fast2DInterpol(poData, pdData: PFloatArray; oW, oH, dW,
  dH: Integer; InverseW, InverseH: Boolean);
{
  Fast bilinear interpolation of a 2D matrix. The matrix is
  zoomed in/out from an original size (width x height) in
  pixels oWxoH to a size dWxdH.
  poData and pdData are pointers to the origin and destination
  matrices, respectively. Memory must have been allocated.
5  If the logical variables InverseW and/or InverseH are set true,
  interpolation is carried out from right to left and/or from
  bottom to top.
}
var
  i, j, jOrg, iOrg: integer;
  jstep, istep, rowstep, colstep: integer;
10  oxfirst, oxlast, oyfirst, oylast: integer;
  dxfirst, dxlast, dyfirst, dylast: integer;
  dy, dx, xRatio, yRatio, yOrg : double;
  xOrg, val00, val01, val10, val11 : double;
  yVal1, yVal2, intVal: double;
15  yoOffs, ydOffs, oOffs: longint;
begin
  xRatio:= dW/oW;
  yRatio:= dH/oH;
  oxfirst:= 0; oxlast:= oW-1;
20  oyfirst:= 0; oylast:= oH-1;
  dxfirst:= 0; dxlast:= dW-1;
  dyfirst:= 0; dylast:= dH-1;
  if InverseW then
  begin
25    istep:= -1;
    iswap(oxfirst, oxlast); {swap values}
    iswap(dxfirst, dxlast);
  end else istep:= 1;
  if InverseH then
30  begin
    jstep:= -1;
    iswap(oyfirst, oylast);
    iswap(dyfirst, dylast);
  end else jstep:= 1;
35  rowstep:= jstep*oW;

```

Listing 4: listing 3 continued

```

{ interpolation starts here}
j:= dyfirst;
while (j>=0) and (j<dH) do
begin
5   yOrg:= j/yRatio;
   jOrg:= Trunc(yOrg);
   dy:= yOrg-jOrg;
   if jOrg = oylast then
begin
10   jOrg:= oylast-jstep;
   dy:= 1.0e0
end;
   if InverseH then dy:= 1.0e0-dy;
   yoOffs:= jOrg*oW;
15   ydOffs:= j*dW;
   i:= dxfirst;
   while (i>=0) and (i<dW) do
begin
20   xOrg:= i/xRatio;
   iOrg:= Trunc(xOrg);
   dx:= xOrg-iOrg;
   if iOrg=oxlast then colstep:= 0 else colstep:= istep;
   if InverseW then dx:= 1.0e0-dx;
   oOffs:= yoOffs+iOrg;
25   val00:= poData^[0offs];
   val01:= poData^[0offs+colstep];
   val10:= poData^[0offs+rowstep];
   val11:= poData^[0offs+rowstep+colstep];
   yVal1:= val00+(val10-val00)*dy;
30   yVal2:= val01+(val11-val01)*dy;
   intVal:= yVal1+(yVal2-yVal1)*dx;
   pdData^[ydOffs+i]:= intVal;
   colstep:= istep;
   Inc(i,istep)
35   end;
   Inc(j,jstep);
end;
end;

```

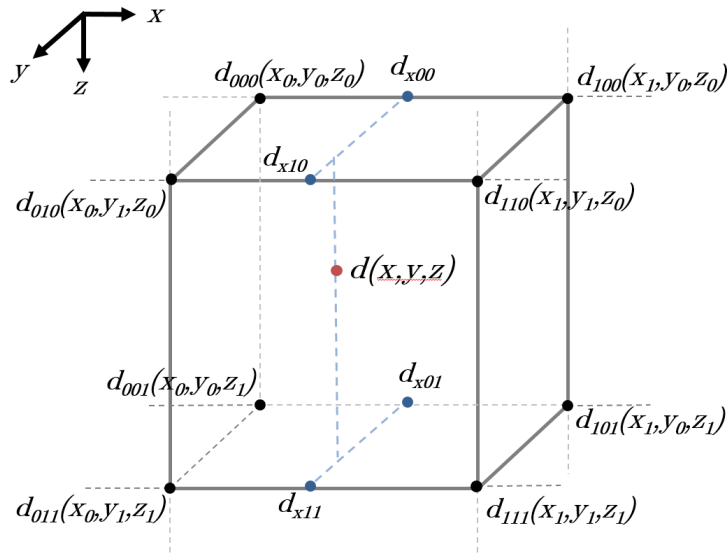


Figure 10: Trilinear interpolation. See text for details.

several dose distributions and their corresponding uncertainties. The total dose  $d_i$  in a bin  $i$  is calculated as

$$d_i = \frac{1}{N} \sum_{f=1}^F d_{if} n_f w_f, \quad (9)$$

where  $d_{if}$  is the dose per history in the  $i$ -th bin for the field  $f$ ,  $N$  is the total number of simulated histories,  $n_f$  is the total number of simulated histories for the field  $f$ ,  $F$  is the number of fields and  $w_f$  is the weight factor for the field  $f$ . The standard statistical uncertainty of  $d_i$  is evaluated as

$$\sigma_i = \frac{1}{N} \sqrt{\sum_{f=1}^F (\sigma_{if} n_f w_f)^2}, \quad (10)$$

where  $\sigma_{if}$  is the uncertainty of the dose at the bin  $i$  for the field  $f$ . Weight factors  $w_f$  are normalized to unity, that is,

$$\sum_{f=1}^F w_f = 1. \quad (11)$$

### CT structures

Contours of structures can be drawn interactively in an interface object (window) or they can be imported from a DICOM-STRUCT file. In both cases, a contour is specified as a set of “connected” points. These points must be defined in the coordinate system of the CT.

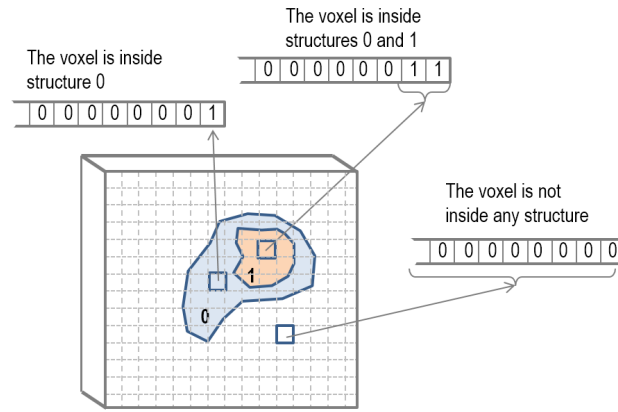


Figure 11: A set of regions of interests are represented in an array of 64-bits unsigned integers elements. Each bit of the data element indicates whether or not the corresponding array element in the CT volume is inside the ROI.

The CT and DICOM-STRUCT files are imported separately. The CT must be imported first. So, when the contours are imported a unique identifier that connects both files is checked. The coordinate system specified in both files is also verified for coincidence.

Contours delimit regions of interest (ROIs) (organs or structures) in the CT volume. The function of the *TStrucVolume* class is to handle all the ROI of a CT. This object has exactly the same attributes of the *TCTVolume* that encapsulates the CT, except for the type of datum stored in the array element, which in this case is a 64-bit unsigned integer. Each element of the array in a *TStrucVolume* class is mapped onto the same element (the one with the same indices) in the *TCTVolume* array. Each bit of the 64-bit integer identifies a ROI (numbered from 0 to 63). If the bit is set (equal to 1), it means that the corresponding bin in the CT pertains to the ROI, as illustrated in figure 11.

The creation of a ROI s is done sequentially on each slice of the CT as follows,

1. A map consisting of an array with the same dimensions of the slice is created and filled with zeros.
2. The points of the contour are written onto the map by setting their corresponding elements to 1.
3. A modified Bresenham's algorithm [18] is used to join the points with straight lines, thus setting to 1 all the map elements along the lines.
4. A point inside the closed contour is identified by applying a search algorithm that uses the centroid of the contour as a starting point.

*TCTVolume* and  
*TStrucVolume*  
objects

region of interest  
creation algorithm

5. A flood-fill algorithm is used to set to 1 on the map the points that are internal to the contour. Flood-filling starts at the point identified in the previous step.
6. The map is transferred to the 3D array of the *TStructVolume* by setting the *s-th* bit in all those bins with a corresponding map element equal to 1.

The original points of the contour are not kept in the *TStructVolume* class. For this reason they have to be recalculated when required, *e. g.*, for visualization. This is done by the simple function depicted in listing 5 in which a bin is determined to be in the ROI contour if, at least one of its adjacent bins does not pertain to the ROI.

For contour visualization, the map of bits is interpolated employing an algorithm similar to the one in listing 3 and the contour is extracted with the function in listing 5.

The approach used to create the ROI has the main drawback of reducing the spatial resolution of the contours to that of the CT volume. This degradation of the resolution occurs when the positions of the original contour points are truncated in the steps 2 and 3 of the algorithm. Better results for ROI creation, visualization and other operations, like calculation of the absolute CT volume covered by the ROI, can be obtained using shape-based interpolation [51, 93].

#### *Digitally-reconstructed radiograph (DRR)*

DRRs are created by a volume rendering operation in which the CT is projected onto a 2D map. The purpose is to create an image that resembles a conventional radiograph. Images produced by this process are, in all cases, a degraded version of the actual radiograph, due mainly to the substitution of the real patient by the CT. Several volume rendering algorithms can be applied to a DRR construction, the most straightforward method is by using a ray casting algorithm [111, 109]. Ray casting is a simplification of light ray tracing as it only calculates the intersection of the ray with the object in the scene without considering lighting and reflections. Evaluating the intersections of the ray with the CT voxel allows to calculate a total attenuation factor which is the value stored in the DRR's pixel.

DRRs are used in PRIMO for guidance in the setting of the radiation fields in a perspective called the BEV in which the patient is seen from an observer situated at the linac target. The DRR is created by the procedure *CreateDRR* which is a method of the visual object that manages the user's interface for field setting. This subroutine uses a ray casting algorithm that incorporates some elements of the algorithm of Siddon [111]. The Cartesian coordinate system is defined



Listing 5: A function to identify that a pixel is in the contour of a ROI.

```

function IsContour(x,y,s: Integer): Boolean;
2  {
   This function returns true if the pixel with indices x,y
   is in the contour of the structure with number s.
   The function IsInROI, returns true if the pixel is in the ROI.
   The variables w and h are attributes of the class TStrucSlice
7  and are the width and height (in pixels) of the bit map of the
   slice, respectively.
   }
var
   res: Boolean;
12
   procedure check(i,j);
   begin
       res:=res or not IsInROI(i,j,s);
   end;
17
begin
   res:= false;
   if not check(x,y) then
   begin
22       result:= res;
       exit;
   end;
   if (x-1)>=0 then check(x-1,y,s);
   if (y-1)>=0 then check(x,y-1,s));
27   if (x+1)< w then check(x+1,y,s));
   if (y+1)< h then check(x,y+1,s);
   if ((x-1)>=0) and((y-1)>=0) then check(x-1,y-1,s);
   if ((x+1)<w)and((y-1)>=0) then check(x+1,y-1,s);
   if ((x+1)<w)and((y+1)<h)then check(x+1,y+1,s);
32   if ((x-1)>=0) and((y-1)>=0) then check(x-1,y-1,s));
   result:= res;
end;

```

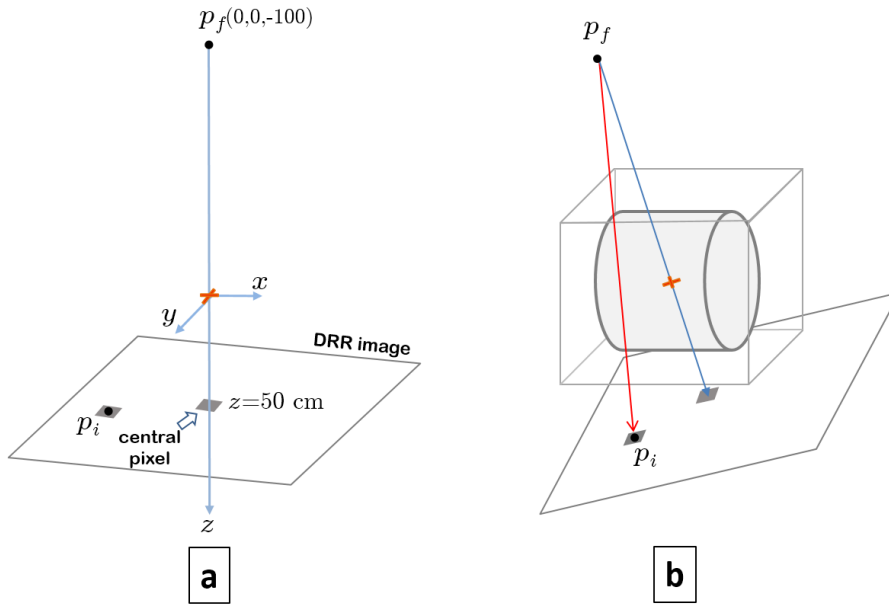


Figure 12: Geometry for the construction of a DRR. The focus point  $p_f$ , representing the source of X-rays, is at 100 cm from the isocenter which is the origin of coordinates. The CAX of the linac coincides with the z-axis. A ray (in red) is cast from the point  $p_f$  to the point  $p_i$  and its total attenuation is calculated and stored in the DRR's pixel.

conveniently with its origin at the position of the isocenter<sup>1</sup> of the linac in the patient. Rays are cast in a perspective projection with the starting point of the ray or 'focus' situated at the target of the linac and the final point at the position of the pixel whose gray level is computed, as represented in figure 12a.

The DRR image is defined as a *TSlice* object. The plane of the image is assumed to be at 150 cm from the focus and its total size is made dependent on a zoom factor set by the user. Varying this factor, the image can represent projected areas in the range of  $30 \times 30 \text{ cm}^2$  to  $90 \times 90 \text{ cm}^2$ . The image is positioned such that the ray along the CAX projects on its central pixel. The size of the image pixel is determined from the size of the projected CT voxel divided by a factor  $\delta \geq 1$ . In the implemented algorithm this factor determines the density of rays cast through the CT volume and hence the degree of the object's 'detail' achieved in the DRR image. A large value of  $\delta$  will, in principle, produce a highly 'detailed' image, however there is a limit determined by the spatial resolution of the CT. Increasing  $\delta$  will proportionally increase the time of computation of the DRR. The algorithm uses a fixed value of  $\delta = 2$  which is considered, in general, a good compromise between image quality and computing time.

*digitally-  
reconstructed  
radiograph creation  
algorithm*

<sup>1</sup> This is a convenient convention as the isocenter is the point of intersection of the axes with respect to which the linac's arm and the couch rotate.

Ray casting is performed with the following algorithm (see figure 12b):

1. A rotation of the focus point  $p_f$  with respect to the reference system displayed in figure 12 is performed with Euler angles ( $\omega=0, \theta=-G, \phi=T$ ), where  $G$  and  $T$  are the angles of rotation of the gantry and the couch, respectively.
2. The rotated focus is translated with the displacements of the isocenter in the CT.
3. For each pixel in the image:
  - The point  $P_i(x, y, z = 50)$  at the center of the pixel is transformed by a rotation and translation with the same Euler angles and displacements employed to transform the focus.
  - The ray is defined as the vector  $\vec{R}$  with origin at the point  $P_f$  and directed to  $P_i$ . The length of  $\vec{R}$  is incremented in steps of a given length as the ray is cast. The intercections with the voxel of the CT are calculated at each step.
  - The algorithm of Siddon [111] is used to determine the first point of intersection of the ray with the CT. Beyond this point, the ray is cast at steps of length  $s$  equal to the minimum spatial resolution of the CT on any coordinate direction. The HU value  $H_i$  at the position of the vector  $\vec{R}$  is calculated at every step by trilinear interpolation in the CT volume.
  - the attenuation factor  $t$  of the ray is calculated as,

$$t = \frac{1}{1000} \mu_w \sum_{i=1}^n s_i o_i H_i, \quad (12)$$

where  $n$  is the number of steps of length  $s$  accomplished and  $\mu_w$  is the linear attenuation coefficient of water for a typical CT X-ray spectrum<sup>2</sup>. The factor  $o_i$  is an opacity factor used to arbitrarily reduce or reinforce the importance of certain tissues in the attenuation of the ray.

- The total attenuation of the ray  $T$  is calculated as,

$$T = 10^4 (1 - e^{-t}), \quad (13)$$

and it is stored in the pixel.

When CT structures are projected in the DRR image, the algorithm also determines the voxel intersected in the *TStrucVolume* object at

<sup>2</sup> In this equation it is assumed that the Hounsfield number of the water is represented as 1000 in the CT.

each step of the ray casting iteration. As it was explained in section C.2 each bit stored in one element of the 3D array of the *TStrucVolume* object represents a ROI. The bit is 1 or 0 if the corresponding CT voxel pertains or not to the ROI. For each voxel intersected in the CT, the ray casting algorithm obtains the value of the corresponding voxel in the *TStrucVolume* object and performs a bitwise OR operation of that value with all those obtained in previous steps to calculate the structures intersected by the ray. The result is stored in a separated image which is blended with the DRR for visualization.

The DRR creation algorithm is implemented in parallel, using the Delphi parallel-programming library included in the Delphi XE8 version. In the implementation, ray casting is distributed among all the available processors in a computer. The parallel version of the algorithm performs about 6-16 times faster than the sequential version in 8-24 cores computers.

### The gamma index

The gamma index is used for the comparison of two dose distributions and was first introduced by Low and co-workers [77]. The technique combines the dose difference and the distance-to-agreement (DTA) criteria to calculate an index that establishes the degree of agreement between a reference and a queried dose distributions. In PRIMO, gamma analysis is used to compare a measured dose profile with the simulated 3D dose distribution. The measured curve is considered the reference dose and the simulated distribution the queried dose.

Gamma analysis is implemented in an object of class *TGammaEngine*. This class is a descendant of *TProcessEngine* which is basically an abstraction for an arbitrary intensive and time consuming calculation process. The class has methods to initialize and to stop the calculation and also to arbitrate the computing resources according to the operative system rules. The abstract method<sup>3</sup> *RunProcess* encapsulates the calculation engine. The class owns a progress bar that is linked to the calculation engine to show the progress of the calculation.

*TProcessEngine*  
object

The *TGammaEngine* class implements the *RunProcess* method with an algorithm to calculate a gamma index  $\gamma_p$  for each point  $p$  in the reference curve. The gamma index  $\gamma_p$  at the point  $p$  is calculated as follows,

*TGammaEngine*  
object

$$\gamma_p = \min \left\{ \sqrt{\left(\frac{\Delta d_i}{\Delta D}\right)^2 + \left(\frac{\Delta s_i}{\Delta S}\right)^2} \right\}, \quad (14)$$

where  $\Delta D$  and  $\Delta S$  are arbitrary constants known as the acceptance criteria for the dose difference and for the DTA, respectively. The

<sup>3</sup> An abstract method is a subroutine which is not implemented in the class where it is declared but in the descendant classes.

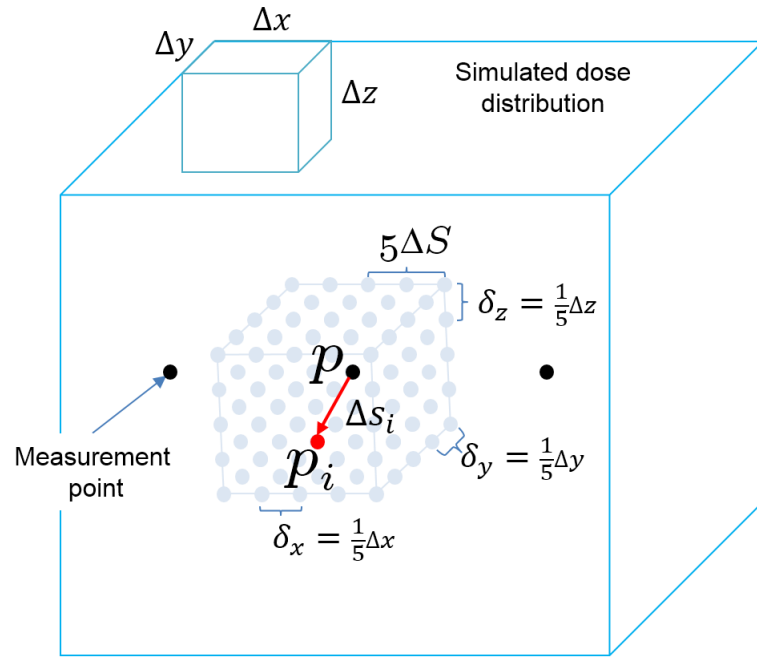


Figure 13: Calculation of the gamma index for a reference point  $p$ . See text for details.

term  $\Delta d_i$  is the difference between the reference dose at the point  $p$  and the queried dose at a point  $p_i$ . The term  $\Delta s_i$  is the distance between  $p$  and  $p_i$ . The minimum of the expression in curly braces is evaluated for the set of points  $\{p_i\}$ . The degree of agreement between the reference and the queried dose distribution is established based on the percentage of points with  $\gamma_p \leq 1$ .

In the algorithm implemented the set  $\{p_i\}$  is formed by points uniformly distributed in a cube of half-side  $d = 5\Delta S$  centered at the point  $p$ . The points are separated by distances  $\delta_x$ ,  $\delta_y$  and  $\delta_z$  equal to one-fifth of the bin size of the queried 3D dose distribution in the  $x$ ,  $y$  and  $z$  directions, respectively. It is illustrated in figure 13. The dose at each point  $p_i$  is calculated by trilinear interpolation in the simulated dose distribution.

#### *Particle processing pipeline*

The analysis of a phase-space file is a process in which a series of processing algorithms are applied to a large quantity of particle data stored in a file. Normally, the large size of the phase-space impedes to load the full file into memory for a fast access to the particle data.

*phase-space analysis  
process*

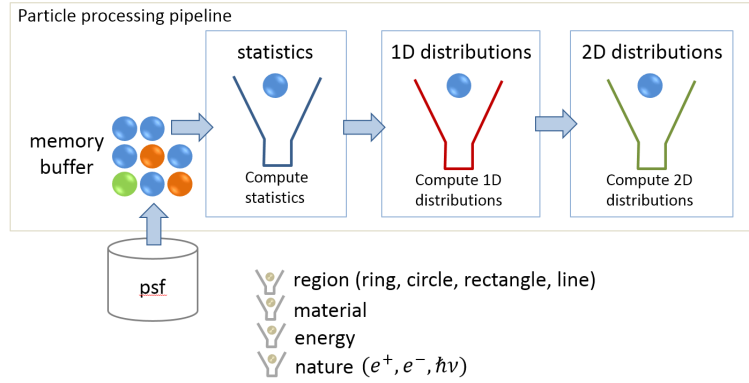


Figure 14: Particle processing pipeline. The particle is filtered at each processing step according to its nature, the material where it was created, its energy, flight direction and position.

Nevertheless, arranging the process in the form of a pipeline<sup>4</sup> offers opportunities for optimization<sup>5</sup>.

The pipeline algorithm implemented in PRIMO is depicted in figure 14. The particle is loaded once and submitted to a sequence of processing stages (or engines) that, in this case, use the particle data to produce statistics and to create probability distributions. The pipeline is represented in the code by the class *TPipeline*. The streaming of particles from the file is managed by the class *TPhaseSpace* owned by the *TPipeline* object. This class implements two streaming mechanisms, a sequential one in which particles are flowed one by one from the file and a buffered mechanism in which a large block of data –of many particles– is loaded on every access. In the latter, two buffers are used in a tandem connection, *i.e.*, a buffer is loaded while the other is transferring particles to the first engine in the pipeline. The buffered mechanism is the default one. The per-particle mechanism uses the subroutines of the IAEA library [24] for reading the particle from the file and it is employed only when memory for the buffers can not be allocated.

In the pipeline the particle is filtered before it is processed. The filtering is actually done by the pipeline object but employing the options activated by each engine. The attributes of the filter include an energy interval, a list of materials, a set of variables to define sub-regions on the plane of the phase space and an angular interval.

The engine objects are of the classes *TStats*, *TEngpd1* and *TEngpd2*, representing an engine to produce statistics, a 1D probability distribution and a 2D probability distribution, respectively. All those objects are descendant of the abstract class *TEng* and must have an implementation for the methods, *ActivateFilter*, *InitProcessing*, *ProcessParticle* and

*TPhaseSpace and  
TPipeline objects*

*filtering the particle  
variables in the  
pipeline*

*TEngpd objects*

4 A pipeline is a chain of processing elements, arranged so that the output of one element is the input of the next.

5 This process is also very suitable for parallelization.

*EndProcessing*. The method *ActivateFilter* is called by the pipeline before processing the particles at each stage, in this method the engine obtain access to the results of previous filtering of the particle data and can also activate its own filtering options. In the method *InitProcessing* the engine initializes its processing variables (*e.g.*, counters for statistics), and can also allocate memory for the probability distributions. The method *ProcessParticle* is the one called by the pipeline to process the particle if it has passed through the filter. In this method the engine makes the calculations (*e.g.*, populate the probability distributions). Finally, the method *EndProcessing* is called when all the particles have been processed. It is the opportunity for the engine to make global calculations, (*e.g.*, to perform normalizations) and memory cleaning. A simplified version of the pipeline code is shown in listing 6.

#### *Distributed simulation*

Code was developed to distribute a simulation among the available processors of a computer. It required writing functions for partitioning source phase-space files and integrating result phase spaces and dose distributions. In the algorithm several instances of the simulation program `PENEASY` are run as subtasks of `GLASS`, and the OS is let to deal with the distribution of these instances among the processors of the computer. Instances are launched such that their screen outputs are redirected (by the OS) to `GLASS` which works as a supervisor. Notwithstanding its relative simplicity, this scheme has demonstrated to be very robust and stable in many tests and computers, even in situations where the number of instances launched matches or exceeds<sup>6</sup> the maximum number of processors available. It also has the advantage with respect to using parallel computing libraries (*e.g.*, `MPI` and `openMP`) that the efficiency is proportional to the number of physical cores used. For instance, in a typical 8-cores personal computer, the combination of this technique and splitting roulette can boost the efficiency in about 360 times. That factor increases to over 1000 for a 24-cores workstation. The main limitation of this approach is that running whole instances of the simulation program requires more shared physical memory ( $\sim 0.5$  GB per instance) which may limit the number of simulation processes that can be simultaneously run. Furthermore, this scheme cannot be adapted to parallel hardware platforms such as graphic processing units (GPU) and many integrated core (MIC).

---

<sup>6</sup> when hyper-threading is activated

Listing 6: A simplified version of the method *Tpipeline.pipeall* that implements the processing of the particles stored in a phase-space file.

```

1 procedure Tpipeline.pipeall;
  { This method processes all the particles in a psf.
    engs[0..nengs-1] is the array of processing engines (objects).
    psf is the object that streams particles from the file.
  }
6  var
    nparticles: double;
    i: integer;

    function pipe(all: double): double;
11  var
    p: double;
    s: integer;
    begin
16    p:= 0e0;
    while (p<all) do begin
      if not psf.getnextparticle then break;
      for s:=0 to nengs-1 do
        engs[s].ProcessParticle(psf.curparticle,FiltParticle(s));
        p:= p+1e0;
21    end;
    result:= p;
    end;

26  begin
    with psf.particles do
      nparticles:= electrons + photons + positrons;
      for i:= 0 to nengs-1 do
        engs[i].InitProcessing;
        nparticles:= pipe(nparticles);
31    for i:= 0 to nengs-1 do
      engs[i].EndProcessing(nparticles);
    end;

```



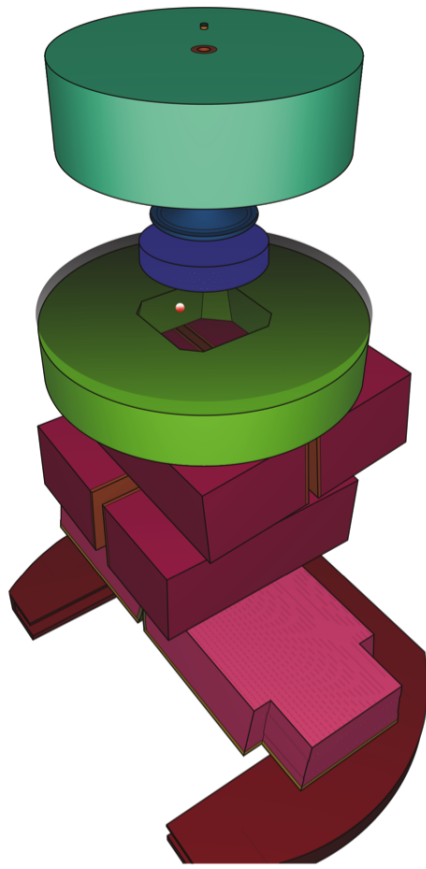


# D

PRIMO: USER'S MANUAL AND QUICK START  
GUIDE

---

# PRIMO User's Manual



Brualla · Rodriguez · Sempau

Copyright © 2010–2015 L. Brualla, M. Rodriguez, J. Sempau

PRIMO USER'S MANUAL  
SOFTWARE VERSION 0.1.5.1300

WWW.PRIMOPROJECT.NET

About the authors:

Lorenzo Brualla  
*Universität Duisburg-Essen*  
*Universitätsklinikum Essen*  
*NCTeam, Strahlenklinik*  
*Hufelandstraße 55, D-45122 Essen, Germany*

Miguel Rodriguez and Josep Sempau  
*Universitat Politècnica de Catalunya*  
*Institut de Tècniques Energètiques*  
*Diagonal 647, E-08028 Barcelona, Spain*

lorenzo.brualla@uni-duisburg-essen.de  
miguel.lazaro.rodriguez@upc.es  
josep.sempau@upc.es

Typeset by the authors with the L<sup>A</sup>T<sub>E</sub>X Documentation System. The *Legrand Orange Book* template downloaded from <http://www.LaTeXTemplates.com> was used to write this manual.

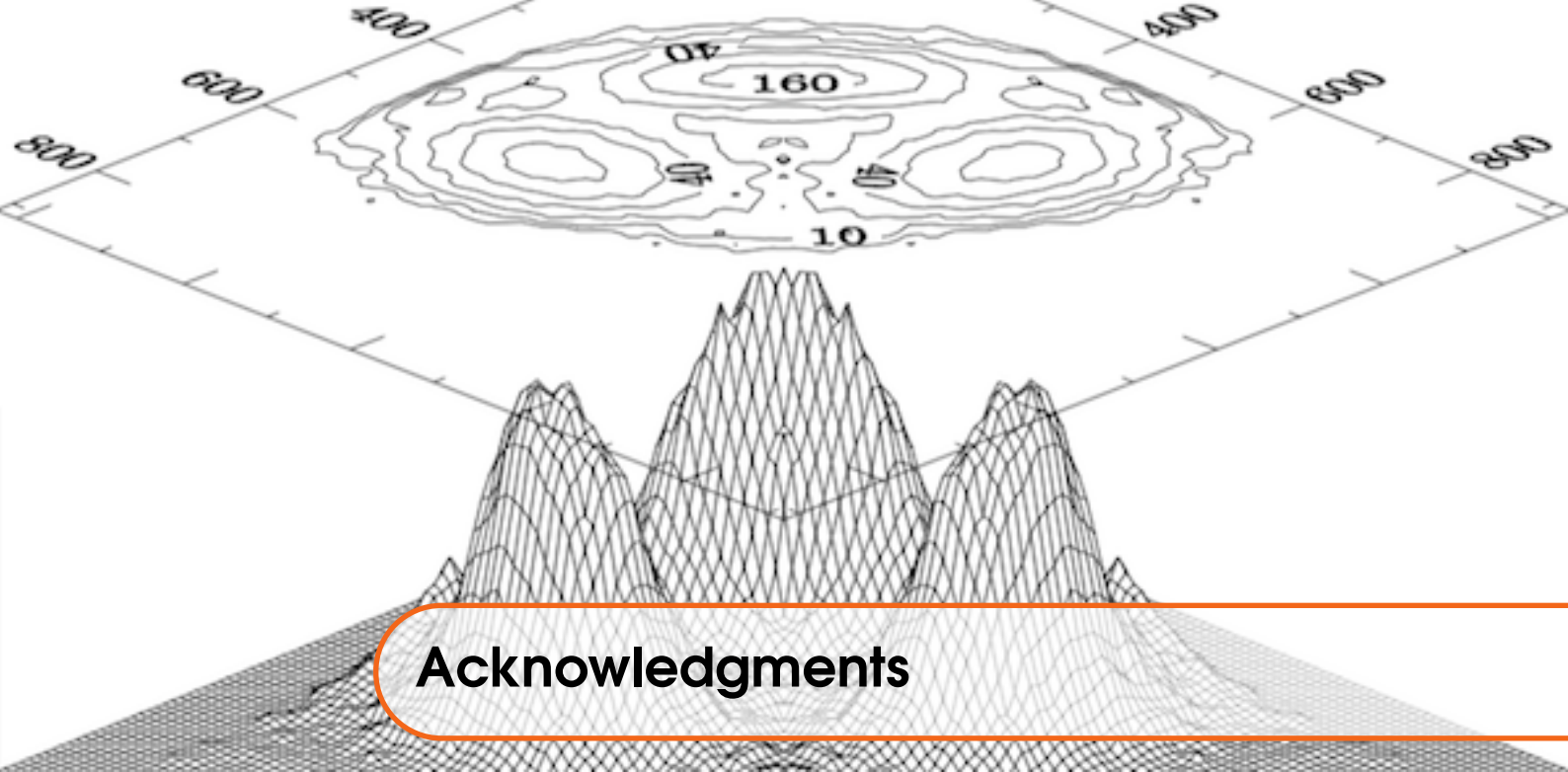
24<sup>th</sup> April 2015

## Disclaimer and Copyright

PRIMO is furnished 'as is'. No warranties, express or implied, that this software is free of error, or is consistent with any particular standard of merchantability, or that it will meet your requirements for any particular application, is made. No responsibility for any mathematical or technical limitations of the procedures and functions which make up this software is accepted. This software should not be relied on for solving a problem whose incorrect solution could result in injury to a person or loss of property. The authors of PRIMO shall not in any event be liable for any damages, whether direct or indirect, special or general, consequential or incidental, arising from use of this software. Your use of this software is entirely at your own risk.

This software is sole property of its authors. Permission to use this software for any purpose is hereby granted without fee. This software, or any part of it, cannot be sold, modified or re-distributed unless a written consent from the authors is obtained. Reverse-engineering is forbidden on any of the distributed files and on the files generated during the execution of the code.





## Acknowledgments

PRIMO is the result of many years of work. In that time several persons and organizations have provided us with their help and support. We are thankful to all of them. In particular, we are grateful to Wolfgang Sauerwein (Universitätsklinikum Essen, Germany) for his continued support and efforts that have contributed to make feasible the distribution of PRIMO. Luca Cozzi, Antonella Fogliata and Francesca Belosi (Istituto Oncologico della Svizzera Italiana, Switzerland) have put in a tremendous amount of work in beta-testing the code. Josep Pujal (Universitat Politècnica de Catalunya, Spain) and Michael Nieporte (Universitätsklinikum Essen, Germany) have taken the burden of all matters related to setting up computer clusters and information technology.

The Monte Carlo engine of PRIMO is PENELOPE. We express our gratitude to Francesc Salvat and José María Fernández-Varea (Universitat de Barcelona, Spain) not only for their work as PENELOPE authors, but also for the long-lasting close and rewarding collaboration that has allowed us to learn from their experience.

For validating the coded geometries we have had recourse to experimental data provided by the following persons: Luca Cozzi and Antonella Fogliata (Istituto Oncologico della Svizzera Italiana, Switzerland), Nuria Escobar-Corral (Universitätsklinikum Aachen, Germany), Andrea Flühs (Universitätsklinikum Essen, Germany), Dietmar Georg and Gabriele Kragl (Medizinische Universität Wien, Austria), Núria Jornet (Hospital de la Santa Creu i Sant Pau, Spain), Ricardo Palanco-Zamora (Karolinska Universitetssjukhuset, Sweden) and Klemens Zink (Universitätsklinikum Giessen und Marburg, Germany). We are appreciative of their help.

We gratefully acknowledge Varian Medical Systems International AG (Zug, Switzerland) and Elekta Limited (Crawley, United Kingdom) for authorizing the distribution of the encoded geometry files related to their respective linac models.

We thank the Deutsche Forschungsgemeinschaft (project BR 4043/3-1, Germany) and the Ministerio de Economía y Competitividad (project FIS2012-38480, Spain) for financial support.







# Contents

<b>1</b>	<b>Introduction</b> .....	<b>9</b>
1.1	Scope	10
1.2	Genesis	10
1.3	Webpage and resources	10
1.4	Version of PRIMO	11
1.5	List of citable references	11
<b>2</b>	<b>Installation</b> .....	<b>13</b>
2.1	Tested hardware and software	13
2.2	Whence to obtain PRIMO	13
2.3	How to install/uninstall PRIMO	14
2.3.1	Installer msi file .....	14
2.3.2	Uninstall .....	14
2.4	How to install the examples	14
2.5	List of examples	14
<b>3</b>	<b>Linac and absorbed dose simulation</b> .....	<b>15</b>
3.1	Linac simulation	15
3.2	Absorbed dose simulation	17
3.3	Histories and particles	17
3.4	Phase-space files	17
3.5	Statistical uncertainty	17
3.6	Variance-reduction techniques	18

<b>3.7</b>	<b>Simulation segments</b>	<b>18</b>
<b>4</b>	<b>PRIMO usage</b>	<b>21</b>
<b>4.1</b>	<b>Simulation setup</b>	<b>21</b>
4.1.1	New project	21
4.1.2	Beam, field and dose tally configuration	24
4.1.3	Material and density assignment	30
4.1.4	Contouring	32
4.1.5	Simulation configuration	34
4.1.6	Transport parameters	34
4.1.7	Variance reduction	35
4.1.8	Importing external phase-space files	37
4.1.9	Duplicating a project	38
<b>4.2</b>	<b>Execution</b>	<b>38</b>
4.2.1	Changing parameters at runtime	40
<b>4.3</b>	<b>Analysis of results</b>	<b>41</b>
4.3.1	Phase-space file analysis	41
4.3.2	Dose analysis	43
4.3.3	Changing the relative field weight	45
4.3.4	Inspecting the dose distribution	46
4.3.5	Comparing to experimental data	46
4.3.6	Dose comparison by gamma analysis	51
<b>5</b>	<b>Examples</b>	<b>53</b>
<b>5.1</b>	<b>Example 01: Photon reference field</b>	<b>53</b>
5.1.1	Simulation setup and execution	53
5.1.2	Analysis of results	59
<b>5.2</b>	<b>Example 02: Electron reference field</b>	<b>70</b>
5.2.1	Simulation setup and execution	70
5.2.2	Analysis of results	74
<b>5.3</b>	<b>Example 03: TrueBeam</b>	<b>74</b>
<b>5.4</b>	<b>Example 04: Dose calculation in a computerized tomography</b>	<b>77</b>
5.4.1	Simulation of segment s1	77
5.4.2	Field setup	77
5.4.3	Construction of the voxelized geometry	77
5.4.4	Dose calculation	82
	<b>Bibliography</b>	<b>83</b>
	<b>Books</b>	<b>83</b>
	<b>Articles</b>	<b>83</b>
	<b>Index</b>	<b>85</b>

## 1 — Introduction

The accurate Monte Carlo simulation of a linac requires a detailed description of its geometry and the application of variance-reduction techniques [JNR98]. The interpretation of linac blueprints and the coding of the geometry into the Monte Carlo system can be a tedious and error-prone task. The introduction of variance-reduction techniques, in turn, may require the modification of the computer code and this can involve a substantial programming effort by the end user [Bru12; Rey+07; SV13; SL08].

PRIMO is a program based on the codes PENELOPE 2011 [Bar+95; SFS11; Sem+97], PENEASY [SBB11], PENEASYLINAC [SBB11] and a graphical user interface that encompasses all these components in a single user-friendly environment. PENELOPE is a set of subroutines for the Monte Carlo simulation of coupled electron and photon transport. PENEASY is a general-purpose main program for PENELOPE that includes several source models, tallies, variance-reduction techniques and the possibility of combining quadric and voxelized geometries. PENEASYLINAC is a complementary tool that generates the input files required for the simulation of most Varian<sup>1</sup> and Elekta<sup>2</sup> linacs with PENELOPE/PENEASY. The Graphical Layer for the Automation of the Simulation System (GLASS) is a graphical user interface that allows users to define the configuration of the simulated machine, that is, irradiation mode, beam nominal energy, jaw positions, position of every leaf of the multileaf collimator (photon mode) or type of electron applicator (electron mode). All the other parameters, those of the simulation and of the applied variance-reduction techniques, are automatically selected by the system without user intervention. PRIMO incorporates graphical and numerical tools for the analysis of phase-space files and absorbed dose distributions tallied during the simulations. PRIMO can also import and simulate phase-space files written by other codes in the International Atomic Energy Agency (IAEA) binary format [Cap+06]. Dose distributions can be tallied in phantoms or computerized tomographies of patients.

In a nutshell, PRIMO is an automated, self-contained, fully Monte Carlo-based linac simulator and dose calculator with a user-friendly graphical interface.

<sup>1</sup>Varian Medical Systems Inc., California, USA

<sup>2</sup>Elekta AB, Sweden

## 1.1 Scope

PRIMO facilitates the Monte Carlo simulation with PENELOPE of most Varian and Elekta linacs and the estimation of the dose distribution in water phantoms and computerized tomographies. Knowledge of the Monte Carlo method, of programming, of the peculiarities of PENELOPE and of the physics of radiation transport is not necessary in order to set up, run and analyze the simulation of a linac and the subsequent dose distribution. Users of other Monte Carlo codes can also benefit from PRIMO thanks to the possibility of importing and simulating external phase-space files written in the IAEA format.

Owing to a number of specifically developed variance-reduction techniques [BSP09; BS10; RSB12; SBB11] PRIMO simulates linac geometries efficiently. Users with a multiple-core computer can reduce simulation time by automatically distributing the simulation among the available computing cores. Also, the code is capable of computing a dose distribution produced by a multiple-field irradiation. Most cases can be simulated in the time frame of one to three hours using an 8-core computer, obtaining a dose distribution within clinical requirements.

PRIMO performs the full Monte Carlo simulation of radiation transport from the primary electron source of a linac downstream to estimate the absorbed dose in a phantom or computerized tomography. This process uses the PENELOPE code as the computation engine. Therefore, PRIMO is based on one of the most accurate general-purpose Monte Carlo codes available [Fad+08; Fad+09; SF09; Sem+03].

Although PRIMO is mainly conceived as research software, it finds multiple applications in the daily clinical practice. For example, it can be used as an independent quality assurance tool. However, it must be stressed that PRIMO is not medical software and it does not have any certification or warranty. Please refer to the disclaimer and copyright statements for further details.

## 1.2 Genesis

Due to its layered software structure, PRIMO inherits important characteristics from the codes that constitute it. The components of PRIMO benefit from having been coded by a reduced number of developers. Additionally, these same components have been available for many years to a large number of users who have extensively tested them. PENELOPE, developed by F. Salvat, J.M. Fernández-Varea and J. Sempau, was first released in 1996. PENEASY, whose main author is J. Sempau, was first released in 2004. L. Brualla, the author of PENEASYLINAC, published its first version in 2009. The authors of PRIMO, L. Brualla, M. Rodriguez and J. Sempau, started to work on the GLASS that integrates all the aforementioned codes into PRIMO in 2010. The layered structure of PRIMO and the fact that all codes contained within are written by only five researchers facilitates the maintenance tasks and the development of new features.

## 1.3 Webpage and resources

**Notice 1.1** PRIMO is free software. However, PRIMO is not open-source and reverse-engineering on any distributed or generated file from PRIMO is expressly forbidden. Please refer to the disclaimer and copyright statements for further details. ■

**Notice 1.2** PRIMO can be downloaded from the PRIMO project webpage <http://www.primoproject.net>. It is forbidden to redistribute copies of PRIMO. ■

The sources of information about PRIMO are the following:

**User's manual** The document you are reading now. This manual is in its early stage. Future versions of the manual will include details on the models implemented. Currently, it only describes how to operate PRIMO at the user's level. Furthermore, the manual does not include any information on how to interpret results or about the intricacies of Monte Carlo simulation.

**The PENELOPE 2011 manual** Help on matters related to the Monte Carlo simulation might be found in the PENELOPE 2011 manual. Users can obtain a copy of the PENELOPE 2011 distribution by contacting the Nuclear Energy Agency (<http://www.oecd-nea.org/>).

**The PENEASY documentation** Since PENEASY is the main program steering the PENELOPE simulation, problems related to the Monte Carlo simulation might also be solved with the help of the documentation included in the PENEASY code available at <http://inte.upc.es/downloads>.

## 1.4 Version of PRIMO

PRIMO is still considered beta-software. Current version is 0.1.5.1300. The first, second, third and fourth number of the version are the major version, minor version, release and build numbers, respectively. New releases will be made available through the PRIMO project webpage.

## 1.5 List of citable references

If PRIMO is used for research conducting to publications the following bibliographical reference should be cited:

- Rodriguez M, Sempau J, Brualla L. PRIMO: A graphical environment for the Monte Carlo simulation of Varian and Elekta linacs. *Strahlenther Onkol* DOI 10.1007/s00066-013-0415-1. Available online at <http://link.springer.com/article/10.1007/s00066-013-0415-1>.  
Main PRIMO reference

The following references can also be useful:

- Sempau J, Badal A, Brualla L. A PENELOPE-based system for the automated Monte Carlo simulation of clinacs and voxelized geometries—application to far-from-axis fields. *Med Phys* 2011;38:5887–5895.  
Main PENEASYLINAC and PENEASY reference
- Salvat F, Fernández-Varea JM, Sempau J. PENELOPE 2011—A code system for Monte Carlo simulation of electron and photon transport. OECD Nuclear Energy Agency, Issy-les-Moulineaux.  
PENELOPE manual
- Sempau J, Acosta E, Baró J, Fernández-Varea JM, Salvat F. An algorithm for Monte Carlo simulation of coupled electron-photon transport. *Nucl Instrum Meth B* 1997;132:377–390.  
PENELOPE reference
- Baró J, Sempau J, Fernández-Varea JM, Salvat F. PENELOPE: an algorithm for Monte Carlo simulation of the penetration and energy loss of electrons and positrons in matter. *Nucl Instrum Meth B* 1995;100:31–46.  
PENELOPE reference

- Brualla L, Salvat F, Palanco-Zamora R. Efficient Monte Carlo simulation of multi-leaf collimators using geometry-related variance-reduction techniques. *Phys Med Biol* 2009;54:4131–4149.  
Movable-skis variance-reduction technique
- Brualla L, Sauerwein W. On the efficiency of azimuthal and rotational splitting for Monte Carlo simulation of clinical linear accelerators. *Rad Phys Chem* 2010;79:929–932.  
Rotational splitting variance-reduction technique
- Rodriguez M, Sempau J, Brualla L. A combined approach of variance-reduction techniques for the efficient Monte Carlo simulation of linacs. *Phys Med Biol* 2012;57:3013–3024.  
Splitting roulette variance-reduction technique
- Brualla L. Simulation of medical linear accelerators with PENELOPE. *On Radiation damage in biomolecular systems*. Eds. García Gómez-Tejedor G, Fuss MC. Springer, 2012.  
General introduction to the subject

## 2 — Installation

### 2.1 Tested hardware and software

We have tested and successfully run the code in the following hardware and software configurations:

- Computer with Intel 64 bits processor
- Windows 64 bits<sup>1</sup> operating system<sup>2</sup>
- Administrator rights
- Recommended 1 GB RAM per computing core. For example, a computer with 2 CPUs each with 4 computing cores requires about 8 GB RAM.<sup>3</sup>
- The hard drive must be either local or accessible through a high speed connection (at least 6 GB/s).
- PRIMO occupies less than 100 MB of disk space. Owing to the fact that large files might be generated during execution a minimum of 100 GB of free disk space is recommended for a normal usage of the software.
- Minimum screen resolution 1280 × 960 pixels. The default font size in Windows (smaller size) should be used for this minimum resolution.

### 2.2 Whence to obtain PRIMO

1. Visit the webpage <http://www.primoproject.net>.
2. Enter the section 'Download' and enter your name, email address and affiliation. You will receive an email with a link for downloading the software. PRIMO is distributed as a Windows installer msi file.
3. With the same link sent for downloading PRIMO it is possible to download already simulated examples. Each example is distributed as a compressed zip file.

<sup>1</sup>PRIMO has been successfully tested on Windows XP, Windows 7, Windows 8/8.1, Windows Server 2008 and Windows Server 2012. The classical view of the Windows desktop does not permit to see the full graphical quality of the program's icons.

<sup>2</sup>PRIMO may give problems on virtual machines (Parallels, VirtualBox, VMWare, etc.) and it does not run on Windows emulators (Wine, CrossOver, etc.)

<sup>3</sup>This 1 GB RAM rule also applies to logical computing cores in case of using hyper-threading. However, use of hyper-threading is not recommended.



## 2.3 How to install/uninstall PRIMO

### 2.3.1 Installer `msi` file

1. Execute the `msi` file.
2. The installation program will guide you through the installation process. The default installation folder is `c:\PRIMO` but it can be changed during the installation process.

**Warning 2.1** Errors during simulation execution could occur when PRIMO is installed in the Program Files folder of a server computer running Windows Server 2012. In that case it is advisable to select a different installation folder, such as `c:\PRIMO`.

### 2.3.2 Uninstall

Use the Windows Start Menu → Control Panel → Programs and Features to uninstall any version of PRIMO that was installed from an `msi` file.

**Warning 2.2** Uninstall any previous version before installing a new one.

## 2.4 How to install the examples

1. Create a directory called `PRIMOexamples` under `c:\`.
2. Unpack the downloaded files inside the just created directory `c:\PRIMOexamples`. The unpacked files will each one create a directory called `Example $mn$` , with  $mn$  a two digits number.

## 2.5 List of examples

**Example01** Photon reference field from a Varian Clinac 2100 C/D. Nominal energy 6 MV. Field size  $10 \times 10 \text{ cm}^2$ . Tallied results: phase-space files and dose distribution.

**Example02** Electron reference field from an Elekta SL15. Nominal energy 15 MeV. Electron applicator  $20 \times 20 \text{ cm}^2$ . Tallied results: phase-space file and dose distribution.

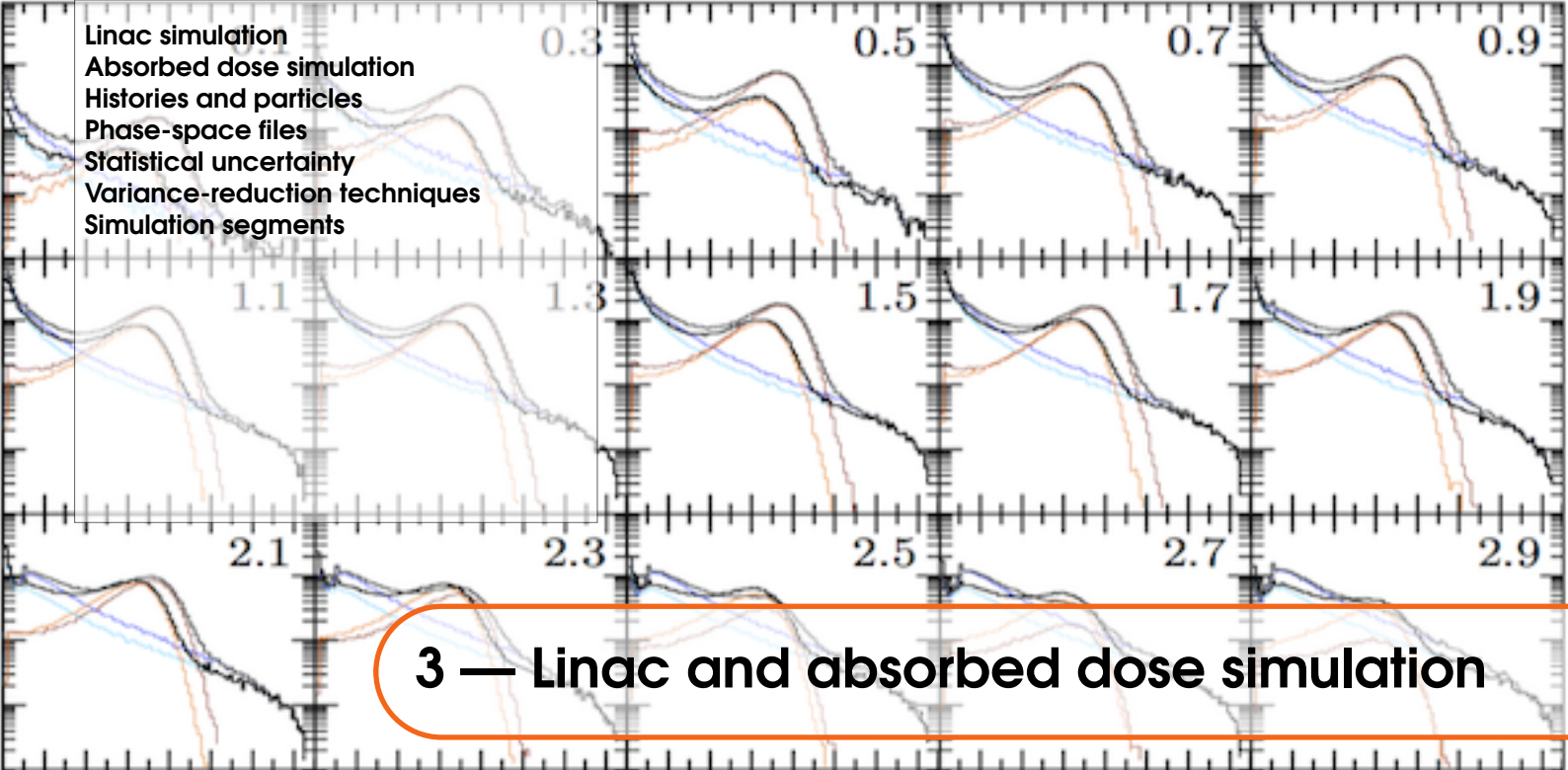
**Example03** Photon reference field from a TrueBeam STx. Nominal energy 6 MV (free flattening filter). Field size  $40 \times 40 \text{ cm}^2$ . Tallied result: dose distribution.

**Example04** Brain irradiation with two fields from a Varian Clinac 2100 C/D conformed with a MLC 120 High Definition. Nominal energy 6 MV. Tallied results: phase-space files and dose distribution.

Please notice the following remark:

- Ⓜ Examples 01, 02 and 03 contain subdirectories named `Experiment` where experimental data files are stored for comparison with simulated results.





Medical linear accelerators (linacs) are routinely used in radiotherapy units for the treatment of cancer. The purpose of all linacs is the same: to accelerate electrons through resonant cavities to energies on the order of a few MeV. The beam leaving the accelerating structure has a relatively narrow energy distribution with a diameter of about 1 mm. In general, Monte Carlo simulations start from that position in the linac head, assuming as primary electron source a beam with given spatial and energy distributions. Particles are then transported downstream the linac head. Therefore, from a Monte Carlo simulation point of view, the relevant constructive elements of the linac are those found downstream of the primary electron source.

### 3.1 Linac simulation

Some linacs operate only with electron beams (*e.g.*, Siemens Mevatron ME), others with photon beams (*e.g.*, Varian Clinac 600 C/D), while others can operate either with electron or photon beams (*e.g.*, Varian Clinac 2100 C/D). Those irradiating with electron beams usually include some thin material layers downstream of the primary electron source, called scattering foils, whose purpose is to spread the beam and hence to cover a large field. Linacs irradiating with photon beams have a thick material target, usually made of tungsten, in the beam path. This target produces photons by bremsstrahlung emission. In many cases a flattening filter is placed at the position of the scattering foils in order to homogenize the photon energy flux. From the primary electron source downstream to this position in the linac head all modeled linacs in PRIMO exhibit cylindrical symmetry. This segment of the linac head is referred to as the upper part. Next to the upper part a series of collimating structures are found whose purpose is to conform the beam to the required field shape. When a linac operates in photon mode, these structures consist of one or two sets of jaws and possibly a multileaf collimator. In the case of a linac operating in electron mode, an electron applicator is added below the multileaf collimator position. The constructive elements found downstream of the upper part are collectively called lower part of the linac, which does not exhibit cylindrical symmetry. Figure 3.1 shows four images of the constructive elements of the Varian Clinac 2100 C/D and Elekta MLCi operating in photon and electron modes. These images are actual representations of some of the simulated geometries in PRIMO.

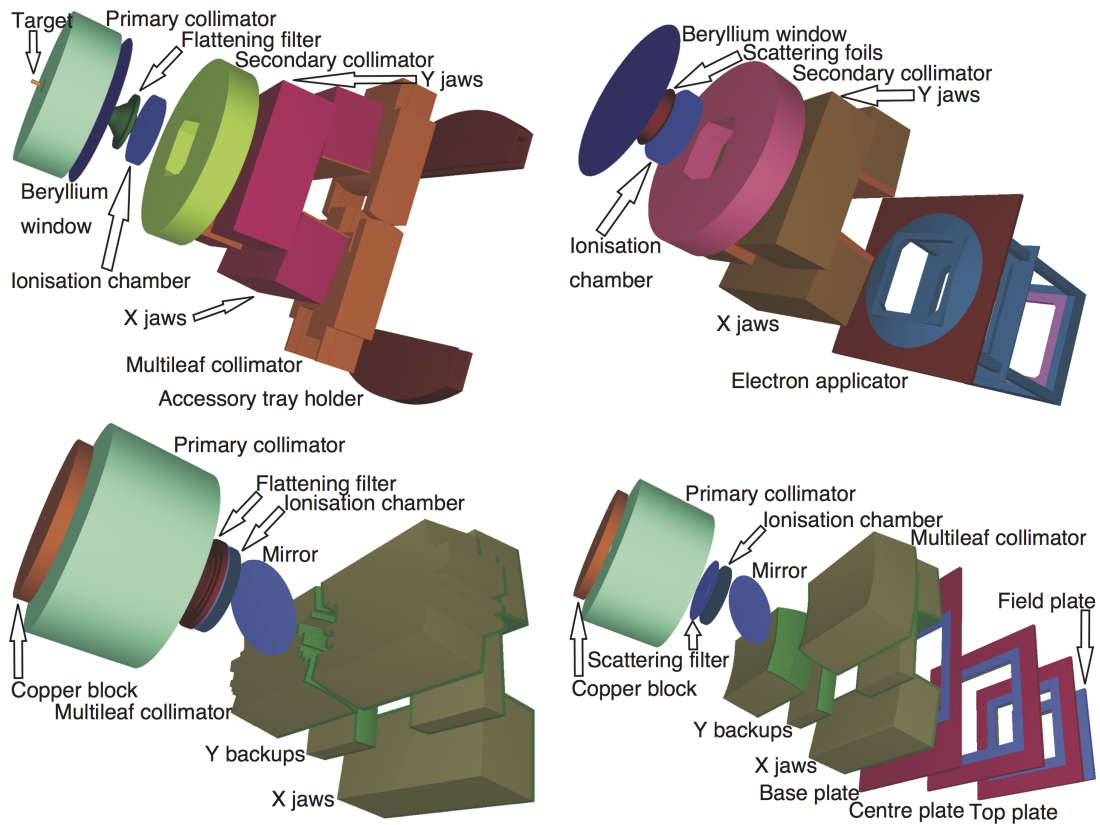


Figure 3.1: Varian Clinac 2100 C/D operating in photon mode at 18 MV (upper left) and electron mode at 6 MeV (upper right), Elekta MLCi operating in photon mode at 10 MV (lower left) and electron mode at 4 MeV (lower right). These images are the actual simulated geometries in PRIMO.

### 3.2 Absorbed dose simulation

After the lower part of the linac, the beam enters the region relevant for dosimetry purposes. The absorbed dose can be tallied either in a binned water phantom or in a voxelized structure. PRIMO can import RT-STRUCT files allowing for the simulation in voxelized phantoms generated in treatment planning systems by delineating structures. DICOM files containing computerized tomography images can also be imported and the dose tallied therein.

PRIMO reports dose in units of eV/g per primary particle. These units are equivalent to Gy/(mA s), whence the dose in Gy can be calculated knowing the current intensity at the target in mA and the irradiation time in s. When comparing with experimental profiles relative dosimetry is assumed.

### 3.3 Histories and particles

When a primary electron enters the modeled geometry, upstream of the upper part of the linac, an electromagnetic shower is simulated. It may occur that the primary electron is absorbed or escapes the geometry without further consequences, or it may happen that the primary electron produces secondary particles, namely, electrons, photons or positrons. In turn, these secondary particles may produce another generation of particles, and so on. The primary particle and all its descendants are simulated until all of them have been either absorbed or escaped the geometry. When this occurs one *history* has been completed. Therefore, the number of simulated particles and the number of simulated histories, in general, do not coincide. All the quantities reported in PRIMO are expressed in units per history, *e. g.*, the dose is expressed in units eV/g per history.

### 3.4 Phase-space files

When simulating radiation transport with the Monte Carlo method it is possible to define a surface, usually a plane, at any location in the geometry. Particles traversing this plane are stopped and their state (*i.e.*, energy, position, direction of flight, etc.) recorded on a file called phase-space file. When a phase-space file is 'sufficiently rich', that is, it contains a 'large number' of particles, it is possible to neglect the geometry upstream of the phase-space surface, and to consider the phase-space file as the radiation source for subsequent Monte Carlo simulations. The expressions 'sufficiently rich' and 'large number' refer to statistical properties of the phase-space file whose description is beyond the scope of this manual [Sem+01].

### 3.5 Statistical uncertainty

A straightforward approach to evaluate if a simulation has run long enough or if a phase-space file is sufficiently rich is by means of the statistical uncertainty estimator of the absorbed dose. PRIMO reports the average statistical uncertainty of all voxels (from computerized tomographies) or bins (from water phantoms) accumulating more than 50% of the maximum absorbed dose. All uncertainties reported by PRIMO are given at 2 standard deviations.

Statistical uncertainties obtained from PRIMO are correctly estimated provided the simulation has been wholly done inside PRIMO, that is, from the primary electron source to the patient or phantom. This is because PENELOPE keeps track, even in phase-space files, of the history to which each particle belongs. Not all general-purpose Monte Carlo codes keep such record. It is impossible to correctly estimate the statistical uncertainty when PRIMO uses phase-space files generated with codes that do not keep this record. Instead, PRIMO gives an approximation to the statistical uncertainty whose accuracy cannot be evaluated.

### 3.6 Variance-reduction techniques

The simulation of radiation transport through the linac head and the patient (or phantom) geometry is a very intensive computational task. A direct approach to the problem using analogue simulation with PENELOPE—‘analogue’ meaning that radiation interactions with matter are modeled as closely to reality as possible—would require of the order of several months of CPU processing for typical voxel sizes and statistical uncertainties [SBB11].

To reduce this unaffordable amount of computing time the so-called variance-reduction techniques can be used. They rely on the idea that a given probability distribution (of depositing a certain energy in a voxel, for instance) can be arbitrarily altered if the corresponding variable of interest (energy deposited, in our example) is also changed appropriately so as to keep its mean value unbiased. If the new probability distribution is chosen wisely, the statistical uncertainty  $\Delta$  achieved in a given amount  $t$  of computation time can be substantially reduced. Or, equivalently, a given uncertainty  $\Delta$  can be achieved in considerably less computing time.

A convenient measure of the efficacy of a certain variance-reduction technique is given by the simulation efficiency  $\varepsilon$ , defined as

$$\varepsilon = \frac{1}{\Delta^2 t}. \quad (3.1)$$

In PRIMO  $\Delta$  (of the absorbed dose distribution) is computed as twice the average standard statistical uncertainty, expressed as a percentage of the mean dose. Notice that this definition renders the dimensions of  $\varepsilon$  equal to those of  $t^{-1}$ . Thus, for a given simulation, the quantity

$$t_{1\%} = \frac{1}{\varepsilon} \quad (3.2)$$

represents the simulation time that would be required to achieve an average relative uncertainty (at two sigma) of 1%.

### 3.7 Simulation segments

PRIMO allows to tally a phase-space file at the downstream end of the upper part of the linac. This part is called segment 1 (s1). Similarly, a phase-space file can be tallied at the downstream end of the lower part of the linac. This region is called segment 2 (s2). The part of the simulation dedicated to the dose estimation is called segment 3 (s3).

Segments must, obviously, be simulated in sequential order, that is s1, s2, s3. However, they can be grouped according to the user’s requirements. They can be simulated individually as (s1, s2, s3); or grouped in a single simulation as (s1 + s2 + s3); or in smaller groups simulating s1 first and then s2 and s3 together (s1, s2 + s3); (s1 + s2, s3) is also possible.

**Warning 3.1** Combined quadric-voxelized geometries are not currently supported. Consequently, segments corresponding to the linac head (s1 and s2) cannot be grouped with the segment s3 if the dose is calculated in a CT.

A simulation can either tally a phase-space file or a dose distribution. Therefore, if simulating for example (s1 + s2 + s3) a dose distribution will be tallied. Simulation of (s1 + s2) and then a subsequent simulation of s3 will produce a phase-space file at the downstream end of the lower segment during the first simulation and a dose distribution during the simulation of s3.

The phase-space file obtained with the simulation of s1 depends on the primary beam parameters and the number of histories simulated. Once the primary beam parameters of a linac

have been tuned for a given nominal energy to reproduce experimental data from that linac, it is desirable to run, once and for all, a long simulation of  $s_1$  that can be re-used in subsequent simulations of the rest of the linac. This approach conduces to a substantial saving in simulation time, particularly in the case of photon beams.

Linacs operating in electron mode have one additional segment, namely,  $(s_1, s_2, s_{2e}, s_3)$ .  $s_1$  and  $s_3$  correspond to the segments previously described. Segment  $s_2$  simulates the movable collimators (*i.e.*, the jaws) and the uppermost two scrapers of the electron applicator, tallying a phase-space file at the downstream end of the middle scraper. Segment  $s_{2e}$  simulates the lowermost scraper and tallies a phase-space file at its downstream end. If the electron field is standard, as conformed by the electron applicator, then  $s_2$  and  $s_{2e}$  should be simulated together. However, if the user is interested in adding a customized collimator at the lowermost scraper it is necessary to simulate only up to  $s_2$ . In that case, segment  $s_{2e}$  with the customized collimator must be simulated with an external program, such as, PENELOPE. The current version of PRIMO does not allow for the simulation of customized electron collimators.

When importing external phase-space files, PRIMO assumes that they have been tallied at the downstream end of  $s_1$ . After importing the phase-space file,  $s_1$  will appear as already simulated and the user will be given the possibility of either simulating  $(s_2 + s_3)$  or  $(s_2, s_3)$ .

- Ⓡ Except under special conditions, the most common approach to simulate the linac and the subsequent dose distribution is  $(s_1, s_2, s_3)$ .





## Simulation setup

- New project
- Beam, field and dose tally configuration
- Material and density assignment
- Contouring
- Simulation configuration
- Transport parameters
- Variance reduction
- Importing external phase-space files
- Duplicating a project

## Execution

- Changing parameters at runtime

## Analysis of results

- Phase-space file analysis
- Dose analysis
- Changing the relative field weight
- Inspecting the dose distribution
- Comparing to experimental data
- Dose comparison by gamma analysis

# 4 — PRIMO usage

This chapter is the core of the manual, covering how to operate PRIMO. The graphical interface is designed to reduce the user's effort to accomplish the steps of *simulation setup*, *execution* and *analysis of results*. Any interface element can be associated to one of these three main categories.

## 4.1 Simulation setup

Immediately after launching PRIMO in the simulation setup environment the user can either start a new simulation project or open an already simulated one (figure 4.1).

### 4.1.1 New project

When the *New project* button is clicked the New Project window (figure 4.2) appears. The following elements are available in that window:

- Project ID: Mandatory field in which the name of the project is entered. A maximum of 15 characters is allowed. The name cannot contain spaces or other characters that are usually not accepted for file names.
- Project name: Optional field in which a succinct explanation about the characteristics of the project may be given.
- *Browse*: By default PRIMO will save the new project in the installation directory. Nevertheless, it is possible to save the new project in any other logical drive and directory. The *Browse* button allows to decide in which drive and directory the project will be saved. A logical drive mounted on a remote disk can be used. However, the recommended communication speed must be at least 6 GB/s. Once a project has been saved in a given directory it cannot be moved to any other location.
- Linac model: This drop-down menu allows to choose the desired linac model to be simulated. Please refer to table 4.1 in order to decide which model corresponds to the



Figure 4.1: Main task bar of the simulation setup environment. First two buttons from left to right are *New project* and *Open project*.

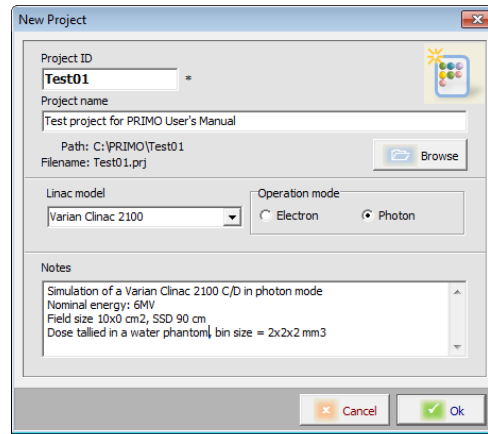


Figure 4.2: New project window.

PRIMO	commercial
Elekta SL	SL series
Elekta MLCi	SLi Plus, Axesse, Affinity, Synergy, Precise
Varian Clinac 600C	Clinac 600 C
Varian Clinac 600CD	Clinac 600 C/D
Varian Unique	Unique
Varian Clinac 2100	Clinac C series, TrueBeam (sect. 5.3)
Varian Clinac 2300	Clinac 2300 C/D
FakeBeam	TrueBeam 6- and 10-FFF beams

Table 4.1: The column ‘PRIMO’ indicates the name given to the available linacs in the program. The column ‘commercial’ indicates the different commercial names that the same linac in PRIMO might have. FakeBeam is an experimentally based geometry of TrueBeam developed in-house.

desired linac.

- Operation mode: These radio buttons allow to decide whether the linac will irradiate either in electron or photon mode.
- Notes: This field can be used for text notes.

**Warning 4.1** Elekta linacs have been recently coded into PRIMO. Therefore, they have not been neither extensively tested, nor fine tuned. It is recommended that users willing to simulate Elekta linacs dedicate some effort in fine tuning the primary beam parameters.

**Warning 4.2** The only multileaf collimator model in PRIMO that has been experimentally validated is the Varian MLC 52.

### Simulation setup tab

After clicking the *Ok* button in the New Project window, the program presents the window corresponding to the Simulation Setup tab selected (figure 4.3).

The screen of the Simulation Setup tab is divided in two areas, namely, the Simulation Segments and the Segment setup. A logical tree containing the objects of the project appears



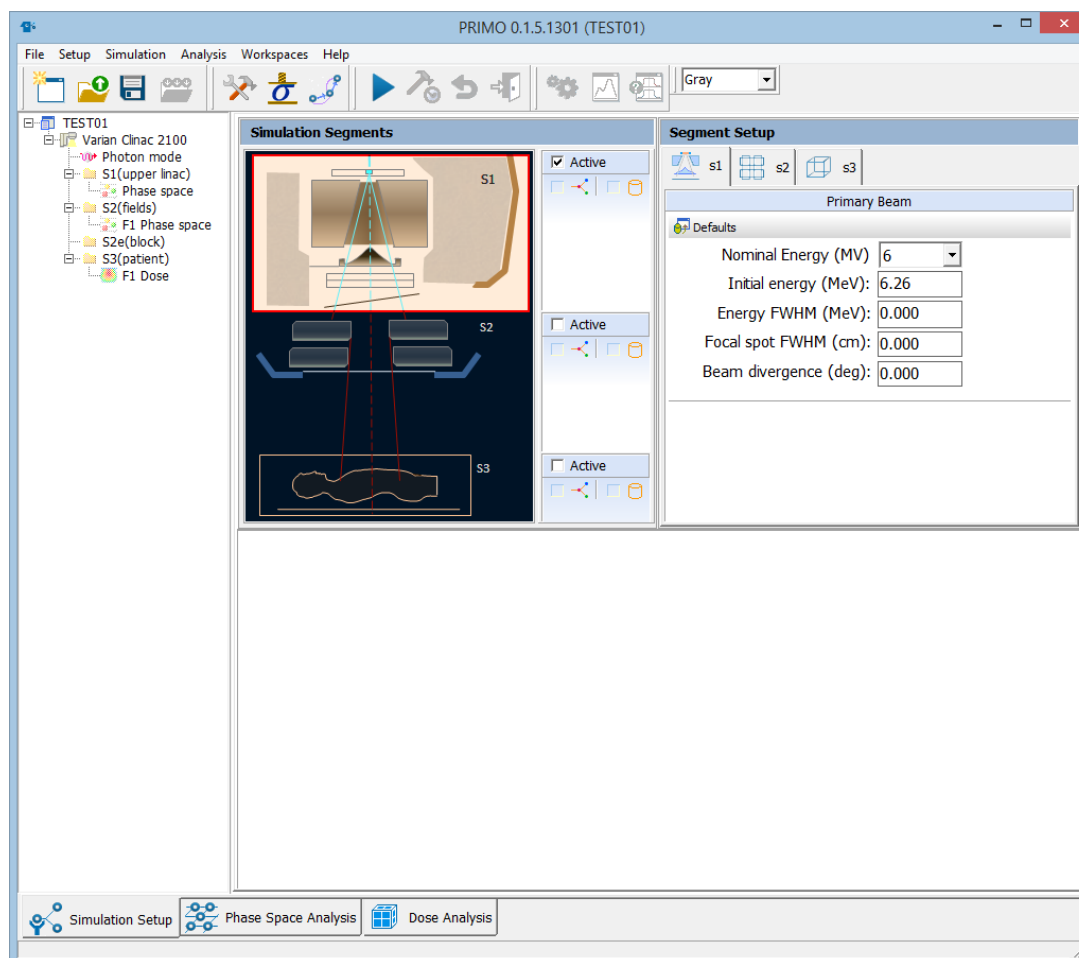


Figure 4.3: Simulation Setup tab.

always at the left in the main screen.

### Simulation Segments

The Simulation Segments area indicates in a schematic view of a linac the segments (see section 3.7) that will be or have been simulated. Segments that are required to be simulated must be checked in their corresponding Active checkbox. When a segment is selected for simulation the corresponding area of the linac is highlighted in the schematic view. Once a segment has been simulated a check mark appears next to the three-particle interaction symbol of the corresponding segment. If that segment has produced a tally (phase-space file or dose distribution) a check mark appears next to the hard drive symbol (a cylinder).

#### 4.1.2 Beam, field and dose tally configuration

The Segment Setup area allows users to configure each segment of the simulation.

**Notice 4.1** When the configuration of s2 and s3 is known beforehand, it is recommended to configure segments s1, s2 and s3 (see section 3.7) at the very beginning of the simulation project, even if only segment s1 will be simulated (*e. g.*, variance-reduction techniques applied at s1 could be related to the field size selected in s2). ■

#### Beam parameters (s1)

The nominal beam energy is chosen in the drop-down menu in tab s1. For each nominal energy a set of recommended initial beam parameters is suggested. These parameters are: the initial electron energy, the energy full width at half maximum (FWHM), the FWHM of the focal spot size and the beam divergence. A Gaussian distribution for the energy and for the radial distribution is assumed. Beam divergence is implemented such that the angular divergence is for each point emitting from the source plane.

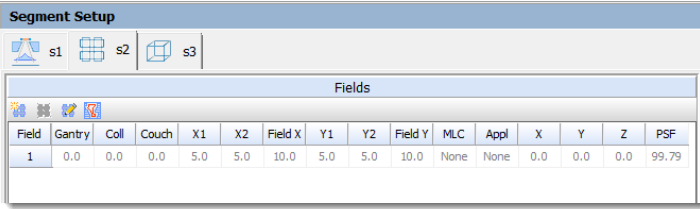
Default beam parameters have been tuned, in the case of Varian linacs, to reproduce experimental results from the corresponding linac and energy. In the case of Elekta linacs the beam parameters have not been tuned yet. In all cases the user can modify the values.

#### Field conformation (s2)

The field to be simulated in s2 is defined by selecting the tab s2 (figure 4.4). It is possible to simulate multiple fields. To simulate more than one field the *Add new field* button must be clicked. There is also a button for deleting a selected field. The *Edit selected field* button opens the corresponding screen (figure 4.5) which allows to define the field size and position, as well as gantry, collimator and couch angles, and isocenter position. Multileaf collimators or electron applicators can be selected in this screen. The field Phase space plane informs the distance of the phase-space plane of s2 from the primary source. This distance cannot be changed by the user, it is fixed such that the phase-space plane is near to the exit of the linac head, although it varies with the linac model, operation mode and accessories selected (*e. g.*, it is 70.0 cm for a Varian Clinac 2100 in photon mode).

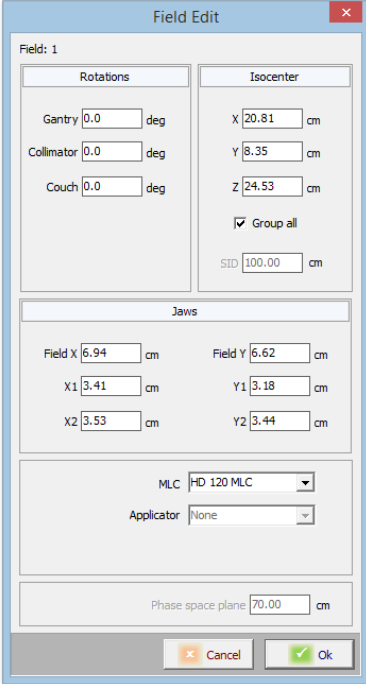
In electron mode, the field aperture as defined by the jaws is automatically set according to the default specified by the manufacturer for the beam energy and electron applicator selected. However, the user is able to reposition the jaws conveniently.

When the patient model in s3 is a CT (see Dose tallying (s3)), each field in a multi-field project can have a different isocenter position. By default all fields are grouped (the option *Group all* is checked) such that they all have a unique isocenter position. To change the isocenter position for a field, uncheck the option *Group all* and change manually the isocenter position in the *Field Edit window*. To assign a unique isocenter position to all fields, select the field that



Field	Gantry	Coll	Couch	X1	X2	Field X	Y1	Y2	Field Y	MLC	Appl	X	Y	Z	PSF
1	0.0	0.0	0.0	5.0	5.0	10.0	5.0	5.0	10.0	None	None	0.0	0.0	0.0	99.79

Figure 4.4: Field definition window. s2 tab.



Field: 1

Rotations		Isocenter	
Gantry	0.0 deg	X	20.81 cm
Collimator	0.0 deg	Y	8.35 cm
Couch	0.0 deg	Z	24.53 cm
		<input checked="" type="checkbox"/> Group all	
		SID	100.00 cm

Jaws

Field X	6.94 cm	Field Y	6.62 cm
X1	3.41 cm	Y1	3.18 cm
X2	3.53 cm	Y2	3.44 cm

MLC: HD 120 MLC  
 Applicator: None

Phase space plane: 70.00 cm

Cancel Ok

Figure 4.5: Edit selected field. s2 tab.

will be used as the reference, open the *Field Edit window* and check the option *Group all*. The position of the isocenter for the selected field will be assigned to all fields.

- R** The linac source-to-isocenter distance is a constant (100 cm) and it is specified in the field *SID*. Extended SSDs are only allowed in simulations of a water phantom.

The *Field Edit BEV* button opens an interactive window that facilitates the setup of the fields (figure 4.6). This allows to set the field parameters from a beam eye view (BEV) perspective in which the position of the jaws, the isocenter and the multileaf collimator (MLC) are represented superimposed to the Digitally Reconstructed Radiograph (DRR) of the CT volume. Delineated or imported structures can also (optionally) appear projected on the image. The position of the isocenter, as well as the projection of the radiation fields, are also represented on the CT/phantom slices corresponding to the isocenter position. The position of the jaws and the isocenter can be changed by dragging the corresponding symbols in the image. In the same way, the MLC leaves can be selected and dragged to move them to a particular position. While dragging, the corresponding position values are automatically updated. The tool bar at the bottom of the image allows to manually edit the selected field, to refresh the DRR, to zoom in and out the image and

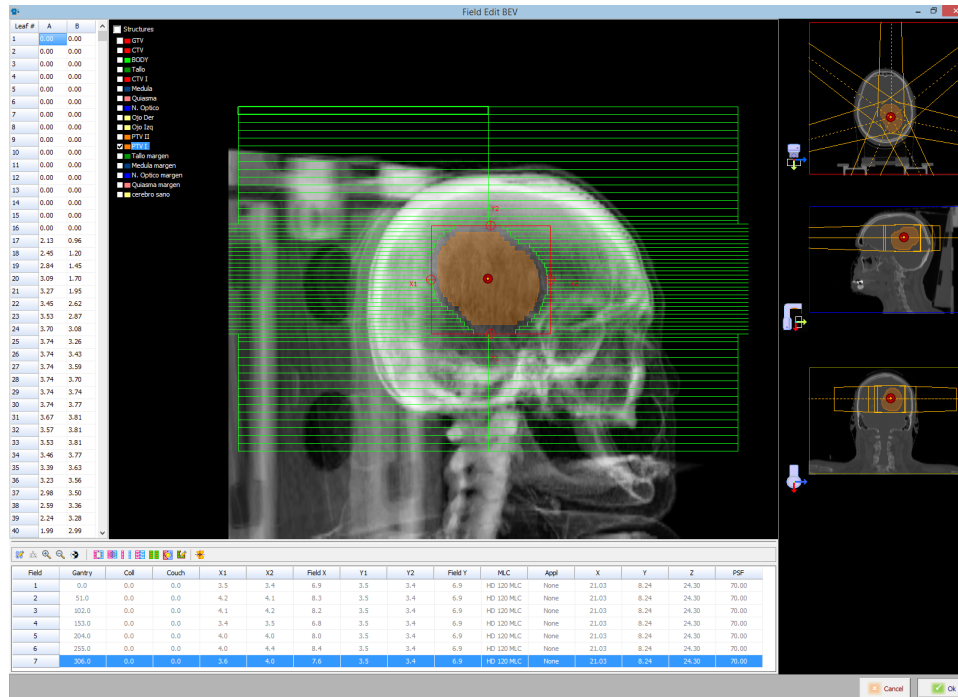


Figure 4.6: Field Edit BEV dialog.

to invert the color lookup table. It also includes functions to visualize and position the MLC. Several contiguous MLC leaves can be selected and displaced together. To select a range of leaves, left click on the first leaf and press shift + left click on the last leaf. The position of the leaves selected can be changed by dragging them or by specifying the new position numerically with the keyboard. This is available by pressing the *Edit selected leaves* button in the toolbar.

Two targeting functions are implemented to facilitate the field positioning, namely, the MLC can be fitted to a structure contour and the isocenter position can be situated at the centroid of a structure. Those functions are available by pressing the buttons *Fit to structure* and *Adjust isocenter to target*, respectively. A dialog appears in each case to select the target structure and a margin out of the structure contour in the case of fitting the MLC.

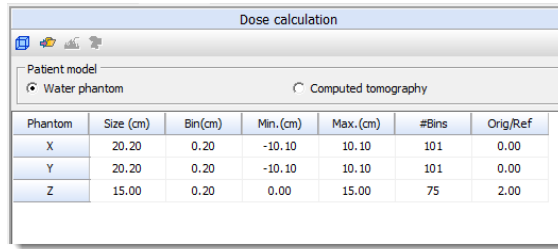
**Notice 4.2** The *Field Edit BEV* window can be resized/maximized conveniently. ■

### Dose tallying (s3)

There are two geometry models (patient models) available for dose calculation that can be chosen in segment s3 setup, namely, an homogeneous water phantom and a computerized tomographic volume. The water phantom is selected by default. To change the model, a CT volume (formatted as a set of DICOM images) must be imported or a slab phantom can be created. A slab phantom is treated in PRIMO as a CT volume. The current dimensions of the phantom or the CT volume are shown in the s3 tab (figure 4.7).

### Water phantom

To change the dimensions of the water phantom, click the button *Edit phantom dimensions*. In the appearing Phantom Setup dialog window (figure 4.8) it is possible to set the source-to-phantom-surface distance, the phantom size and the bin size along the three coordinate axes. It is also possible to define a measurement depth along the phantom central axis in order to calculate



Phantom	Size (cm)	Bin(cm)	Min.(cm)	Max.(cm)	#Bins	Orig/Ref
X	20,20	0,20	-10,10	10,10	101	0,00
Y	20,20	0,20	-10,10	10,10	101	0,00
Z	15,00	0,20	0,00	15,00	75	2,00

Figure 4.7: Dose calculation. s3 tab.

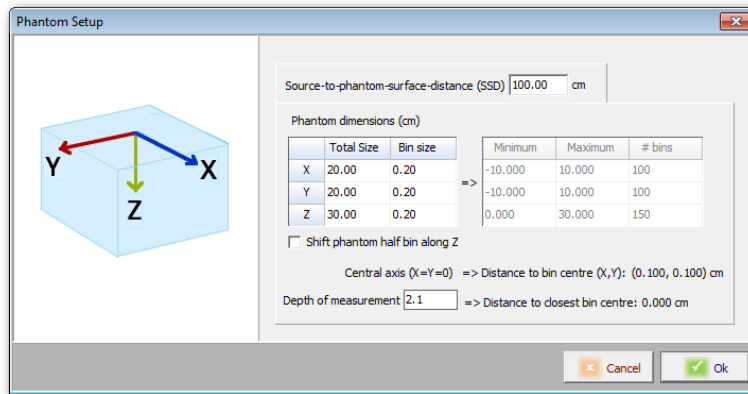


Figure 4.8: Phantom Setup dialog.

the distance from that depth to the center of the closest bin in the  $z$ -axis direction, according to the phantom and bin dimensions chosen. This is useful if one needs to ensure that a particular depth along the central axis coincides exactly with the center of a bin. In this case the calculated distance should be equal to 0.0 in the three axes. There is also the option Shift phantom half bin along Z that produces a displacement of the bins deeper into the water such that the surface of the phantom (the interface air-water) lies at a distance of one bin of the center of the first layer of bins in the  $z$ -axis direction.

The coordinate systems used in phantom simulations and CT volume simulations are shown in figure 4.9. The default isocenter is set at the center of the upstream surface of the phantom. Only one field can be created at gantry angle 0 degrees and the source-to-phantom-surface-distance should be defined in the s3 segment setup. Observe that for an water phantom, the *Field Edit BEV* dialog is useful only to set the field size and the MLC leaves positions.

- R If the dose is calculated in a water phantom only one field is allowed and it is placed centered at the linac CAX at a given SSD. The gantry, collimator and table angles are fixed at 0, 0, 0, respectively for the field.

**Notice 4.3 — Maximum number of bins.** The maximum number of bins ( $n_{bin_x} \times n_{bin_y} \times n_{bin_z}$ ) allowed in a phantom simulation is  $10^7$ .

- R When simulating the whole linac and a water phantom, consider to separate the segments s1 and s2 (involving the radiation transport through the linac geometry) from the segment s3 (involving the phantom geometry only) as it is more efficient than simulating the three

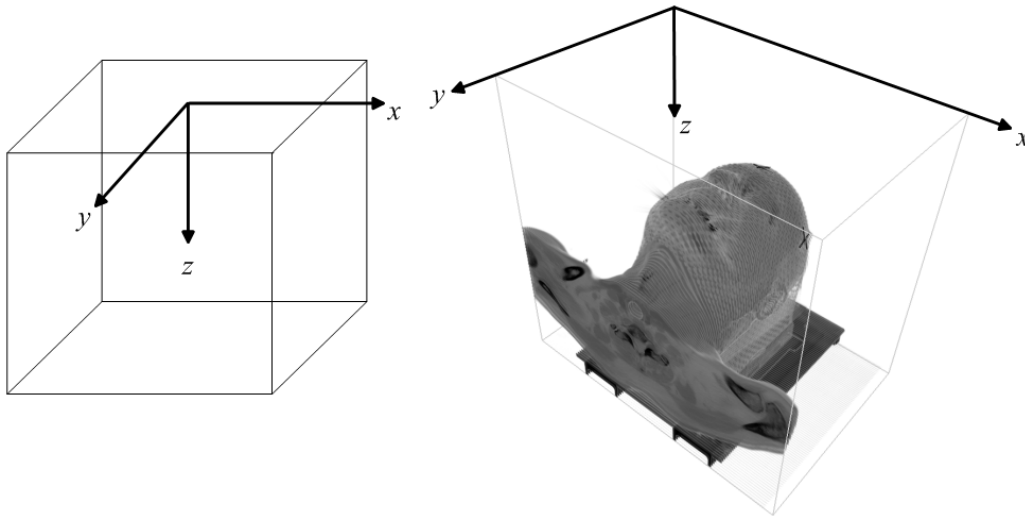


Figure 4.9: Coordinate system of the phantom and the CT volume used to setup the dose tally.

segments together. In the current version, separation of the segment s3 simulation is mandatory when the patient model is a CT volume.

### CT scan

To change the patient model from the default water phantom to a CT volume select the *Import a CT volume* option on s3. In the appearing standard File Open window from the operating system select a set of files containing each a CT slice. All slices must belong to the same study. Files must conform with the Digital Imaging and COmmunications in Medicine (DICOM-CT) format. The DICOM Image Import dialog (figure 4.10) will process the DICOM files to check for errors or inconsistencies and will build the tomographic volume. The set of images imported and some relevant data are presented on the screen. Once the button *Import* is pressed, the images are imported into the project and each slice is converted to a size of  $256 \times 256$  pixels.

**Warning 4.3** Any segmentation done on the CT volume and any previously imported or delineated structure will be permanently deleted if the patient model is changed to a water phantom or another CT is imported.

The origin of the coordinate system is set in the CT volume at its upper left behind corner, as represented in figure 4.9. The isocenter is by default placed at the center of coordinates, *i. e.*, at position  $(0,0,0)$ . When the project is created, one default  $10 \times 10 \text{ cm}^2$  field is created with gantry angle equal  $0^\circ$ , centered at the position of the default isocenter. It must be noted that in this situation only half the CT is represented in the DRR. To change the isocenter to the desired position there are three forms: (1) manually; (2) by dragging it over the DRR image or (3) by positioning it at the centroid of an structure. For the field created by default, option (2) will only modify the  $x$  and  $y$  coordinates. To change the third coordinate with this procedure, a new field can be created with gantry angle at  $90^\circ$  and used to displace the isocenter over the DRR. This “auxiliary” field can be erased after positioning the isocenter.

The imported CT volume is used to generate a voxelized simulation geometry. This geometry consists of a set of material and mass density value pairs. There is one pair per voxel in the

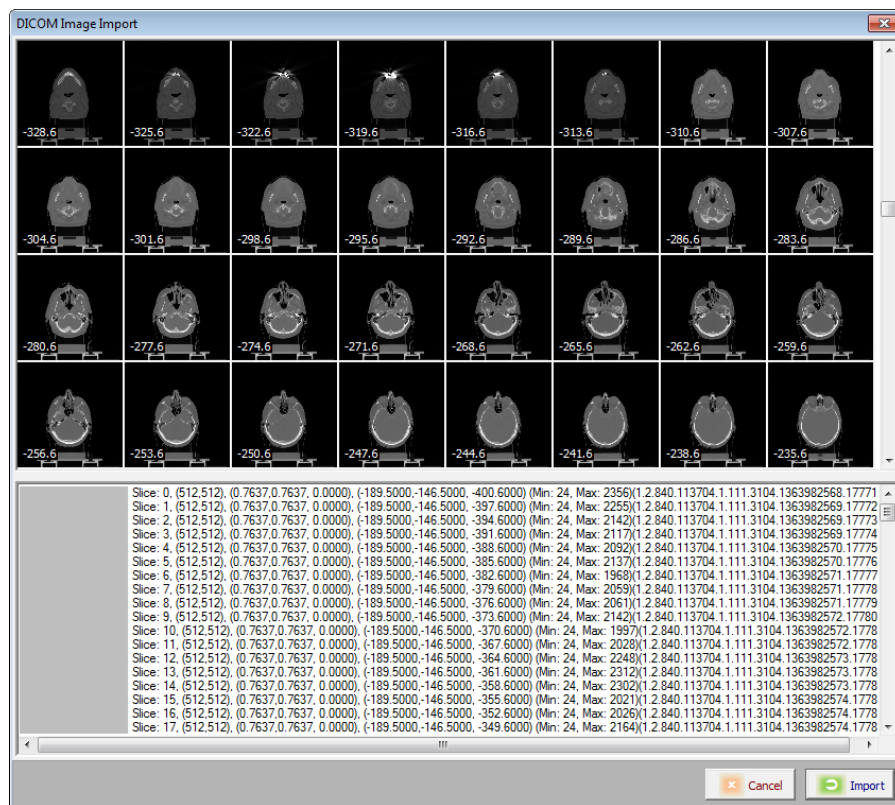


Figure 4.10: DICOM Image Import dialog. The gray scale is mapped into each image Hounsfield number range.



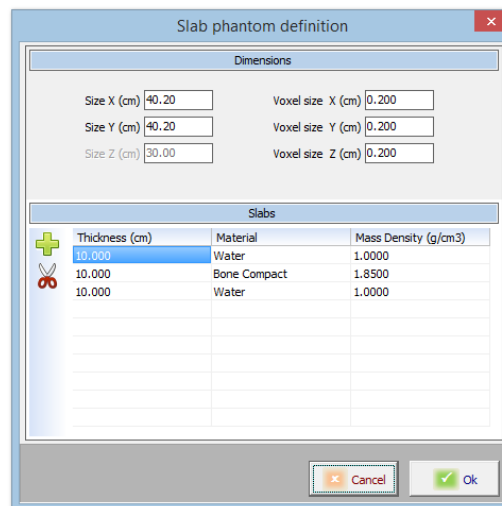


Figure 4.11: The *Slab phantom definition* dialog.

tomographic volume. The option Calculate density and materials in the s3 tab displays the CT Volume Segmentation dialog that allows to create the voxelized geometry (figure 4.12).

### Slab phantom

In the case of the patient being a slab phantom, there is not need to import the phantom from a DICOM-CT file. A phantom of slabs can be created using the function *Construct a slab phantom* in segment s3 setup. The *Slab phantom definition* dialog allows to establish the dimensions and the voxel size of the phantom and to add/delete slabs with a material composition selected from a list (Figure 4.11). The slabs are created with a default thickness of 10 cm but this value is editable. A maximum of 10 slabs can be included in the phantom. A slab must contain an integer number of voxels, consequently the voxel size along  $z$  and the slabs thicknesses must be adjusted accordingly. The phantom is created as a CT volume, so all the functions associated to the management of a CT patient model apply to the phantom. There is no need to use the function *Calculate densities and materials* for a phantom of slabs. When the phantom is created the isocenter is automatically positioned at the center of the upstream surface of the phantom (SSD=100 cm) for all the fields. The user can, as in the case of a CT patient, reposition the isocenter and change other parameters of the fields in the *Field Edit BEV* window.

- R** Note that once the *Slab phantom definition* dialog is closed and the slab phantom is created as a CT scan it cannot be further changed.

**Warning 4.4** The dimensions and voxel size of the slab phantom must be such that the total number of voxels does not surpass  $10^7$ . Dimensions are limited to a maximum of 50 cm along  $x$  and  $y$ . Additionally, two different materials with the same mass density cannot be included in the phantom.

### 4.1.3 Material and density assignment

The volume segmentation is done by assigning a material to a CT number interval. Up to 10 materials, chosen from a list of more than 40, can be assigned to a CT volume. The list of assigned materials, and their corresponding CT number interval, appears under the title *Materials*



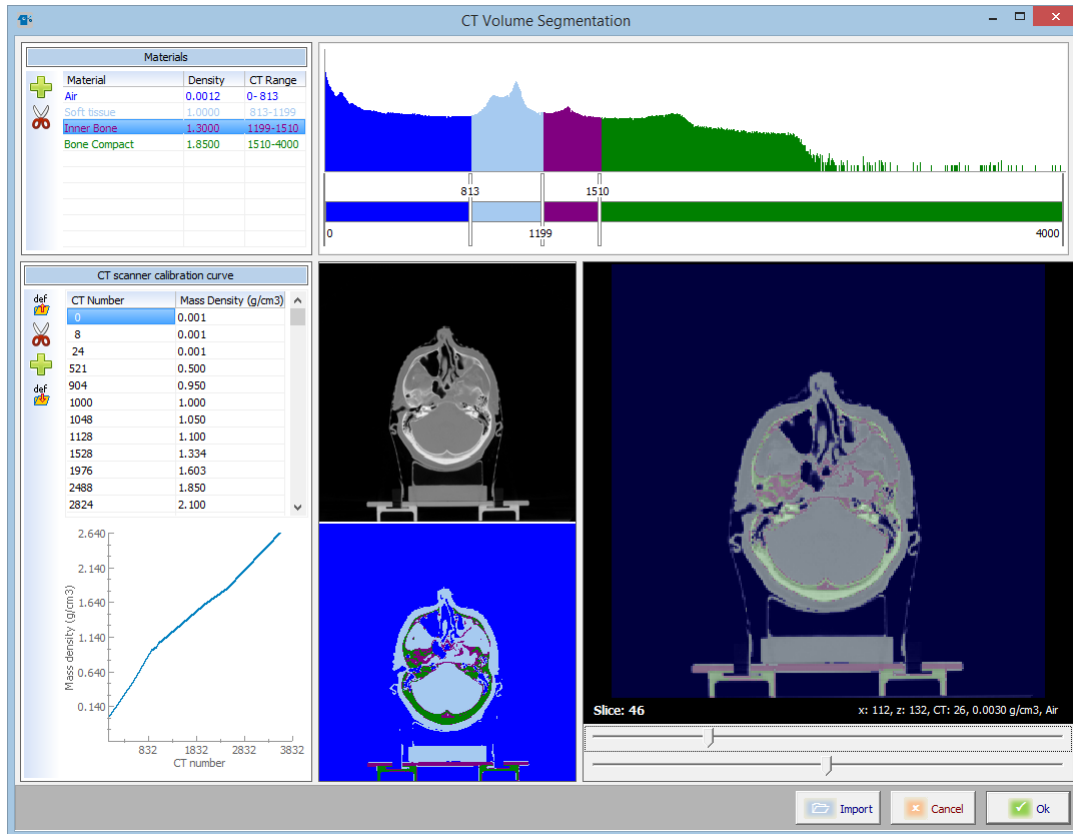


Figure 4.12: CT Volume Segmentation dialog. Generation of voxelized geometries.

on the left top corner of the dialog. Each material/interval is differentiated with color. The buttons on the left allow to remove and add materials to the list. Changing a CT number interval for a material is performed by dragging the sliders along the CT number histogram.

**Notice 4.4 — Working with the Hounsfield numbers histogram.** As the mouse is moved over the histogram, the corresponding CT number is displayed. Right clicking at the histogram will move the slider situated on the left side of the mouse pointer to the clicking position in the histogram. Left clicking at the histogram will move the slider situated on the right side of the mouse pointer position.

The CT scanner calibration curve is used for assigning mass densities to CT numbers. A default curve is provided, but it is possible to edit the default curve to create a custom one. In the CT scanner calibration curve sheet, it is possible to select a cell and change its value. It is also possible to add or delete entries. The plot updates automatically according to the changes made in the cells. The *Save the curve as default* button allows to save the edited curve as the default one.

**Warning 4.5** Once a mass density versus CT number calibration curve is saved as default, the original default curve supplied with PRIMO is lost.

The image displayed at the right bottom corner of the CT Volume Segmentation dialog is a blended image of densities and materials. Densities are mapped to a gray scale and materials to

a discrete color scale. Moving the slider at the bottom of the image to the left will foreground the densities in the image and moving it to the right will foreground the materials. The lower slider allows to change the displayed CT slice, this can also be done by scrolling with the mouse wheel. Clicking the *Ok* button will save the current segmentation and will create the voxelized geometry. A segmentation created in another project can be imported into the current one by clicking the *Import* button.

**Notice 4.5** Moving the mouse over the blended image produces a display (at the bottom of the image) of the coordinates, CT number, mass density and material of the image pixel under the mouse pointer. ■

#### 4.1.4 Contouring

A set of structures can be created from drawing its contours on the CT volume. Contours can also be imported provided they are formatted as the DICOM-RT STRUCT standard. Press the *Delineate structures* button in the *s3* tab to open the contouring dialog (figure 4.13). In the Contouring dialog menu bar, click the button *Create* to create a new structure. A dialog (figure 4.13) appears querying for the ID and color that will identify the structure. To draw a structure contour, first select a structure in the list, and then press the button *Draw*. To delineate the contour do the following:

- Select the reference image by moving the sliders in the three dimensions.
- Create the first point of the contour by clicking the left mouse button on the image.
- To create a continuous contour, move the mouse along the path of the contour while keeping the left mouse button pressed.
- To create a discrete (point to point) contour, click the left mouse button at the points. The contour is formed by connecting the points with lines.
- To delete a point, click the right button of the mouse.
- To close the contour, approximate the mouse pointer to the first point, a small circle appears, then click inside the circle and the contour will be closed.

Contours can be drawn on selected image planes. Missing contours will be created by clicking the *Interpolate* button.

Zoom In/Out the CT volume by pressing the *Zoom the Image* button. Drag the mouse over the image in the left-up/right-down direction to zoom in/out the CT volume. The point where the three planes intersect is taken as the center of the zoomed region, which can be changed by panning the image with the *Pan* button.

#### Import structures

Structures created externally can be imported by selecting the button *Import a structure set*. In the standard File Open dialog select the file containing the structure set. The file must be formatted as a DICOM-RT STRUCT object and the structures must belong to the same study of the imported CT volume. The DICOM-RT STRUCT Import dialog (figure 4.15) shows a graphic representation of the contours on each image plane and a list of structures and their attributes. Uncheck the box under the column *Import* to exclude a structure. In case a Hounsfield number is associated to a structure the CT volume can be modified by assigning that number to all the voxels inside the structure. To accomplish this, check the box under the column *Mod. CT*. Finally, press the button *Import* to start importing the selected structures into the PRIMO project.

**Warning 4.6** The Hounsfield numbers of the CT volume regions changed when importing structures with the *Mod.CT* option checked are lost.

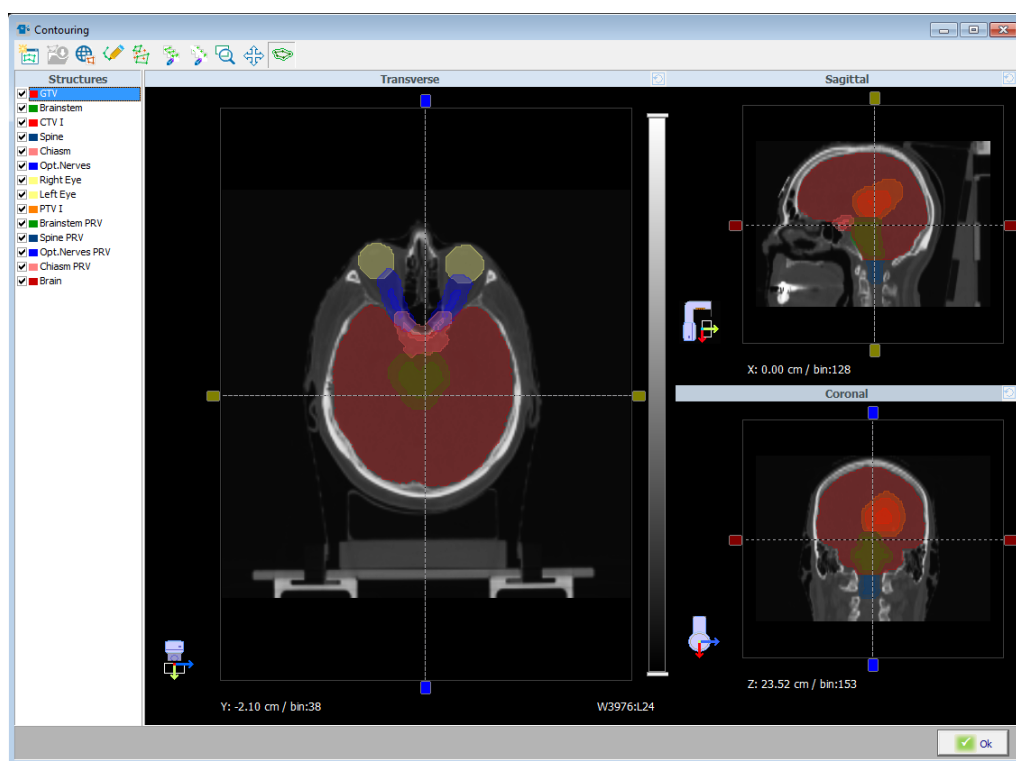


Figure 4.13: Contouring dialog. Creation of structures.

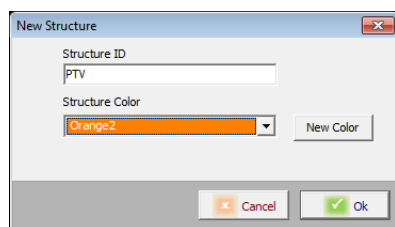


Figure 4.14: New structure dialog.

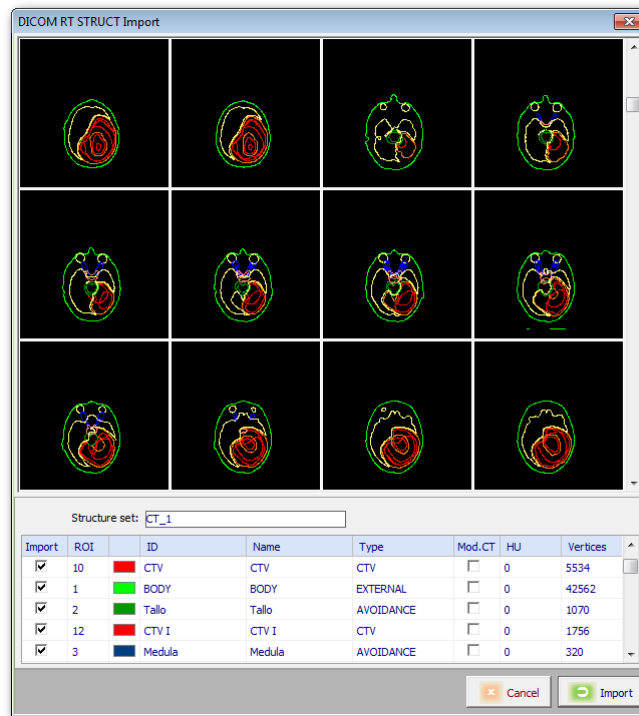


Figure 4.15: Import DICOM-RT STRUCT dialog.

#### 4.1.5 Simulation configuration

To have access to the parameters that govern the simulation execution, select *Configure simulation options* in the main menu or main tool bar. The Simulation configuration dialog (figure 4.16) allows to set the seeds of the random number generator, the stop conditions of the simulation, the frequency of reporting partial simulation results, and the number of CPU cores used. New seeds can be entered manually or automatically generated by clicking on the small dice in the dialog. In the latter case seeds are generated with a separation of  $10^{15}$  calls to the pseudo-random number generator [BS06; SFS11].

- Ⓡ If both stopping conditions, time and number of histories simulated are set, the first fulfilled condition will end the simulation.

**Warning 4.7** Setting more simulation processes than available CPU cores slows down the simulation. The minimum amount of Random Access Memory (RAM) required for a simulation is about 0.5 GB per process.

#### 4.1.6 Transport parameters

PENELOPE requires to define a set of simulation parameters which determine the trade-off between speed and accuracy. Refer to the PENELOPE 2011 manual [SFS11] for detailed information about the transport parameters. PRIMO provides default values for this set. Nevertheless, users can modify the default set by editing the table found under the option *Configure transport parameters* in the main menu and the main tool bar. It is advisable to carefully read the PENELOPE 2011 manual before attempting any change in this table, which can be edited by checking

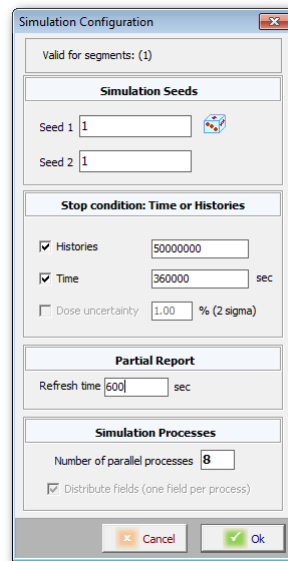


Figure 4.16: Configure simulation options dialog.

the Enable Editing box of the dialog (figure 4.17). The works of Brualla and co-workers [BSP09], Sempau and Andreo [SA06] contain some advices on how to set the transport parameters. The button Load default values will restore the default set of transport parameters.

**Warning 4.8** Modification of the transport parameters table should only be attempted by experienced users.

#### 4.1.7 Variance reduction

Several variance-reduction techniques are available under the main toolbar function *Configure variance reduction*. These techniques include forcing of bresstrahlung interactions in the linac target, simple splitting in the water phantom or the CT, and two splitting techniques developed by the authors of the code, the rotational splitting [BS10] and splitting roulette [RSB12]. Additionally, moveable-skins [BSP09] are used in the jaws, the primary collimator and the MLCs. An appropriate skin thickness is automatically selected by the code for each component and nominal energy.

A suitable combination of interaction forcing in the target and splitting in the patient can in some cases improve the efficiency considerably, the appropriate combination of forcing and splitting factors depends on the energy and the field size and it is currently under study. As a general rule it is recommended to keep the forcing factor relatively small (*e. g.*, in the vicinity of 16), because large forcing factors increase considerably the simulation time and counterbalance the effect that reducing the variance has on the simulation efficiency.

The methods that, so far, have proven to be the most efficient are rotational splitting and splitting roulette, combined with a simple splitting in the dose tallying region. As a general rule, splitting roulette is more efficient than rotational splitting at low energies, so it is the recommended method for beams with energies under 15 MV.

Most of the parameters of these two techniques are configured such that to obtain the optimal efficiency. One of the few setting let to the user is that of defining the size of the splitting region, a circular region located at a plane upstream the jaws, and that is used, although playing a different

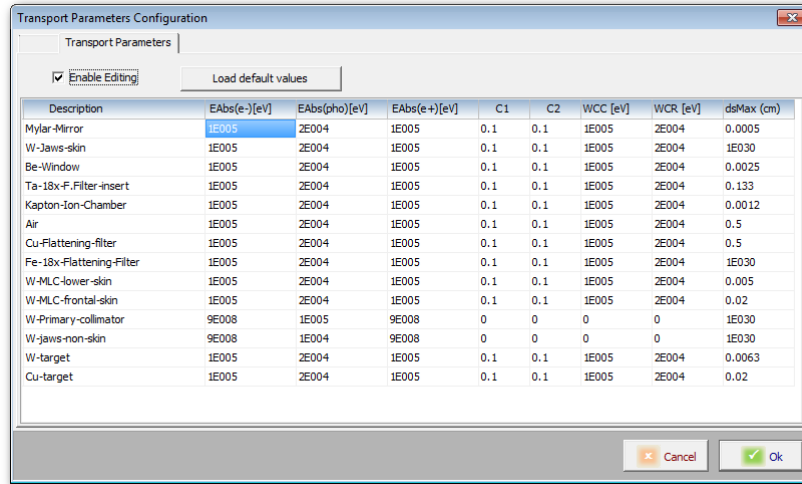


Figure 4.17: Configure transport parameters dialog.

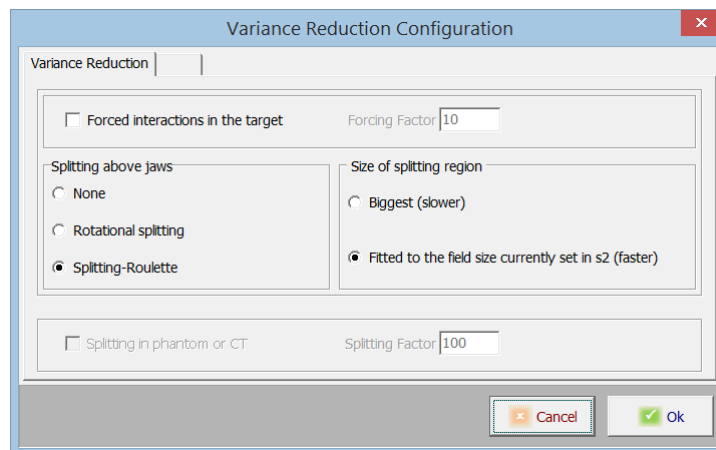


Figure 4.18: Variance reduction configuration dialog.

role, in both techniques. Particles flying in the direction of this region are considered in the splitting roulette technique with high probability of contributing to the dose. Rotation splitting is done only to particles crossing the plane of the splitting region inside its limits. In both cases, particles crossing the plane of the splitting region out of its limits are removed from simulation. Consequently, the diameter of the splitting region must be set larger than the diagonal size of the field as defined by the jaws (assuming the field is symmetrical with respect to the CAX).

On one side, smaller splitting regions are more efficient. On the other side, arbitrarily removing from simulation particles that otherwise would contribute to the dose biases the results. The optimal situation for a given field is to adjust the size of the splitting region to be slightly larger than the field size. This is done using the option *Fitted to the field size currently set in s2*. However, this criterion can conduct to errors if a PSF is tallied at segment s1 to be used as the source of particles in simulations of s2 and s3 with variable field sizes. There could be cases in which the region irradiated by the field is not completely covered by the splitting region used to tally the PSF. To avoid biasing the dose in these cases, the CROI size must be set with respect to the maximum possible field size ( $40 \times 40 \text{ cm}^2$ ) using the option *Biggest*.

When rotational splitting is selected at s1 while an off-axis field is configured at s2, PRIMO automatically applies the fan splitting technique. This variance-reduction technique has been developed for improving simulation efficiency of off-axis fields [SBB11].

**Warning 4.9** When employing splitting roulette or rotational splitting to tally a phase-space file at segment s1, it is safer to select the option *Size of splitting region* as the *Biggest* to avoid biasing the simulation of fields larger than the splitting region. Select *Fitted to the field size currently set in s2* only when there is certainty that the largest field that will be simulated with the source PSF is the one currently set in the segment s2 configuration.

**Notice 4.6 — Adequate variance-reduction parameters.** The following variance-reduction parameters are a reasonable first choice when attempting a given simulation. For nominal energies below 15 MV (photon mode) it is recommended to use splitting roulette for s1. For nominal energies above 15 MV rotational splitting is usually more efficient. Regarding simulation of s3, a splitting factor of 100 usually works fine. It is advisable to check the estimated time after launching the simulation of s3. If the estimated time with a splitting factor of 100 is exceedingly long then reset the simulation, modify the splitting factor for s3 and launch it again. For an explanation on simple splitting at s3 see section 5.1.1.

#### 4.1.8 Importing external phase-space files

Phase-space files produced with PRIMO or other simulation codes can be imported provided they are saved in the IAEA format [Cap+06]. Phase-space files can be imported in the s1 segment only. Importing a phase-space file is a process similar to producing it by simulation. Previously to importing, a new project must be created and the selected linac must be the same as the one simulated to produce the phase space (with the exception of TrueBeam, see below). Also, the s1 tab segment configuration must be set with the same parameters of the initial beam that were used to create the phase-space file, or an approximation to them. The segment s1 checkbox must be active and the project must be saved. To import the phase-space file, select the option *Import a phase space* in the main menu or main tool bar. In the standard File Open dialog select one or several phase space header files (\*.IAEAheader). The Phase space import dialog (figure 4.19) shows the progress of the importing process. After imported, the phase-space file is incorporated into the project as if it were the result of simulating the s1 segment.

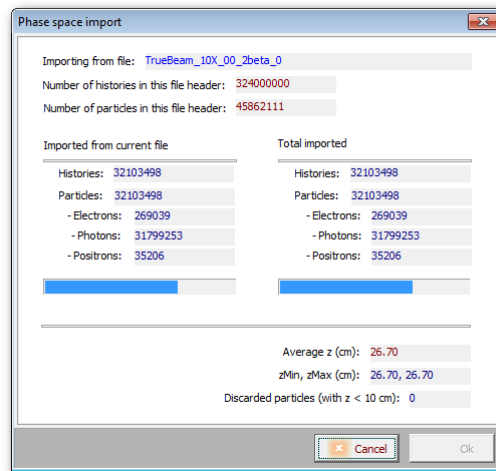


Figure 4.19: Phase space import dialog.

- R** To import a TrueBeam phase-space file, create a project for a Varian Clinac 2100 linac, then set the s\ tab parameters to match those used to simulate the phase-space file (see the Example 03).

**Warning 4.10** When importing an external phase space separated in several files, in the File Open dialog select all the header files belonging to the phase space.

#### 4.1.9 Duplicating a project

The option Save As in the main menu allows to create a copy of the project, including all the imported data and the simulation results. The Save As dialog is the same as the New Project dialog with the exception that the option to select the linac is blocked. Enter a new project ID (different to that of the original project) and path and, optionally, a new name and comments. The copy process could take a considerable amount of time, depending on the size of the project files, especially of the phase-space files. No progress dialog will appear.

## 4.2 Execution

Once the simulation setup is finished and saved the *Run* button in the main window can be clicked and the simulation starts running. The main window disappears and *in lieu* the execution window is presented (figure 4.20).

The execution window reports the progress of the processes requested for the simulation during the setup process. A maximum of 120 processes can be unleashed simultaneously.

The toolbar of the execution window contains 4 buttons as shown in figure 4.21. From left to right these buttons are

- *Run simulation*: this button is not operative during simulation.
- *Change simulation parameters in flight*: allows to change simulation parameters during runtime (section 4.2.1). The button becomes active after all cores have initialized their respective simulations and one minute of simulation time has elapsed for all of them.
- *Reset simulation*: This button allows to reset the simulation, that is, to return to the simulation setup **losing all computed results**.





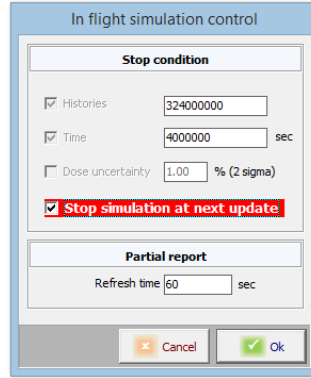


Figure 4.22: In flight simulation control window.

$$\sigma_k = 2.0 \times \sqrt{\frac{1}{\eta} \sum_{j=1}^{\eta} \sigma_{kj}^2}, \quad (4.2)$$

where  $\eta$  is the number of bins in the dose distribution with a dose larger than half the maximum dose ( $d_j \geq d_{max}/2$ ) and the sum is done for those  $j$  bins satisfying the condition.  $\sigma_{kj}$  is the statistical uncertainty of the dose in the  $j$ -th bin for the  $k$ -th process.

In the case a phase space is being tallied the overall window shows the current number of particles tallied in the phase space. The information appearing in the upper panel is refreshed every update interval as it has been specified in the simulation setup. The lower panel (green characters on a black background) shows the log file corresponding to process selected in the upper panel. The log file is generated during execution and contains the simulation input parameters and the output of PENEASYLINAC and PENEASY generated during their execution. An explanation of the PENEASY output can be found in its documentation (section 1.3). Simulation of several fields is done sequentially.

#### 4.2.1 Changing parameters at runtime

Through this dialog it is possible to modify the total simulation time requested, the number of simulated histories and the update interval. Additionally, it is possible to stop the simulation while keeping the already simulated results. The changes take effect at the end of the update interval. For example, if the update interval was set in the setup section as 600 seconds, and the Stop simulation at next update checkbox is marked, the simulation will continue running until the next update interval is reached, in at most 600 seconds. Then simulated results will be collected and presented to the user. This way of stopping the simulation is notoriously different from using the *Reset* button. If the *Reset* button is clicked the simulation stops immediately and all results are lost.

**Warning 4.12** In a multi-field simulation it is not possible to change the stop conditions (by number of histories or time) in flight. In the same manner, once the simulation of the first field has finished, the simulation cannot be stopped by pressing *Stop simulation at next update*, otherwise the relative weights of the fields (in simulated histories per field) would be altered.

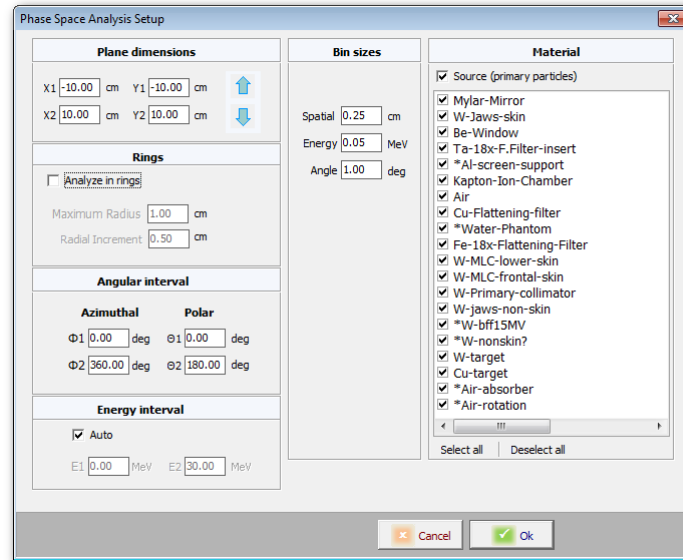


Figure 4.23: Phase space analysis setup dialog.

### 4.3 Analysis of results

There are two main categories of results that can be tallied with PRIMO, namely, phase-space files and dose distributions. Each of these categories has a dedicated tab in the main screen, which are named Phase Space Analysis and Dose Analysis.

#### 4.3.1 Phase-space file analysis

Once the simulation of a phase-space file has been completed, or a phase-space file has been imported, a check mark will appear next to the corresponding hard drive symbol in the Simulation Segments panel. Also the simulated phase-space file will appear in full color in the objects tree instead of being grayed-out.

In order to analyze a given phase-space file, select the Phase Space Analysis tab in the lower part of the main PRIMO window, drag the colored phase-space icon from the objects tree and drop it into the main blank area of the analysis window. Alternatively, right click on the phase-space icon and select the option *Analyze* in the appearing pop-up menu. The Phase Space Analysis setup dialog (figure 4.23) contains several options for filtering the phase space and setting the probability distribution intervals.

Only particles from a rectangular region of the phase-space plane are included in the analysis. This region is determined by the values entered in  $x_1$ ,  $x_2$ ,  $y_1$  and  $y_2$ . It is possible to select the intervals and bin sizes of the angular and energy probability distributions. If the phase-space file has been tallied with PRIMO, then it is also possible to filter the particles by the material where they were produced. This feature has no effect on imported phase-space files tallied with other Monte Carlo codes.

If the option Analysis in rings is checked, then the analysis is performed by subdividing the whole phase space plane into concentric rings centered at the central axis. The number of rings is determined by the values of the maximum radius and the radial increment set in the dialog.

**Warning 4.13** There is a maximum number of bins for the probability distributions cal-

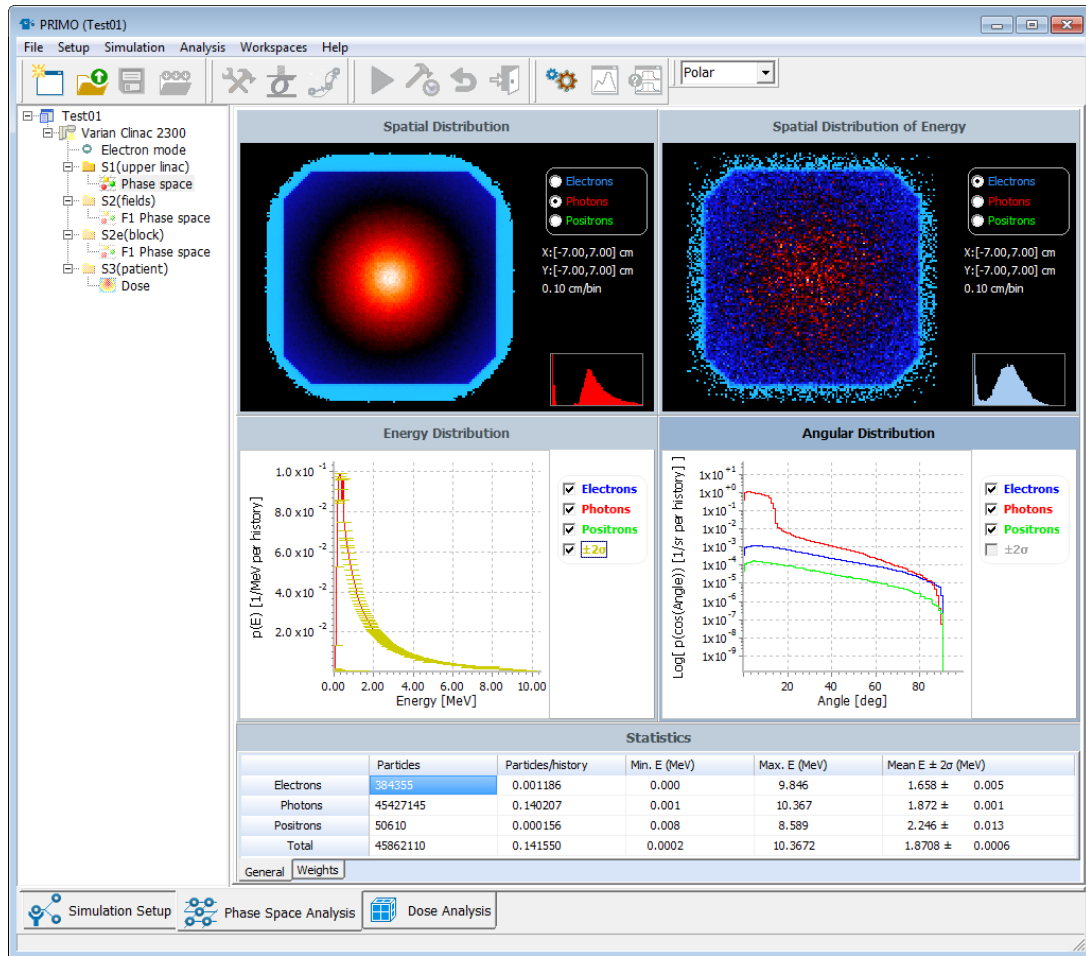


Figure 4.24: Phase space analysis window.

culated in a phase-space analysis. The maximum number of bins allowed for the angular, energy and spatial distributions are  $1.8 \times 10^2$ ,  $5 \times 10^3$  and  $10^6$ , respectively.

Once the analysis parameters are defined, the particles in the phase-space file are read, filtered and the probability distributions calculated. Figure 4.24 shows the analysis window. The images in the upper part are the 2D spatial probability distributions of the particles (left) in units of  $[\text{cm}^{-2} \text{ per history}]$  and the energy (right) in units of  $[\text{MeV cm}^{-2} \text{ per history}]$  in the rectangular region of the phase space set in the configuration. Distributions are separated by kind of particle. Click the radio-button at right hand side of the image to select the desired distribution. Change the color lookup table by using the drop-down list in the main tool bar. To extract horizontal and vertical profiles of the 2D distribution, select the option *Make a profile* in the main menu or main tool bar (figure 4.25). In the Profile of the spatial distribution dialog, drag the horizontal and vertical lines over the image to update the profiles.

The 1D probability distributions of the energy and angle are also separated by kind of particle. In this case, the distributions can be included in, or removed from, the graph by checking or unchecking the box at the right hand side of the plot. The statistical uncertainty ( $\pm 2\sigma$ ) of the distribution can also be plotted by checking the corresponding box.

Some additional tools can be found by right clicking on the title bar of a probability distribution graph. A pop-up menu appears with the following options:

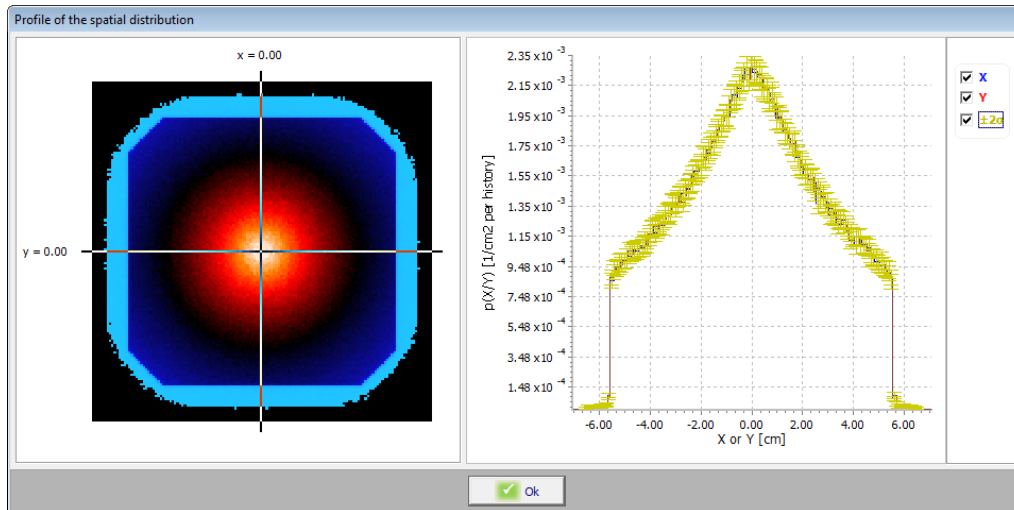


Figure 4.25: Profile of the spatial distribution dialog.

- *Maximize*: Maximize the graph to the whole analysis window.
- *Restore*: Restore a maximized window to its normal size.
- *Copy to clipboard*: Copy the graph to the clipboard as an image.
- *Logarithmic scale*: Show the graph (1D distributions) in a semilogarithmic scale.
- *Normalize to cosine*: Angular distributions are normalized to  $\cos \theta$ , where  $\theta$  is the particle polar angle. This is the normalization by default of the angular distribution.
- *Show as surface*: A 2D distribution is shown as a surface.
- *Save as text*: The (1D or 2D) distribution is presented as a text file that can be saved.

Some statistical data of the phase space (classified per kind of particle) are shown in the table at the bottom of the analysis window. These include the total number of particles in the phase space, the number of particles per history, and the mean energy. Statistical weights are shown in a separated (Weights) tab.

In an analysis performed in rings, separated 1D probability distributions and statistical data are calculated for each ring, i.e. by including only the particles located inside the ring. In this case, a tool bar appears on top of the statistics table containing buttons to change the ring whose results are presented in the analysis window. Also the rings are shown superimposed to the 2D spatial distributions (figure 4.26).

### 4.3.2 Dose analysis

Once the simulation of a dose distribution has been completed, a check mark will appear next to the corresponding hard drive symbol in the Simulation Segments panel. Also the dose icon in the objects tree will appear in full color.

In order to analyze the dose distribution, select the Dose Analysis tab in the lower part of the main PRIMO window, drag the colored dose icon from the objects tree and drop it into the main blank area of the analysis window. Alternatively, right click the dose icon and select the option *Analyze* in the appearing pop-up menu. The 3D dose distribution is shown superimposed to the phantom or CT slices in the transversal, sagittal and coronal planes (figure 4.28). The whole dose volume can be navigated by changing the planes. To change a plane, drag its sliders to the desired position. The planes can also be changed with the mouse wheel. Each plane is identified with a different color and the sliders are of the color of the plane that it represents.

The planes  $xz$ ,  $yz$  and  $xy$  are the transversal, sagittal and coronal planes, respectively. Dose

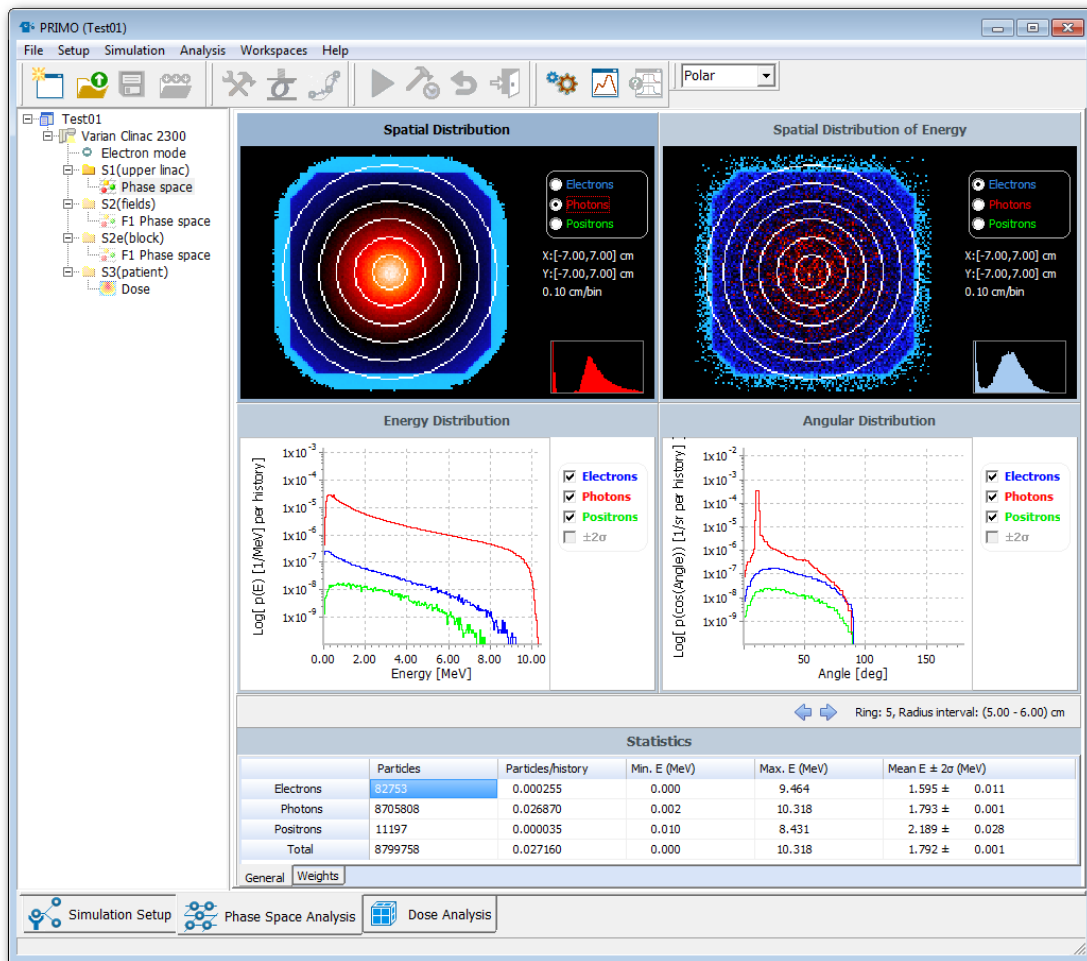


Figure 4.26: Phase space analysis window showing the results of an analysis made in rings. The tool bar on top of the Statistics table allows to navigate the results of the ringed regions.

profiles can be taken along the  $x$ ,  $y$  and  $z$  directions (figure 4.28). In case the dose distribution is calculated in a CT volume dose profiles do not appear by default. Instead dose-volume histograms (DVH) for all structures are created or imported. In case no structure was created the DVH of the whole CT volume is shown. To show dose profiles select the option *Toggle profiles and DVHs* in the pop-up menu that appears by right clicking on the title bar of the window. Other options are:

- *Maximize*: To maximize the selected window to the whole analysis window.
- *Restore*: To restore a maximized window to its normal size.
- *Show dose as color wash*: To toggle between visualizing the dose as a color map (default) or as isodose curves.
- *Interpolate planes*: When the option is active the visualization algorithm uses an interpolation to show the dose distribution, the Hounsfield numbers and the structures according to the window size.
- *Zoom*: To activate the zoom in/out function. To zoom in/out, drag the mouse over the image in the left-up/right-down direction. The point where the three planes intersect is taken as the center of the region zoomed.
- *Restore original size*: To restore a zoomed image to its original size.
- *Inspect dose*: To inspect the dose values in the dose distribution (see Inspecting the dose distribution section below).
- *Saves curves as text*: To save the dose profiles or DVHs in text files. In dose profiles, the coordinates of the point set by default are those of the center of the bin. It can be changed to be the coordinates of the low end of the bin by checking the box *Save the coordinates of the low end of the bin* in the *Save curves as text* dialog.
- *Pan*: To move the zoomed region.
- *Move planes to the origin*: To position the sliders (intersect the three dose planes) at the origin of coordinates.
- *Move planes to dose maximum*: To position the sliders (intersect the three dose planes) at the point of maximum dose.
- *Toggle profiles and DVHs*: To toggle between visualizing dose profiles or DVHs.
- *Differential DVHs*: To calculate differential DVHs instead of cumulative DVHs (default option).
- *Copy to clipboard*: To copy the dose plane or graph to the clipboard as a bitmap.
- *Change field weight*: To change the relative weight of the fields (see next section).

### 4.3.3 Changing the relative field weight

Weight factors can be assigned to fields. This is done post simulation. In a multi-field simulation, a separate dose distribution is generated per field and are integrated to form the total dose distribution for analysis and visualization. Weight factors are applied to the partial dose and its uncertainty in each bin during integration. The total dose  $d_i$  in a bin  $i$  is calculated as,

$$d_i = \frac{1}{N} \sum_{f=1}^F d_{if} n_f w_f, \quad (4.3)$$

where  $N$  is the total of histories simulated,  $n_f$  is the total of histories simulated for the field  $f$ ,  $F$  is the number of fields and  $w_f$  is the weight factor for the field  $f$ . The uncertainty  $\sigma_i$  of the dose in the bin is evaluated as,



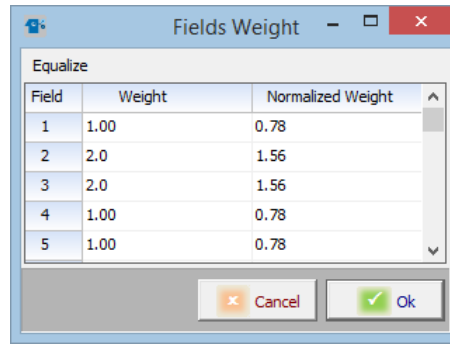


Figure 4.27: The *Fields weight* dialog.

$$\sigma_i = \sqrt{\frac{1}{N^2} \sum_{f=1}^F (\sigma_{if} n_f w_f)^2}, \quad (4.4)$$

where  $\sigma_{if}$  is the uncertainty of the dose in the bin  $i$  for the field  $f$ . Weight factors  $w_f$  are normalized such that,

$$\frac{1}{F} \sum_{f=1}^F w_f = 1. \quad (4.5)$$

The dose distribution is loaded from disk and integrated every time the weight factors are changed. To change the field weights, select the option *Change field weight* in the pop-up menu that appears by right clicking on any title bar of the dose analysis window. The dialog represented in figure 4.27 will appear. The option *Equalize* makes all the weights equal. Weight factors set by the user are normalized to satisfy equation 4.5. Normalized values are shown in the dialog.

#### 4.3.4 Inspecting the dose distribution

The option *Inspect dose* in the pop-up menu (that appears by right-clicking on the title bars of the windows) of the dose analysis windows is useful to know the absolute value of the dose (in units of  $\text{eV g}^{-1}$  per history) at any point of the dose distribution. In the *Inspect dose* dialog (figure 4.29), define the point of inspection by entering its coordinates in the fields  $x$ ,  $y$  and  $z$ . The dose ( $\pm 2\sigma$ ) is displayed at the bottom. The buttons at left hand side allow to inspect the dose at predefined positions, namely, the point of maximum dose, the center of coordinates of the dose distribution and the point of intersection of the three orthogonal planes. The button *Move the planes at this point* displaces the planes to the position given by the values in the fields  $x$ ,  $y$  and  $z$ .

**Notice 4.7** To locate the point of maximum dose in a 3D dose distribution, in the *Inspect dose* dialog, first select the 3D dose maximum option and then *Move planes to this point*. ■

#### 4.3.5 Comparing to experimental data

A calculated spatial dose distribution can be compared to measurements of lateral, diagonal or depth dose curves. This can be done by selecting the option *Compare with experimental data* in the main menu or main tool bar. In the standard *Open File* dialog, select a text file containing



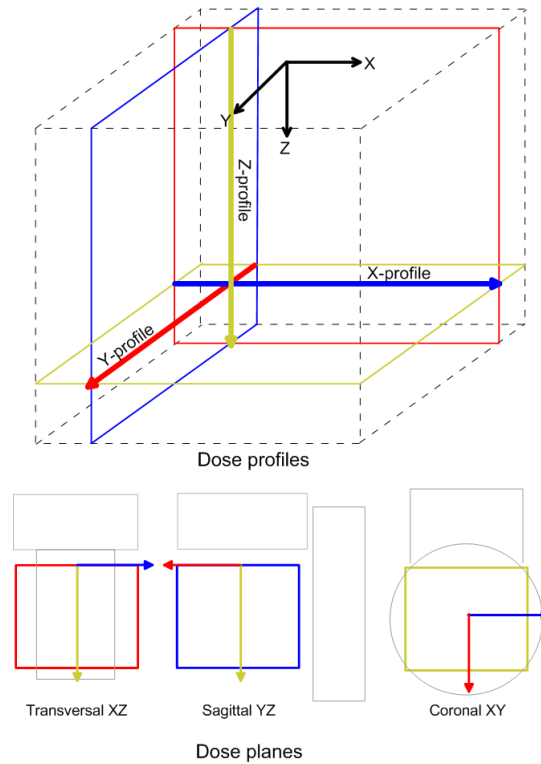


Figure 4.28: Planes of the 3D dose distribution and how the dose profiles are read.

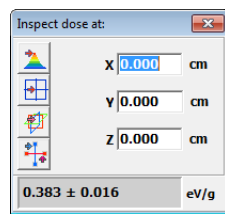


Figure 4.29: Inspect dose dialog.

the experimental data. The file must be formatted as specified in figure 4.30. It must be a plain text file consisting of a list of four data values per line, namely, the three coordinates ( $x, y, z$ ) of the measurement point and the dose. The coordinates of the measurement point must be specified in the same reference system as those of the simulation of the water phantom (refer to section 4.1.2) and in units of cm whereas dose units are relative. Measurement points are not required to be equally spaced. In case they are not equally spaced, the minimum distance between two consecutive measurements is taken as reference and the experimental curve is linearly interpolated so as to obtain a uniform grid of coordinates. The simulated curve is obtained on the same uniform grid of the experimental curve by using tri-linear interpolation. The normalization values for the dose curves can be chosen in a drop-down menu among either the dose at the central axis (default for lateral profiles), or the maximum dose (default for depth-dose curves), or the dose at an arbitrary point. Additionally, the curves can be normalized to the ratio of the integral under the experimental curve to the integral of the simulated curve *Integral ratio*. The integral is taken in the region beyond  $d_e^{\max}$ .

**Notice 4.8** The Normalization drop-down box offers the possibility of not normalizing the dose curves. This option is suitable in case both data sets are in the same units. For example, in the case the ‘experimental’ dose curve was obtained by exporting a dose distribution as a text file previously simulated in PRIMO (see Example 01: ‘Comparison with another simulation’).

The Dose curve comparison dialog (figure 4.31) presents the comparison including the graph of the experimental, calculated and difference curves, a number of parameters calculated from the curves and the gamma analysis (section 4.3.6). Comparisons can be evaluated at an arbitrary point by entering its value in the field Evaluate at or by clicking on the graph area. In both cases a vertical yellow bar (curve cursor) is positioned at that point. This point can also be used as the point of normalization by selecting At current cursor position in the Normalization drop-down box and then pressing the *Apply* button below.

The difference curve is calculated for a position  $p$  as:

$$\Delta d(p) = 100 \frac{d_e(p) - d_c(p)}{d_e^{\max}}, \quad (4.6)$$

where  $d_e(p)$  and  $d_c(p)$  are doses at the position  $p$  of the experimental and calculated curves, respectively, and  $d_e^{\max}$  is the maximum dose of the experimental curve.

The following parameters are reported for each curve:

- *Position of maximum*: The position of maximum dose.
- *Dose at 10 cm depth*: The value of the dose at 10 cm depth (only for depth-dose curves).
- *Dose at 20 cm depth*: The value of the dose at 20 cm depth (only for depth-dose curves).
- *Practical range*: The practical range (valid only for depth-dose curves produced from electron beams) is calculated by performing a linear fit of a segment of the curve defined around the depth of maximum dose gradient and reporting the value where the fitted line intersects the abscissa.
- *Range at 50% of the dose*: The depth at which the dose falls to 50% of its maximum value (valid for depth-dose curves).
- *Left off-axis distance at 50% of dose*: In lateral or diagonal dose profiles, this is the distance from the central axis to the point at which the dose falls to 50% of its maximum value, measured in the direction of the negative  $x$  or  $y$  coordinate.
- *Right off-axis distance at 50% of dose*: In lateral or diagonal dose profiles, this is the distance from the central axis to the point at which the dose falls to 50% of its maximum

```

# -----
# Example: Experimental data file format
# This exemplifies the format of the text
# file used to import lateral and diagonal
# profiles or depth dose curves into PRIMO.
#
# Rules:
# - The file must be a plain text file
#   (ASCII or ANSI encoding).
# - Lines (like this one) beginning with the
#   symbol '#' are considered comments. Any
#   other line is considered as data.
# - Comment lines can be inserted anywhere
#   in the file.
# - Coordinates are expected in units of cm.
# - Dose units are irrelevant (as the
#   comparison is relative).
# - Special characters (such as tabs) must
#   not appear in the data lines.
# - Data must be arranged in columns separated
#   by space characters.
# - The order of columns is this:
#   X[spaces]Y[spaces]Z[spaces]Dose
#
# - All columns are mandatory
# - Coordinate values must be listed in
#   sequential order (ascending or descending).
# -----
# Data
# X (cm) Y (cm) Z (cm) Dose
0.0 0.0 0.000 1.3178E+00
0.0 0.0 0.300 1.3405E+00
0.0 0.0 0.600 1.3721E+00
0.0 0.0 0.900 1.3878E+00
0.0 0.0 1.200 1.3981E+00
0.0 0.0 1.500 1.4091E+00
0.0 0.0 1.800 1.4140E+00
0.0 0.0 2.100 1.4171E+00
0.0 0.0 2.400 1.4196E+00

```

Figure 4.30: Example of text file used to import experimental data for dose comparisons. Format specifications are explained in the file header.

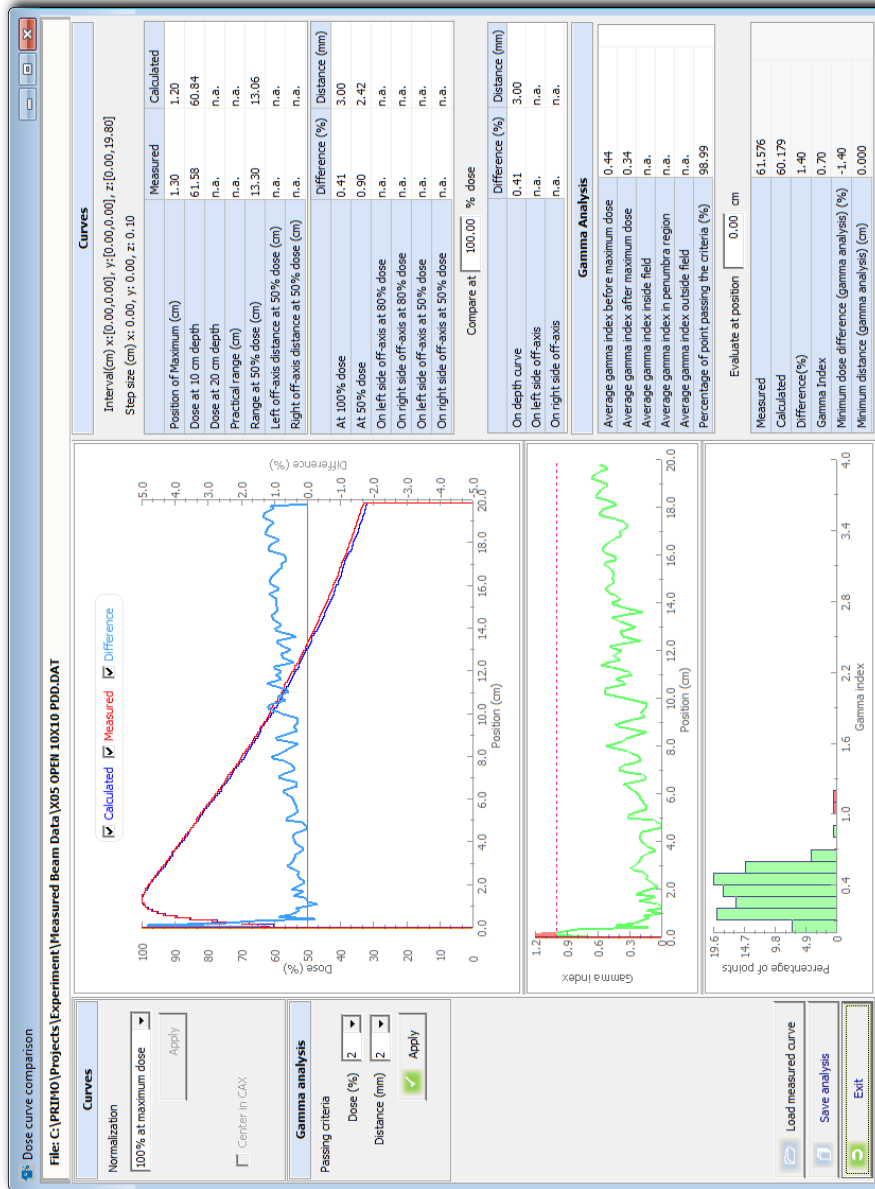


Figure 4.31: The Dose curve comparison dialog. Results from a comparison of an experimental depth-dose curve with a calculated dose distribution. The acceptance criteria for the gamma analysis are set at 2% and 2 mm, for the dose and the distance-to-agreement, respectively.

value, measured in the direction of the positive  $x$  or  $y$  coordinate.

Parameters relative to both curves are the following:

- *The difference at 100% dose:* The relative difference of the dose at the position of  $d_e^{\max}$ .
- *The difference at 50% dose:* The relative difference of the dose at the position of  $0.5d_e^{\max}$ .
- *The distance at 100% dose:* The distance (in mm) between the points of maximum dose.
- *The distance at 50% dose:* The distance (in mm) between the points of 50% of the maximum dose.
- *The difference on the left side off-axis at 80% dose:* The relative difference of the dose at the position of  $d = 0.8d_e^{\max}$ , taken in the negative direction of  $x$ .
- *The difference on the right side off-axis at 80% dose:* The relative difference of the dose at the position of  $d = 0.8d_e^{\max}$ , taken in the positive direction of  $x$ .
- *The distance on the left side off-axis at 80% dose:* The distance (in mm) between the points of 80% dose, taken in the negative direction of  $x$ .
- *The distance on the right side off-axis at 80% dose:* The distance (in mm) between the points of 80% dose, taken in the positive direction of  $x$ .
- *The difference on the left side off-axis at 50% dose:* The relative difference of the dose at the position of  $d = 0.5d_e^{\max}$ , taken in the negative direction of  $x$ .
- *The difference on the right side off-axis at 50% dose:* The relative difference of the dose at the position of  $d = 0.5d_e^{\max}$ , taken in the positive direction of  $x$ .
- *The distance on the left side off-axis at 50% dose:* The distance (in mm) between the points of 50% dose, taken in the negative direction of  $x$ .
- *The distance on the right side off-axis at 50% dose:* The distance (in mm) between the points of 50% dose, taken in the positive direction of  $x$ .

The relative difference of the dose and the distance between points of the two curves can also be evaluated at an arbitrary percentage of the experimental maximum dose by entering it in the Compare at field.

**R** The option Save analysis creates a text file containing all the results of the analysis.

The option Center in CAX produces a shift of the experimental profile such that the distance from the central axis to the points of 50% dose in both, the negative and positive directions of the  $x$  or  $y$  axis, are the same. The shift is applied to the experimental curve and then the simulated curve is extracted again from the 3D dose distribution.

#### 4.3.6 Dose comparison by gamma analysis

The gamma analysis [Low+98] is a method that combines the dose-difference criterion and the distance-to-agreement criterion to compare two distributions. The Dose curve comparison dialog performs the gamma analysis of the experimental curve and the simulated 3D dose distribution. The dose difference is evaluated by exploring the dose distribution in the vicinity of the experimental points. For a given experimental point  $p$  and the dose at that point  $d_e(p)$  the gamma index,  $\Gamma$ , is evaluated as

$$\Gamma = \min \left\{ \sqrt{\left(\frac{\Delta d_i}{\Delta D}\right)^2 + \left(\frac{\Delta s_i}{\Delta S}\right)^2} \right\}, \quad (4.7)$$

where  $\Delta D$  and  $\Delta S$  are arbitrary constants known as the acceptance criteria for the dose difference and for the distance-to-agreement, respectively. The term  $\Delta d_i$  is the difference between  $d_e(p)$  and the simulated dose at a certain point  $p_i$ . The term  $\Delta s_i$  is the distance between  $p$  and  $p_i$ . The minimum of the expression in curly braces is evaluated for the set of points  $\{p_i\}$ . This

set contains the points in the vicinity of  $p$  that extends up to a distance of  $5\Delta S$  (or a maximum of 1.5 cm). The resolution in each spatial direction is enhanced to one fifth of the bin size by tri-linear interpolation of the simulated dose distribution.

The values of *Dose difference (gamma analysis)* and *Distance (gamma analysis)* reported in the Dose curve comparison dialog represent the values of the terms  $\Delta d_i$  and  $\Delta s_i$ , respectively, where the minimum in equation 4.7 is reached.

**Notice 4.9** PRIMO assumes that the imported data for a comparison is the reference data set. Consequently, data from the active simulation is the evaluated data set. Notice that the gamma test is not symmetric with respect to the reference and the evaluated data sets. ■

### Example 01: Photon reference field

Simulation setup and execution  
Analysis of results

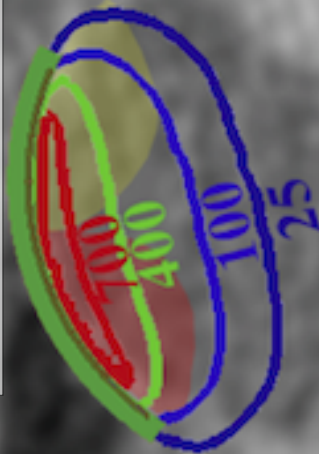
### Example 02: Electron reference field

Simulation setup and execution  
Analysis of results

### Example 03: TrueBeam

### Example 04: Dose calculation in a computerized tomography

Simulation of segment s1  
Field setup  
Construction of the voxelized geometry  
Dose calculation



## 5 — Examples

The examples presented in this chapter are intended to be followed in sequential order. The first example is presented very detailedly, while the following examples only show details on the aspects not previously discussed.

**R** The files with the results of executing the examples presented in this chapter can be downloaded from <http://www.primoproject.net>. The downloaded files are to be uncompressed in the directory `c:\PRIMOexamples`. The file size of each compressed example is the following:

- Example01.zip 971 MB
- Example02.zip 11 MB
- Example03.zip 82 MB
- Example04.zip 836 MB

### 5.1 Example 01: Photon reference field

This example covers the simulation of all segments of a linac in the sequence `s1,s2,s3` (section 3.7) to tally intermediate phase-space files and the dose distribution in a water phantom. The configuration chosen for this example is the following:

**Linac** Varian Clinac 2100 C/D

**Mode** Photon

**Nominal energy** 6 MV

**Field size**  $10 \times 10 \text{ cm}^2$

**MLC** None

**Dose tallying** Water phantom

**SSD** 95 cm

**Bin size**  $0.2 \times 0.2 \times 0.2 \text{ cm}^3$

**Dose tallying volume**  $16.2 \times 16.2 \times 31.0 \text{ cm}^3$

**R** Unless it is explicitly stated all default values have been used.

#### 5.1.1 Simulation setup and execution

After launching PRIMO click the *New Project* button and select the linac model and operation mode in the corresponding window (figure 5.1).



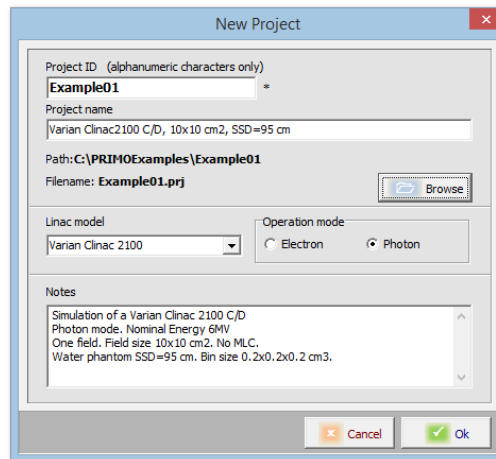




Figure 5.1: New Project window. Notice the linac model and operation mode chosen.


For this simulation project a phase-space file will be tallied at s1. After clicking the configuration button , the Simulation Configuration window is presented (figure 5.2). The simulation time is set to 10 hours (36,000 seconds). This time can be reduced to a few minutes if the aim is to test the functionalities of PRIMO without seeking a low statistical uncertainty.

For simulations of segment s1 of linacs operating in photon mode it is advisable to use either rotational splitting or splitting-roulette. For energies below about 15 MV splitting-roulette is more efficient than rotational splitting. The splitting-roulette technique will be applied to a  $10 \times 10 \text{ cm}^2$  field, therefore the phase-space file produced can only be used for simulations of this field size. Smaller field sizes can also be simulated with the phase-space file tallied with this example provided they are located within the illuminated area of a centered  $10 \times 10 \text{ cm}^2$  field. For a phase-space file to be used on any field size the Biggest option, which is slower to simulate, must be chosen (figure 5.3). Access to the variance-reduction options is done by clicking the corresponding button  on the main PRIMO window. The default beam parameters given in panel Segment Setup, tab s1 are used for this simulation. In tab s2 the default field, a centered  $10 \times 10 \text{ cm}^2$  field size is also left unmodified.

**Notice 5.1** Since the phase-space file generated at s1 occupies about 5 GB of disk space it is not distributed with the example. ■

Finally, the characteristics of the water phantom where the dose will be tallied are also defined at this step by clicking tab s3. The binned region is defined as a parallelepiped of  $16.20 \times 16.20 \times 31.00 \text{ cm}^3$  with bin size equal to  $0.2 \times 0.2 \times 0.2 \text{ cm}^3$ . Notice that the center of a column, along the  $z$ -direction, of bins is located along the central axis of the beam (figure 5.4). It must be stressed that the binned region defines the size of the calculation grid, not the size of the water phantom. The size of the water phantom is always chosen much larger than the field size.

Once the project is saved the simulation can be launched by pressing the *Run* button (figure 5.5).

After the run has finalized the *Exit* button  becomes active. Click it to return to the main PRIMO window. Mark the checkbox for s2 and click the *Configuration* button to proceed with the next segment (figure 5.6). Notice that we now choose the number of histories as the stopping condition since we are interested in using all particles tallied in the phase-space file computed during the simulation of s1. The number of histories appearing in the Histories field correspond



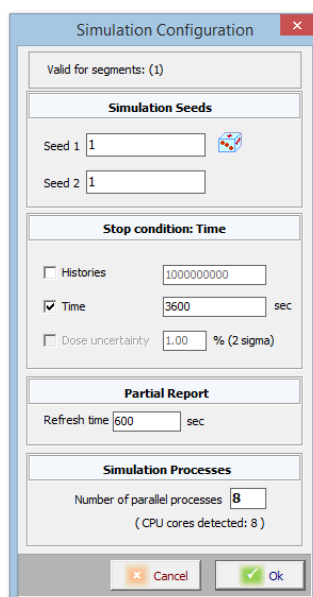


Figure 5.2: Simulation Configuration window. Notice that the only stopping condition is the simulation time. The update interval is set to 600 seconds. Eight simulation cores are chosen.

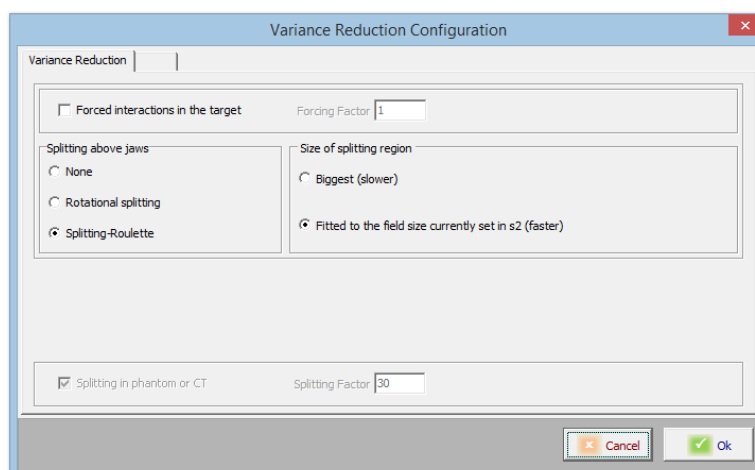


Figure 5.3: Variance reduction configuration window. Splitting-roulette fitted to the field size is chosen.

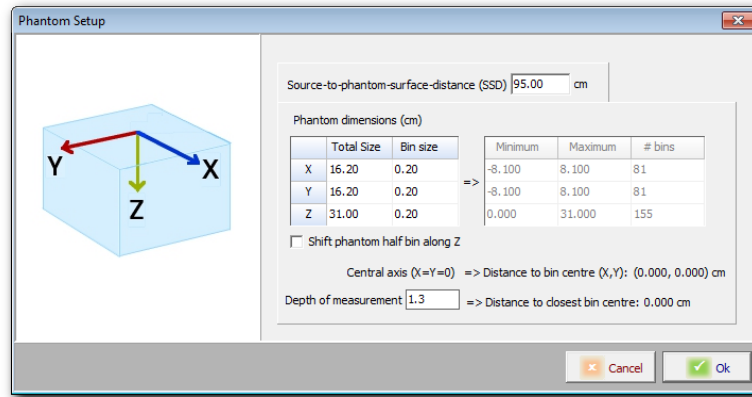


Figure 5.4: Patient Model window, Phantom tab. SSD is set to 95 cm.

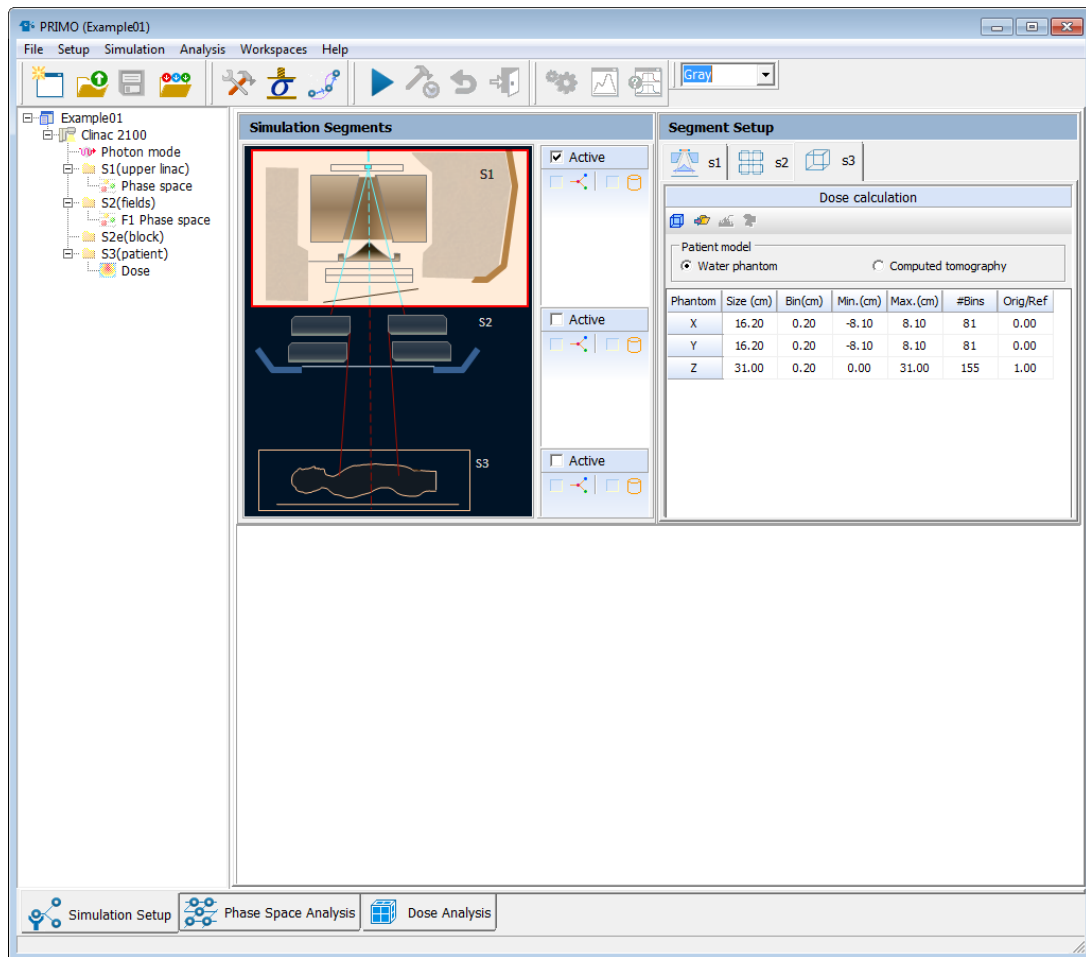


Figure 5.5: Main PRIMO window just before launching the simulation of s1.

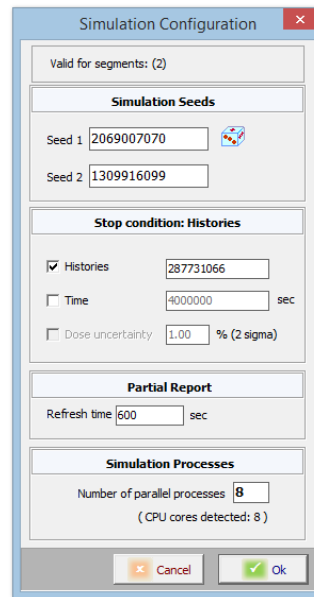


Figure 5.6: Simulation Configuration window prepared for simulating s2.

to the number of histories simulated in s1. The seeds of the random number generator should not be changed, since they are the latest seeds used in the previous segment. In that way it is guaranteed that the sequence of pseudo-random numbers of s2 does not overlap with that used during the simulation of s1. Notice that it is not necessary that all segments use the same number of computing cores.

No user-selectable variance-reduction techniques are available for s2. They are automatically chosen and adjusted by PRIMO. After saving the project the simulation can be launched (figure 5.7).

Once the simulation of s2 is completed the procedure to resume the main PRIMO window is the same described above. To reduce the statistical uncertainty in the estimation of the absorbed dose distribution we apply particle splitting to the phase-space file tallied at the downstream end of s2. By clicking the button for variance reduction the corresponding window is shown (figure 5.8). For this case a splitting factor of 200 is entered. The adequate value of the splitting factor can be found by trial and error. A simple method to estimate an appropriate value of the splitting factor is the following:

1. Simulate s3 using a splitting factor equal to 1.
2. Let us call  $\Delta$  the obtained average statistical uncertainty as reported by PRIMO.
3. Define  $n = \Delta/\delta$ , where  $\delta$  is the desired statistical uncertainty.
4. Let us call  $f$  the ceiling of  $n^2$ , that is,  $f = \lceil n^2 \rceil$ . Then  $f$  is the splitting factor required for the simulation of s3.
5. Re-run s3 using the recently calculated splitting factor  $f$ .

However, bear in mind that:

- The simulation of s3 with a splitting factor  $f$  will last on average a time  $ft$ , where  $t$  is the time that it took to simulate s3 with a splitting factor of 1 (assuming all the other conditions equal).
- It is possible that despite using the splitting factor  $f$  the desired statistical uncertainty is not reached. If this happens it is because the tallied phase-space file is not large enough to attain the requested uncertainty under the given simulation conditions. Possible solutions

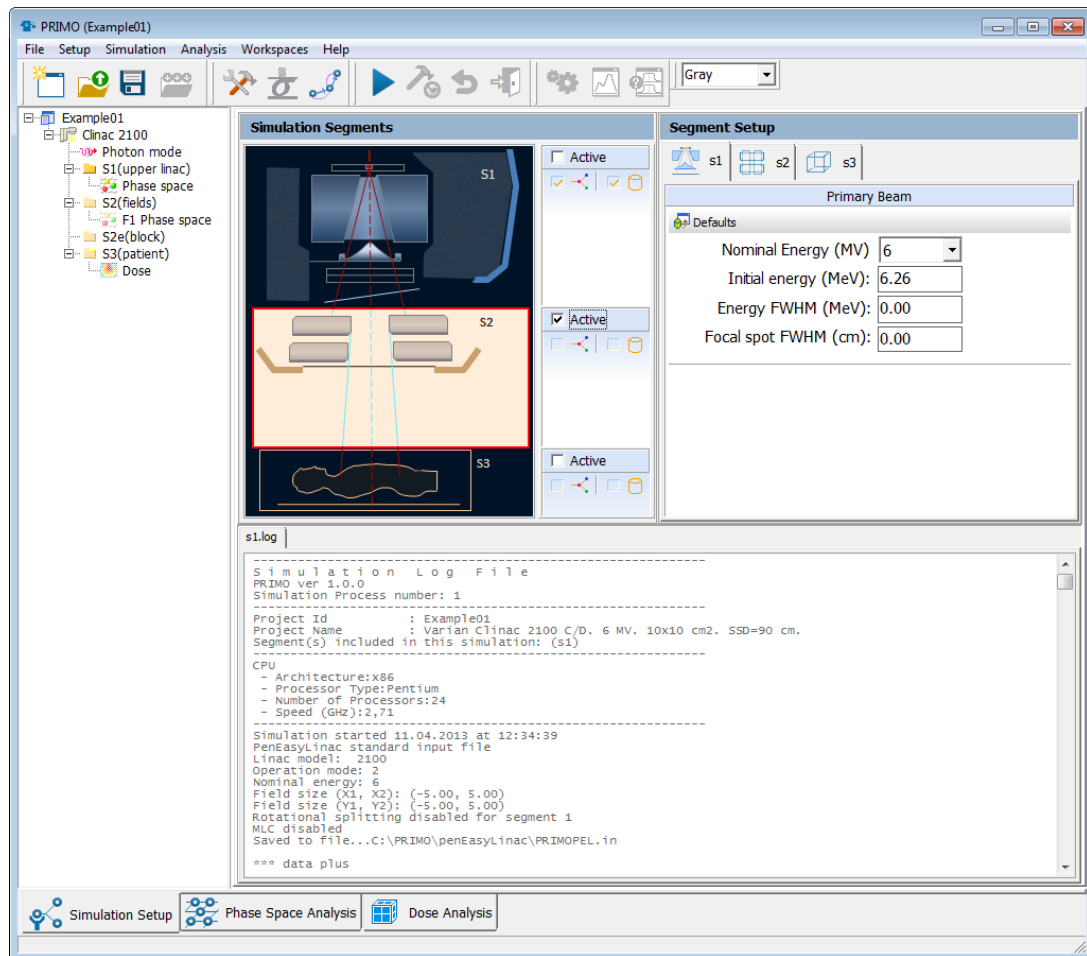


Figure 5.7: Main PRIMO window just before launching the simulation of s2.

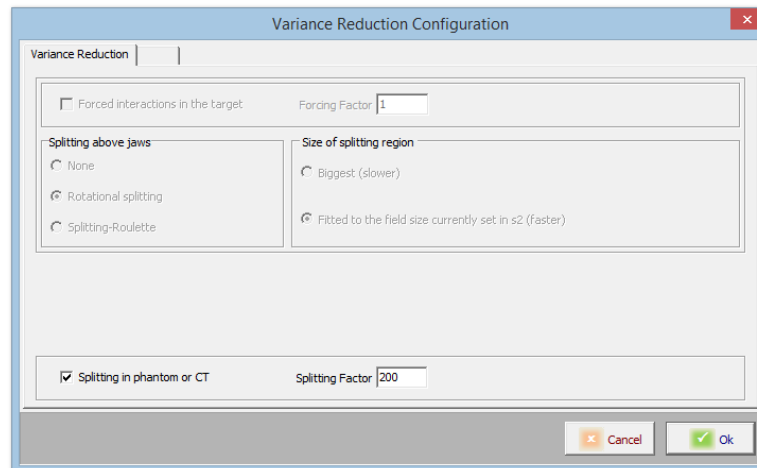


Figure 5.8: Variance reduction for s3.

to this problem are to tally a larger phase-space file or to make the simulation conditions at s3 less demanding (e.g., use a larger bin size).

**Notice 5.2** All statistical uncertainties reported by PRIMO are given to 2 standard deviations ( $\pm 2\sigma$  rule).

After saving the project the simulation of s3 can be launched (figure 5.9).

Once the simulation of s3 is completed PRIMO will return to its main window by clicking the open door button, as usual.

### 5.1.2 Analysis of results

In this section the data analysis screens are depicted. In this example the analysis is shown at the very end of the simulation of all segments for clarity reasons. However, the analysis of the results produced at each segment can be done immediately after the simulation of the segment has been completed.

For analyzing the phase-space files tallied at s1 and s2 the Phase Space Analysis tab is clicked in the main PRIMO window and the desired phase-space file is dragged from the logical tree and dropped in the empty area to its right (figures 5.10 and 5.12).

On each of the graphical panels in the Phase Space Analysis tab it is possible to right click the figure and further options are offered to the user. Below the graphical panels two tabs give statistical information about the analyzed phase-space file, including data about the statistical weight of tallied particles.

By clicking the plot button in the main PRIMO window it is possible to plot profiles of the spatial distribution of particles in the phase-space file (figure 5.11). The drop-down menu in the upper part of the main PRIMO window, which by default shows *Gray*, allows to change the color scale used for showing maps. A *Rainbow* scale better reveals the spatial extend of the phase-space file.

Simulation details from the phase-space files can be viewed by right-clicking the phase-space file icon in the logical tree and then left-clicking the *Properties* button.

The dose distribution can be analyzed by selecting the Dose Analysis tab and then dragging the dose icon from the logical tree and dropping it in the blank area. After reading the tallied spatial dose distribution dose maps in the transverse, sagittal and coronal planes are shown.

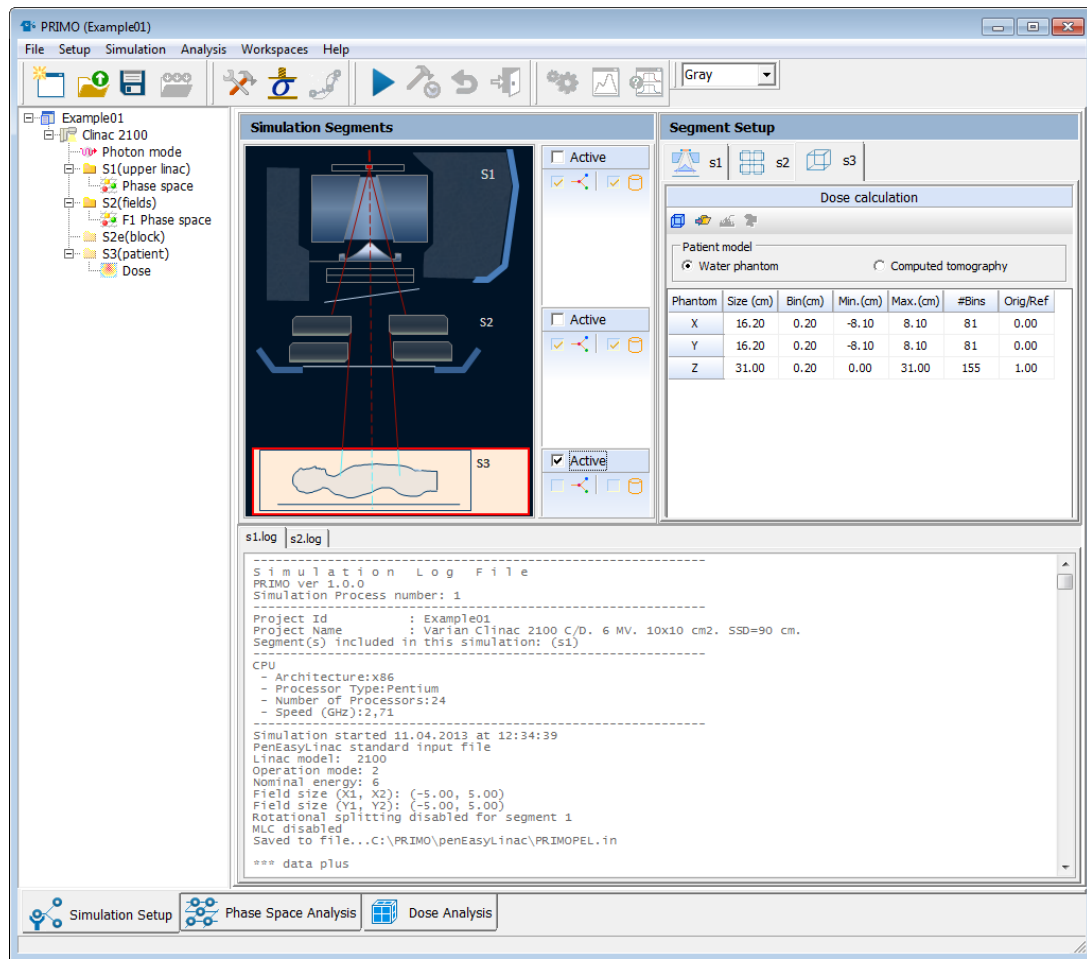


Figure 5.9: Main PRIMO screen just before launching the simulation of s3. Segments s1 and s2 appear as simulated and both with a phase-space file tallied.

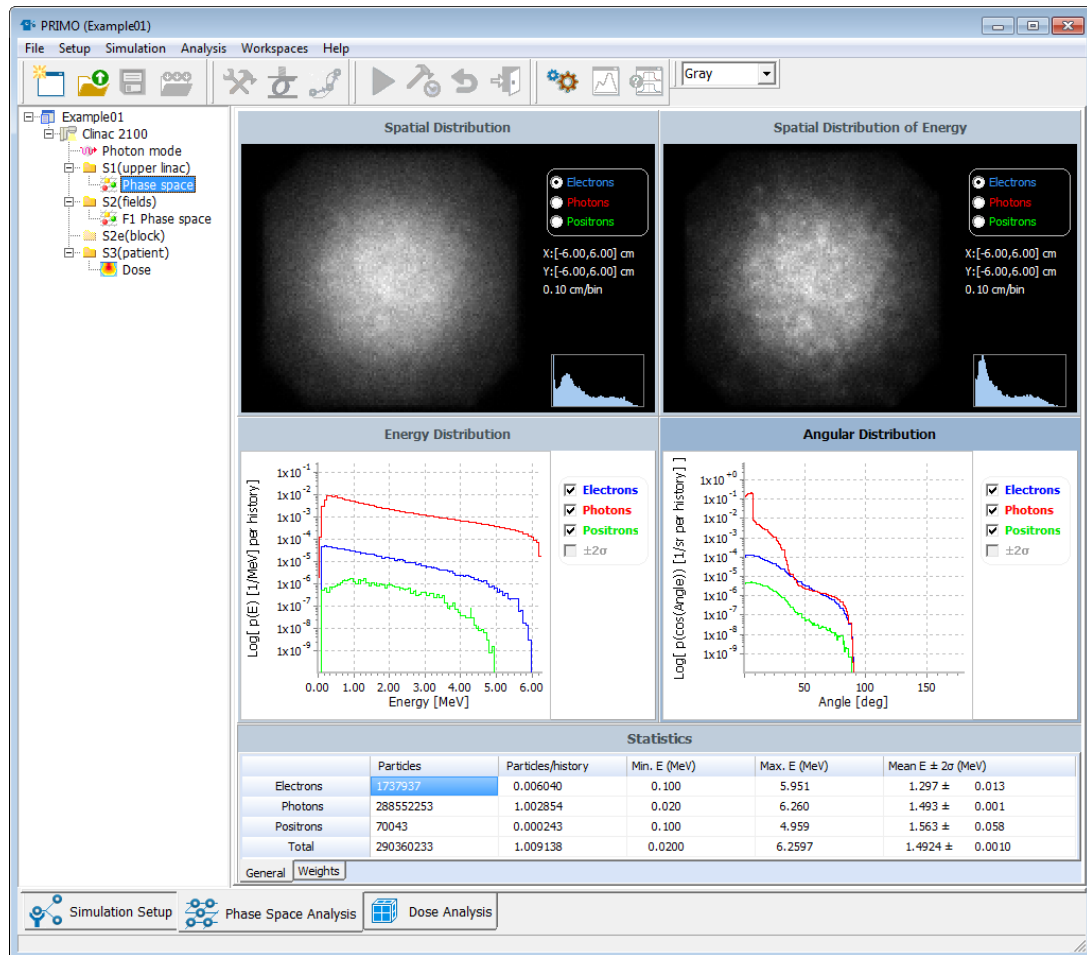


Figure 5.10: Analysis of the phase-space file produced at the downstream end of s1. For this figure the spatial and energy distribution maps show the electron contamination. The energy and angular distribution plots are depicted in semi-log scale (option available by right clicking on the figure).

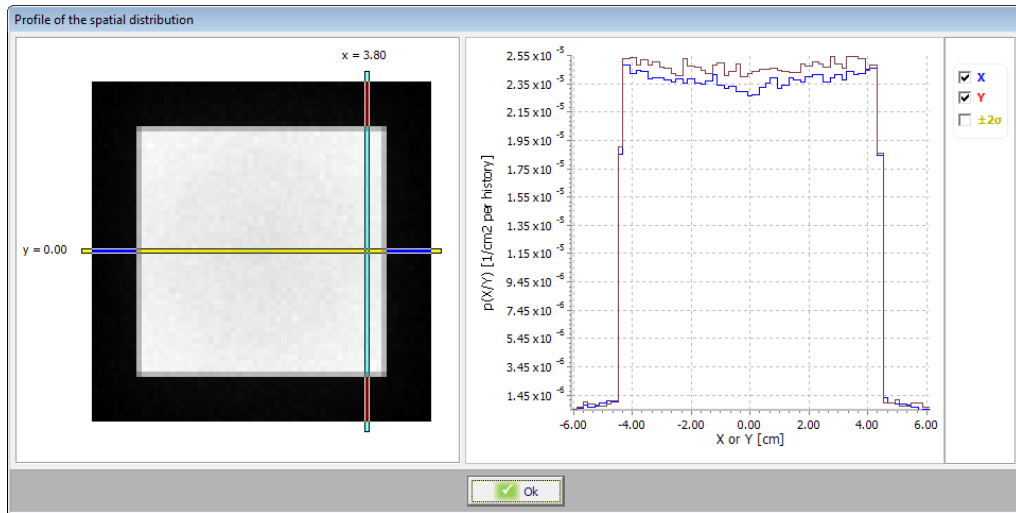


Figure 5.11: Profiles of the spatial distribution of photons in the phase-space file obtained with the simulation of s2.

Additionally, dose distributions along  $x$ ,  $y$  and  $z$  are plotted. The position of these variables used for the profiles can be modified by dragging the handles on each of the aforementioned dose maps.

In the Dose Analysis window it is also possible to right click over any of the four graphical panels in order to obtain menus with further options. For example, it is possible to change the view of the dose maps from color wash to isodoses (figures 5.13 and 5.14).

### Comparison with experiment

For the simulation of s1 the default primary beam parameters were used, that is, a monoenergetic pencil beam electron source of 6.26 MeV. PRIMO has built-in analysis tools for comparing experimental dose profiles with the simulated spatial dose distribution. An experimental percentage depth dose and a lateral profile measured on a Varian Clinac iX are found inside the directory `Experiment` distributed with the file `Example01.zip`. To load these profiles into PRIMO for comparison purposes click the *Compare* button on the main window. It must be stressed that the primary beam parameters of the simulation have not been tuned for the particular linac used for measuring the experimental data. The reason for this is twofold. First, we are interested in assessing how PRIMO shows the discrepancies between the two datasets. Secondly, we show how the default beam parameters are a reasonable choice for most linacs. In this case, almost 90% and more than 95% of the analyzed data points exhibited a gamma index (section 4.3.6) smaller than 1 with a passing criterion of 1%–1 mm in the depth dose and lateral profile curves, respectively (figures 5.15 and 5.16).

For importing the experimental data the button for dose curve comparison in the main PRIMO window is clicked while this window is showing the simulated dose maps. Then the experimental percentage dose distribution located in the directory

`c:\PRIMOexamples\Example01\Experiment` is loaded (figure 5.15).

Similarly, the same procedure is followed for loading the experimental lateral profile (figure 5.16).

It is possible to export most of the plotted curves as text files. The exported text files can be used with software for data analysis, publication quality plotting, etc. All exported data files



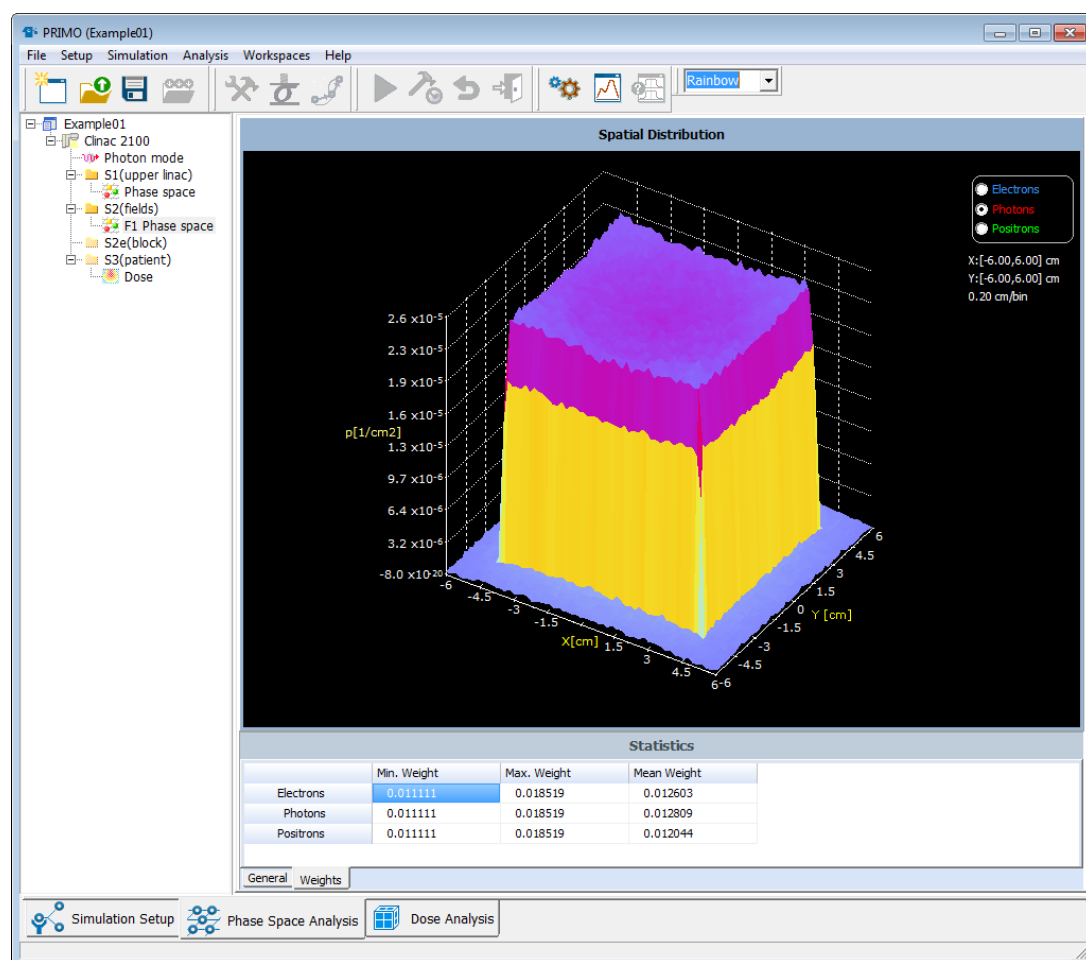


Figure 5.12: Analysis of the phase-space file produced at the downstream end of s2. For this figure the spatial distribution map is shown as a surface. The panel containing the surface plot has been maximized. The figure can be rotated on the screen using the mouse (left button pressed).

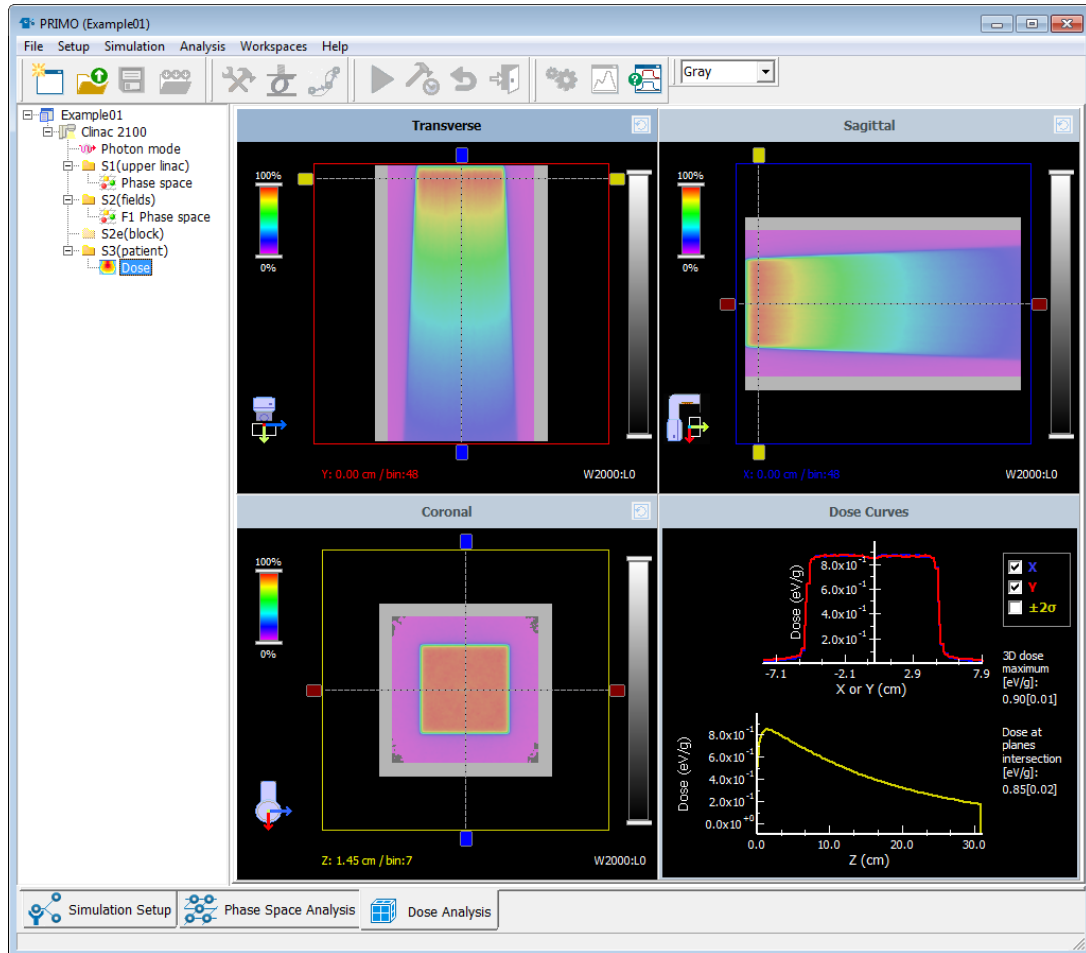


Figure 5.13: Dose maps shown with color wash.

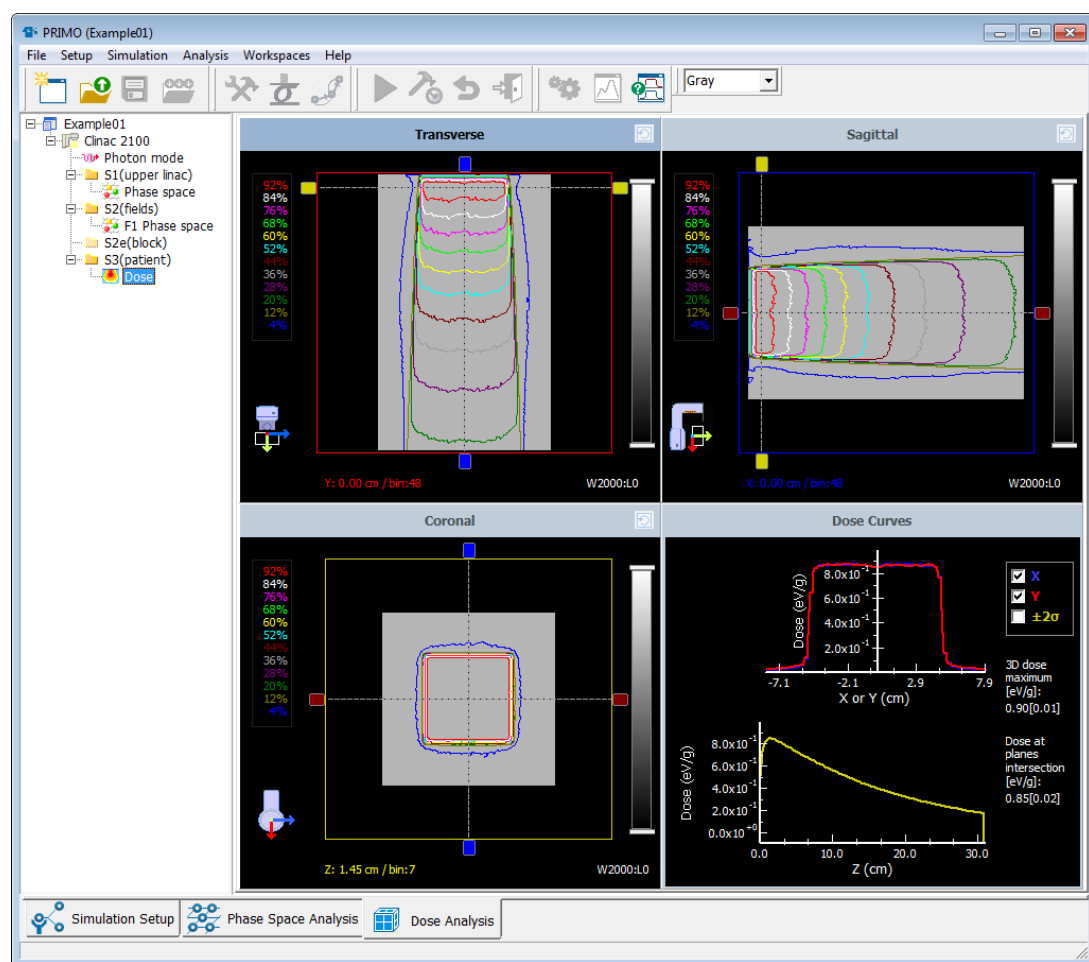


Figure 5.14: Dose maps shown with isodose curves. This option is accessible through the menu obtained by right clicking on any of the three dose maps.

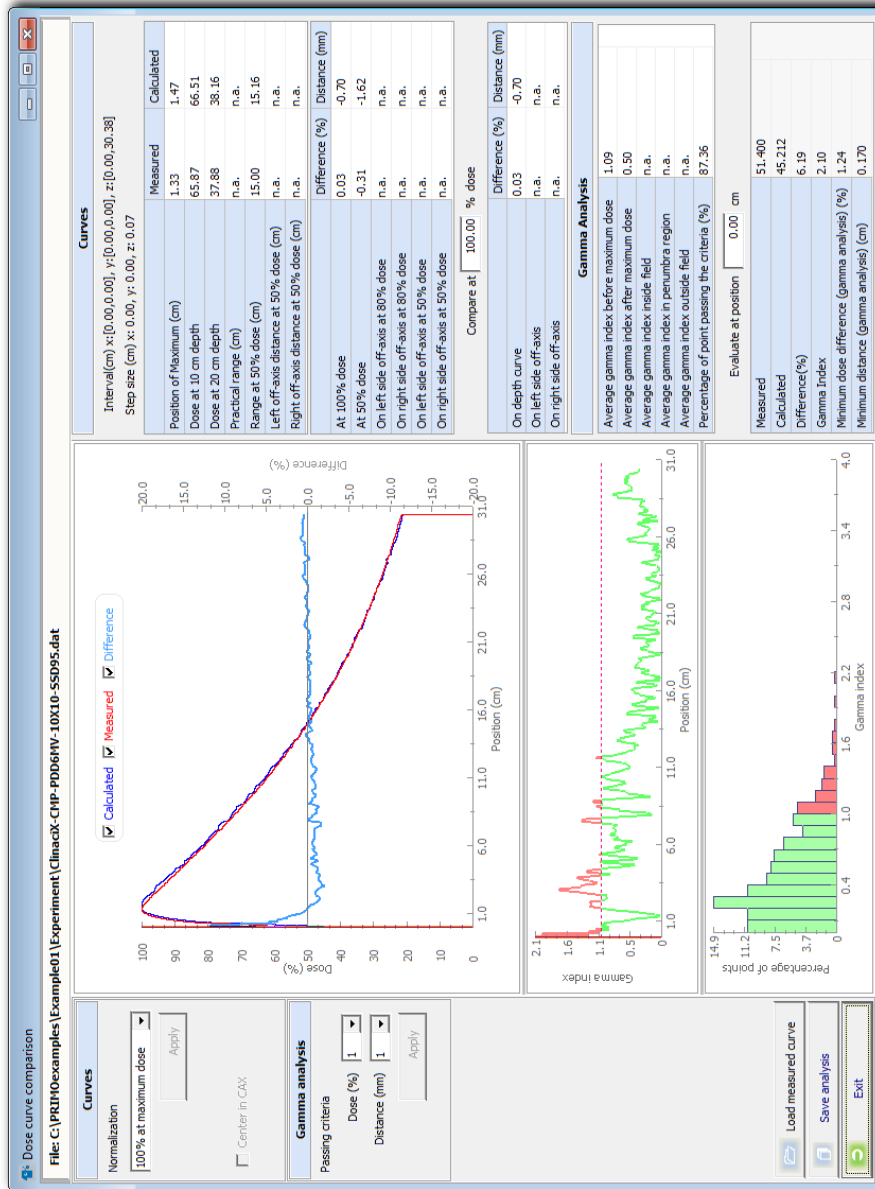


Figure 5.15: Comparison of experimental and simulated percentage depth doses.

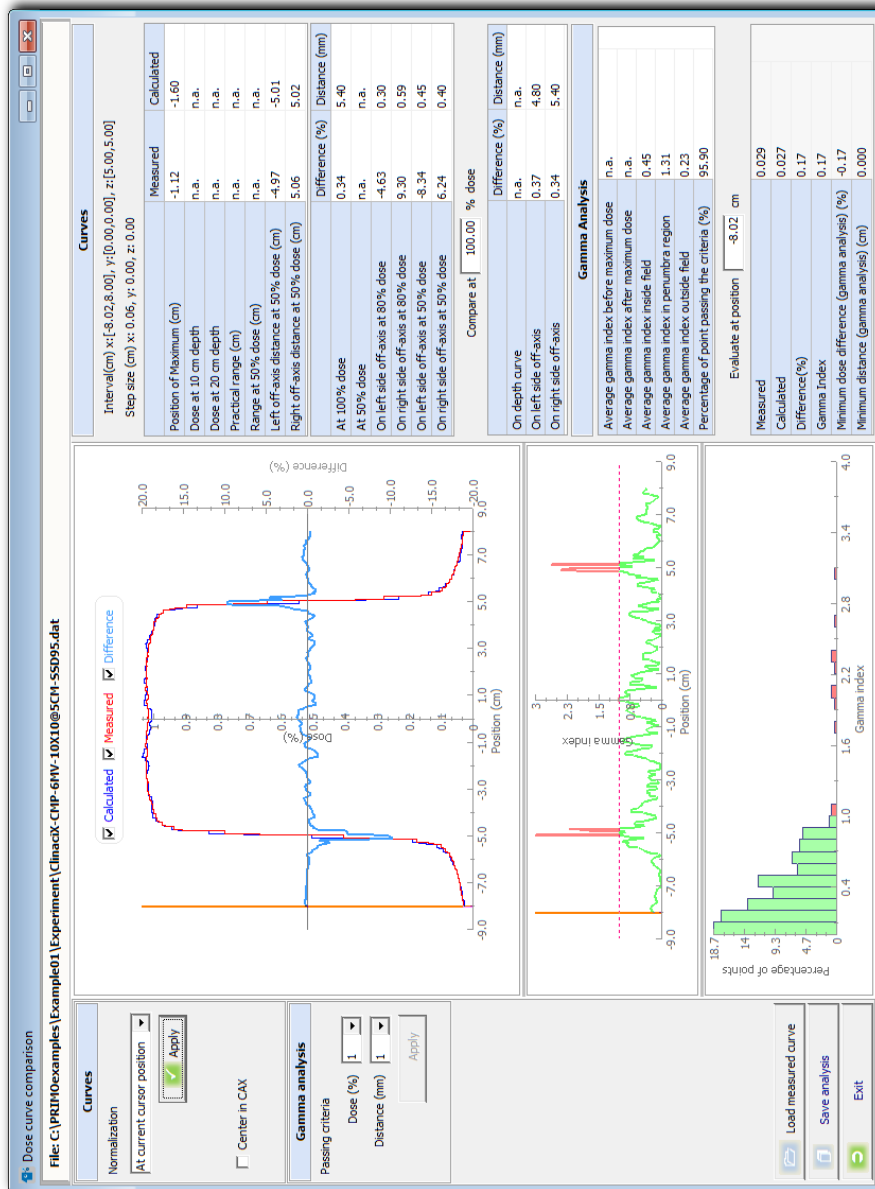


Figure 5.16: Comparison of experimental and simulated lateral profiles.

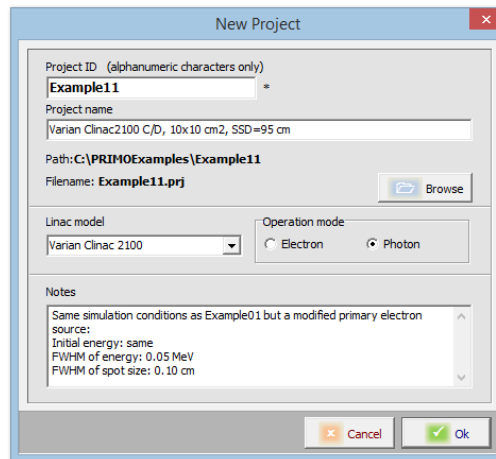


Figure 5.17: New Project window for the simulation with new primary source parameters.

follow the syntax of the Gnuplot program<sup>1</sup>.

- R The names assigned to exported data files must have the extension `.dat` if these files are to be later read by PRIMO.

### Comparison with another simulation

It is possible to compare dose profiles from two simulations. In order to do so the desired profiles must be exported from one simulation as data files. Then these exported profiles will be read by another simulation as *experimental* data.

To exemplify this feature another simulation of the same linac under the same conditions was run and named Example011. The only difference between Example01 and Example011 is that the latter has a primary electron source defined with a Gaussian distribution of the energy and a finite focal spot size (figures 5.17 and 5.18).

The simulation process followed for Example011 is the same that has been already described. Once the dose distribution has been tallied and it is being inspected, it is necessary to right click over the dose profiles shown in the lower-right panel of the Dose Analysis tab (figure 5.19). It is possible to save the depth dose curve and the lateral profiles along  $x$  and  $y$  (figure 5.20). The exported files must have the `.dat` extension.

When this step is completed the simulation Example01 is opened again using the *Open* button in the main PRIMO window.

- R The Example011 files are not distributed in the PRIMO project website. However, the exported profiles can be found in the directory `Experiment` under the names of `Simulated-PDD.dat`, `Simulated-LatX.dat` and `Simulated-LatY.dat`.

The final step is to import the simulated data as it has been done for the experimental data in section 5.1.2 (figures 5.21 and 5.22).

**Notice 5.3** PRIMO assumes that the imported data for a comparison is the reference data set. Consequently, data from the active simulation is the evaluated data set. Notice that the gamma test is not symmetric with respect to the reference and the evaluated data sets. ■

<sup>1</sup><http://www.gnuplot.info>

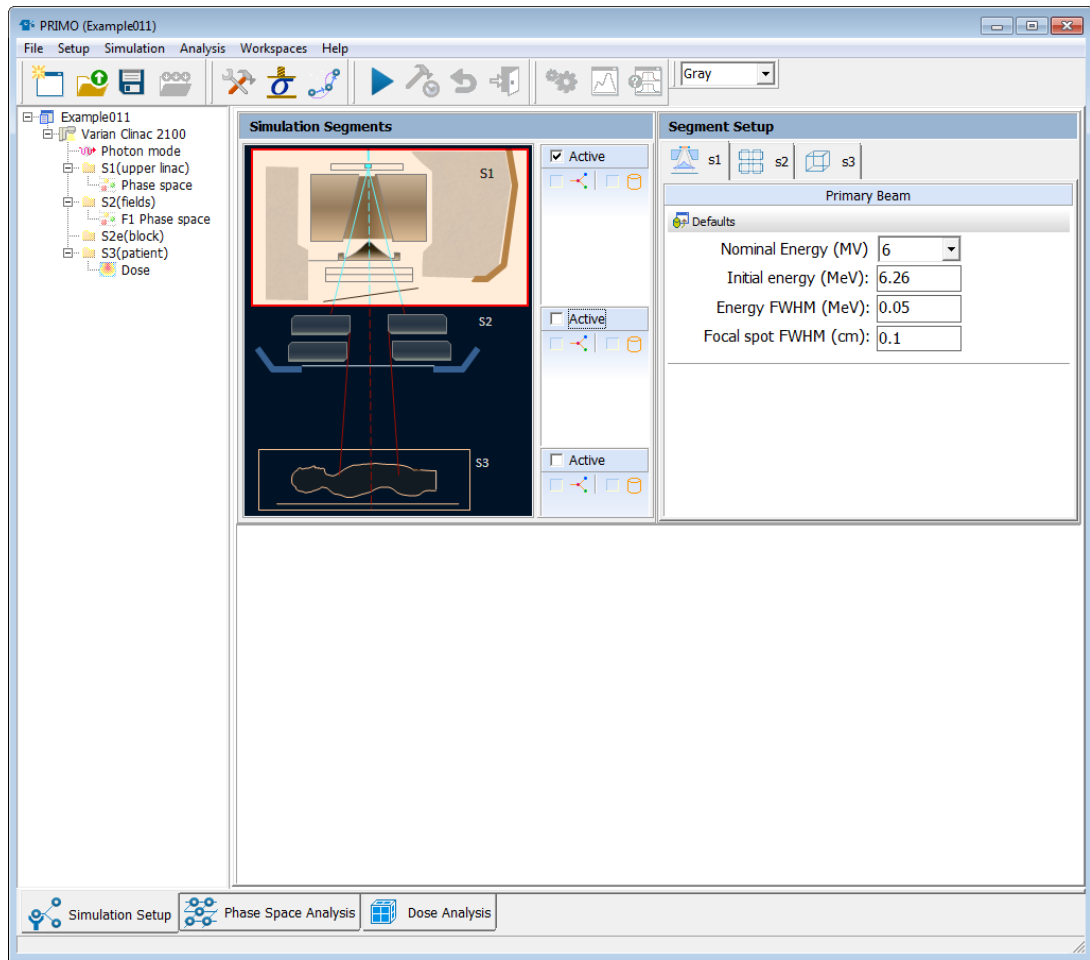


Figure 5.18: Main PRIMO screen before launching the simulation of s1 with the new primary source parameters.

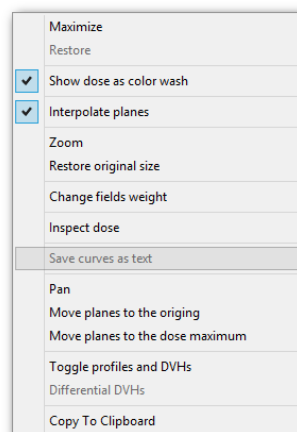


Figure 5.19: Menu obtained when the dose profiles are right-clicked. The mouse pointer is showing the option to save curves as text files.

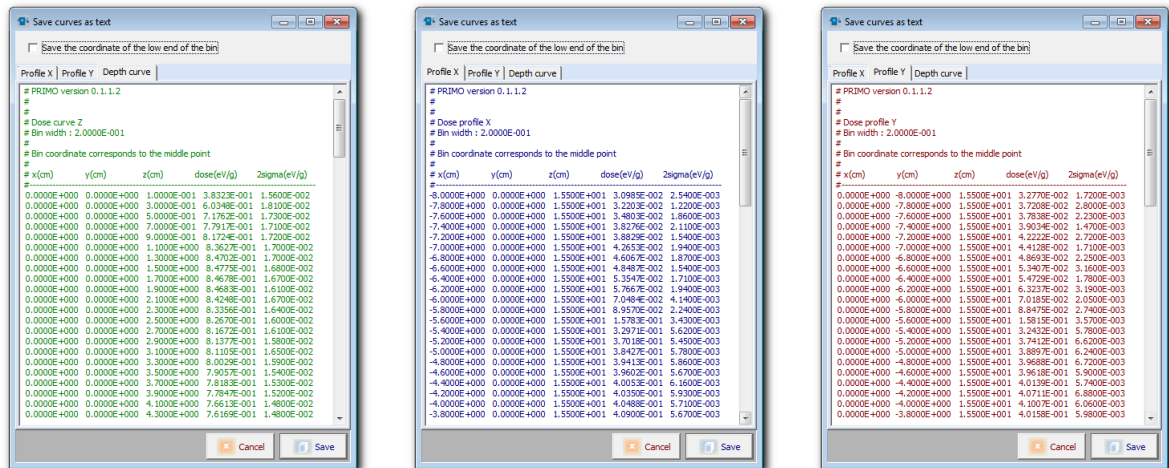


Figure 5.20: Screens showing how to export dose profiles as text files.

## 5.2 Example 02: Electron reference field

This example covers the simulation of an Elekta SL15 signac in electron mode. The nominal energy is 15 MeV and the field is conformed by the  $20 \times 20 \text{ cm}^2$  electron applicator.

Simulation of s1 in any linac operating in electron mode is usually very fast. It is so fast that sometimes it does not pay off tallying a phase-space file for that segment. Instead, it could be more efficient to simulate the whole linac at once. Whether or not is more convenient to simulate each segment separately is something to be decided by the user depending on the particular problem under consideration.

In this example we shall simulate the linac in a single simulation step and in a second step we shall tally the absorbed dose in a water phantom. Thus, the simulation will be performed in the form (s1 + s2 + s2e, s3) (section 3.7).

**Notice 5.4** In order to reduce the file size of Example02.zip the phase-space file of this example is not included among the distributed files. ■

### 5.2.1 Simulation setup and execution

The simulation conditions entered in the simulation setup screens are the following:

**Linac** Elekta SL15 (in PRIMO Elekta SL)

**Mode** Electron

**Nominal energy** 15 MeV

**Initial energy** 15.10 MeV

**Energy FWHM** 0.8 MeV

**Focal spot size FWHM** 0.7 cm

**Field size**  $20 \times 20 \text{ cm}^2$  electron applicator

**SSD** 95 cm

**Bin size**  $0.2 \times 0.2 \times 0.2 \text{ cm}^3$

**Dose tallying volume**  $26.2 \times 26.2 \times 12.0 \text{ cm}^3$  (water phantom)

**Stop condition** Time, 25,200 seconds<sup>2</sup>

**Refresh time** 600 seconds

<sup>2</sup>This example has been simulated for a long time for the purpose of this manual. Users can obtain clinically acceptable uncertainties in a fraction of this time.



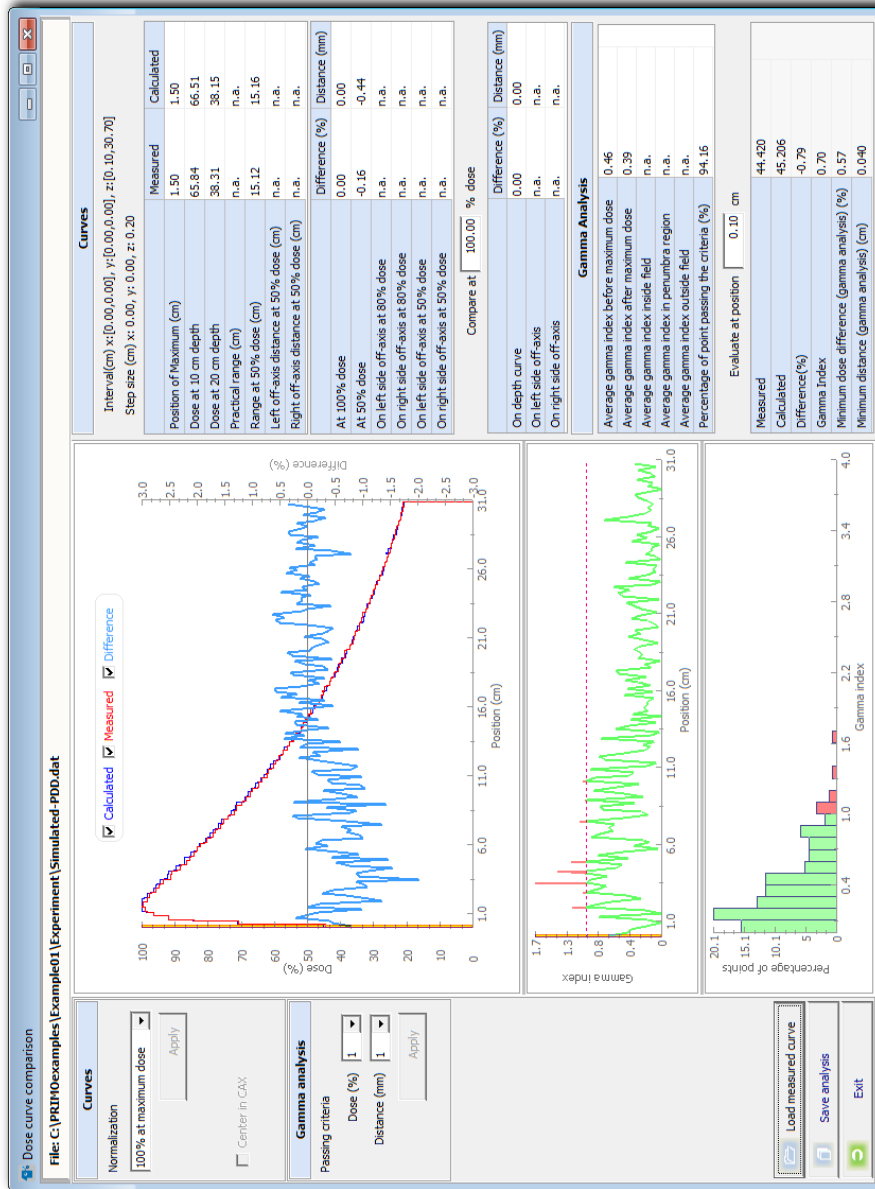


Figure 5.21: Depth dose profile comparison between simulations Example011 and Example01.



Figure 5.22: Lateral dose profile along x. Comparison between simulations Example011 and Example01.

**Number of parallel processes** 12 cores<sup>3</sup>

**Variance reduction for s3** Splitting with factor 25

- R** In order to obtain a tallying binned region with a column of bins centered with respect to the central beam axis choose an arbitrary bin size in each direction and set the total size of the binned region as the bin size times an arbitrary odd number. In this example the binned region size is  $26.2 \times 26.2 \text{ cm}^2$  and the bin size  $0.2 \times 0.2 \text{ cm}^2$ .
- R** There are no user-selectable variance-reduction techniques for segments s1,s2,s2e in electron mode.

**Notice 5.5** The maximum number of bins in a water phantom is  $10^7$ . ■

Varian and Elekta linacs do not use the jaws in the same way when irradiating in electron mode with an electron applicator. Varian linacs have preset positions of the jaws for every combination of applicator size and nominal energy. These preset positions, far apart from the beam path, are used in the simulation. Elekta linacs use the jaws to modify the beam. The field conformed by the jaws is similar to the field conformed by the electron applicator. The position of each jaw for every combination of applicator size and nominal energy is adjusted while commissioning the linac, since small changes in the positions of the jaws have an effect in the dose distribution. For that reason, PRIMO allows jaws to be moved when using electron mode with an applicator. The program suggests for each combination of applicator size and nominal energy a reasonable position that must be further adjusted in order to match the dose distribution of the simulated Elekta linac.

**Notice 5.6** Jaw positions for Elekta linacs in electron mode must be adjusted by the user for each combination of applicator size and nominal energy in order to match experimental dose profiles. ■

The following positions of the jaws were used in the simulation:

**X1** 11.7 cm

**X2** 11.7 cm

**Y1** 13.0 cm

**Y2** 13.0 cm

Notice that the field defined by these values is not square.

The Elekta recommendation for SSD when measuring dose profiles in electron mode is  $95 \text{ cm}^4$ . However, the lowermost surface of all Elekta applicators is also at 95 cm from the primary source. In order to avoid contact of the water with the electron applicator the experimental measurements are performed at a SSD slightly larger. For the same reason, the phase-space file tallied at the downstream end of s2e is positioned at  $z = 95.1 \text{ cm}$ . The minimum distance allowed between the phase-space file and the surface of the water phantom is 0.1 cm. Consequently, the minimum value that can be entered for SSD is 95.2 cm.

**Notice 5.7** Due to geometrical constrains the minimum distance allowed in PRIMO between a phase-space file and the surface of a water phantom is 0.1 cm. ■

<sup>3</sup>Users running this example by their own should adapt the number of processors so as to not exceed the maximum number of processing cores available in their computer.

<sup>4</sup>For further information see section 2.2.3.1 of “Elekta Digital Linear Accelerator. Customer Acceptance Tests”, 1024313 01, Elekta Limited, UK, March 2012

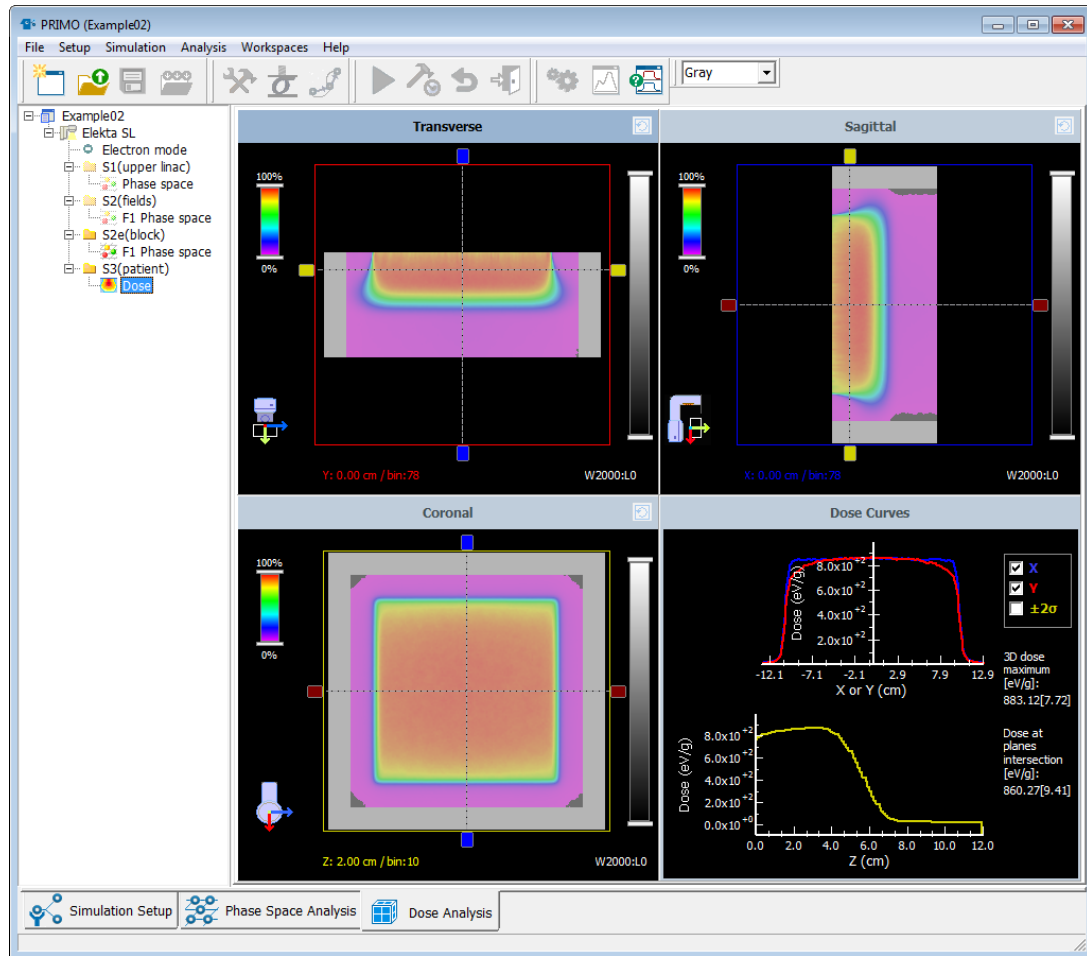


Figure 5.23: Dose maps and dose profiles obtained for the simulation of an Elekta SL15 operating in electron mode. The nominal energy is 15 MeV. The electron applicator size is  $20 \times 20 \text{ cm}^2$ .

### 5.2.2 Analysis of results

The analysis procedure is the same as described for the previous example. Figure 5.23 shows the dose maps and the dose profiles obtained. It can be seen that the  $x$  and  $y$  lateral profiles are slightly different due to the different aperture of the corresponding jaws.

## 5.3 Example 03: TrueBeam

This example presents the simulation of a Varian TrueBeam linac operating in free flattening filter mode at 6 MV. The geometrical details of the TrueBeam linac corresponding to the s1 region are undisclosed by the vendor. Instead, Varian gives access to TrueBeam owners to a set of phase-space files tallied at the downstream end of s1. These phase-space files are recorded according to the IAEA format and, therefore, PRIMO can read them and simulate the radiation transport downstream. Since the phase-space files are proprietary confidential information from Varian Medical Systems they are not distributed in the example. The example only includes the absorbed dose distribution tallied in a water phantom obtained from the simulation of the aforementioned phase-space files.

The scope of this example is to illustrate the steps for simulating a TrueBeam linac using

external phase-space files. These steps are the same in case of simulating any other external phase-space file written in IAEA format.

**Notice 5.8** Users willing to simulate this example by their own must have access to the TrueBeam phase-space files distributed by Varian. ■

- Ⓡ After unzipping the Example03.zip file inside c:\PRIMOexamples, the project files will be found in c:\PRIMOexamples\Example03\TB6FFF40\.

For the simulation of TrueBeam it is necessary to start a new project choosing the Clinac 2100 C/D in photon mode as the linac to be simulated. The nominal energy chosen must be either 6 or 10 MV, depending on the nominal energy of the phase-space file.

- Ⓡ To simulate a TrueBeam a Varian Clinac 2100 C/D must be chosen.

It is recommended to set the conditions for s2 and s3 before actually starting the simulation. For this example the simulation parameters were set to:

**Linac** TrueBeam (in PRIMO Varian Clinac 2100 C/D)

**Mode** Photon

**Nominal energy** 6 MV

**Field size** 40 × 40 cm<sup>2</sup> electron applicator

**SSD** 100 cm

**Bin size** 0.2 × 0.2 × 0.2 cm<sup>3</sup>

**Dose tallying volume** 46.2 × 46.2 × 31.0 cm<sup>3</sup>

**Stop condition** 10<sup>9</sup> number of histories<sup>5</sup>

**Refresh time** 600 seconds

**Number of parallel processes** 12<sup>6</sup>

**Variance reduction for s3** Splitting with factor 25

Only s1 must be checked as active. After saving the project, the button *Import PSF* becomes active. After clicking this button a browser window will appear. The phase space of TrueBeam is distributed in several files. The user can choose to simulate as many files as desired. For this example all files were selected.

**Notice 5.9** Only phase-space files tallied at the downstream end of s1 can be imported. ■

Once the phase-space files have been imported, the program shows s1 as if it had been simulated in PRIMO. The simulation of the rest of the segments proceeds as usual. Figure 5.24 shows the dose map obtained.

**Warning 5.1** The phase-space files distributed by Varian for the simulation of TrueBeam linacs do not contain a field called the incremental shower number for each tallied particle. Therefore, the uncertainties estimated by PRIMO are only an approximation whose accuracy is unknown (section 3.5).

In this example we use a diagonal lateral profile to compare the simulated dose distribution with the experimental data. In the Dose Analysis tab click the *Compare* button to import the dose profile stored in the Experiment directory within the example. PRIMO identifies, by means

<sup>5</sup>This arbitrarily large number guarantees that all histories of the distributed phase-space file at s1 are used.

<sup>6</sup>Users running this example by their own should adapt the number of processors for not exceeding the maximum number of processing cores available in their computer.

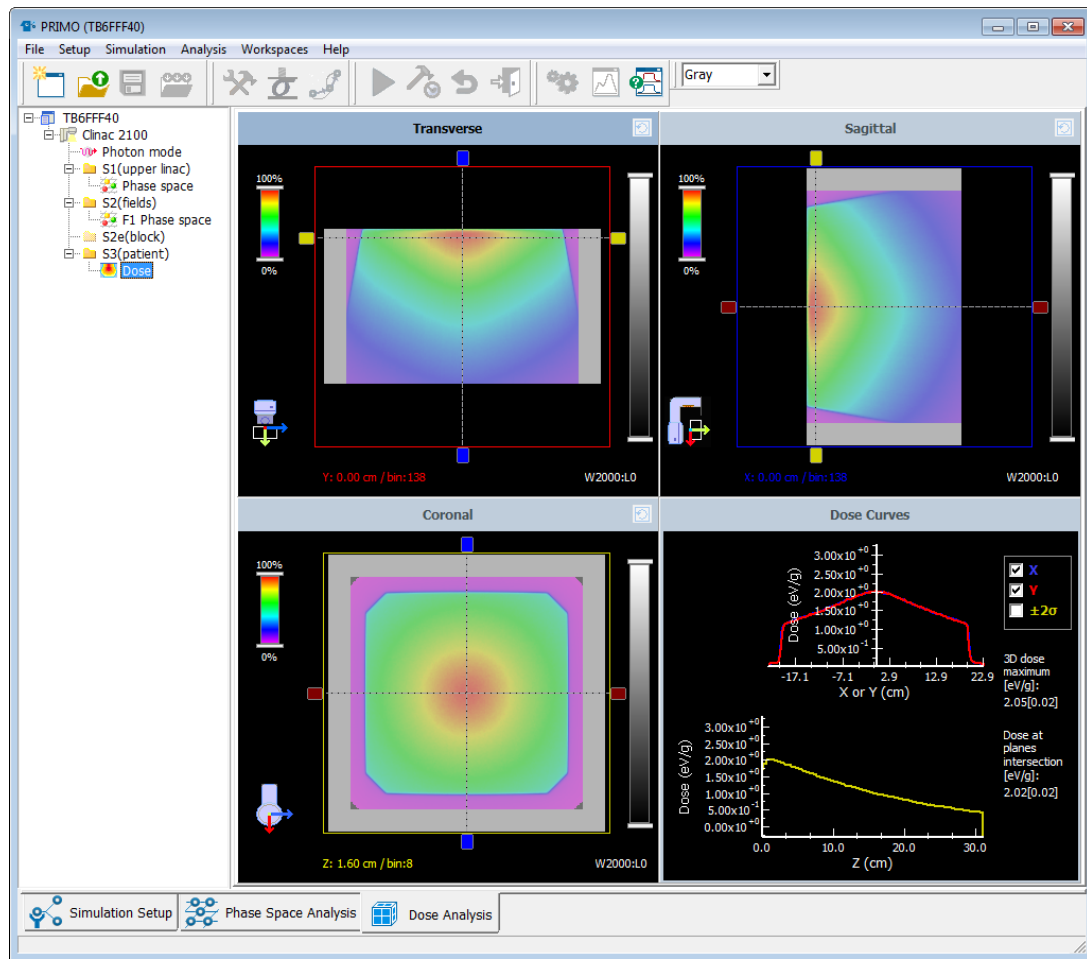



Figure 5.24: Dose maps and dose profiles obtained for the simulation of a TrueBeam linac operating in free flattening filter mode at 6 MeV. The field size is  $40 \times 40 \text{ cm}^2$ .

of the point coordinates, that the data correspond to a diagonal and performs the comparison (figure 5.25).

**Warning 5.2** Phase-space files have been removed from the distributed data in this example. Although PRIMO shows the phase-space file icons in the logical tree as available, do not try to analyze them since dragging them into the Phase Space Analysis tab will produce an error.

 PRIMO can compare simulated data with experimental diagonal profiles.

## 5.4 Example 04: Dose calculation in a computerized tomography

In this example the creation of a typical radiotherapy plan for the irradiation of the whole brain with two parallel-opposed fields is described. Since the CT study of the patient and the delineated structures are not distributed with the example, users will not be able to repeat the steps related to data import. Also, the phase-space file of the segment s1 is not included in the files of this example, so, although it appears in the *Project Tree* as active, dragging it to the phase-space analysis window will produce an error message.

### 5.4.1 Simulation of segment s1

The segment s1 was simulated to create a phase-space file with the following conditions:

**Linac** Varian Clinac 2300

**Mode** Photon

**Nominal energy** 6 MV

**Stop condition** Histories

**Number of histories** 500,000,000

**Variance reduction method** Rotational splitting<sup>7</sup>

### 5.4.2 Field setup

Prior to field setup it is necessary to import the CT volume. This is done in the setup tab s3, selecting the option *Import a CT volume*. The next step is to create the structures. In this case, the structures were delineated in the Eclipse Treatment Planning System (Varian Medical Systems), saved in a DICOM file and imported into PRIMO.

Two fields were added using the option *Add new field* and configured using the option *BEV Edit*. The isocenter and jaws positions were set according to the shape of the PTV (in this case the whole brain). The configuration of the fields is shown in figure 5.26.

The segment s2 was simulated to produce two phase spaces (one per field). All particles in the source phase space were simulated (approximately  $3.1 \times 10^8$ ). The result of the analysis of the phase-space file of field 1 is shown in figure 5.27.

### 5.4.3 Construction of the voxelized geometry

The voxelized geometry is created by associating a material and a mass density to each voxel in the CT volume. This is done in the s3 tab with the option *Calculate densities and materials*. Four materials were considered for this example, namely, air, soft tissue, inner bone and compact bone. The default CT scanner calibration curve was used (figure 5.28).

<sup>7</sup>Although it had been more efficient for this energy to use splitting-roulette, rotational splitting was chosen to show that is also a suitable variance-reduction technique.

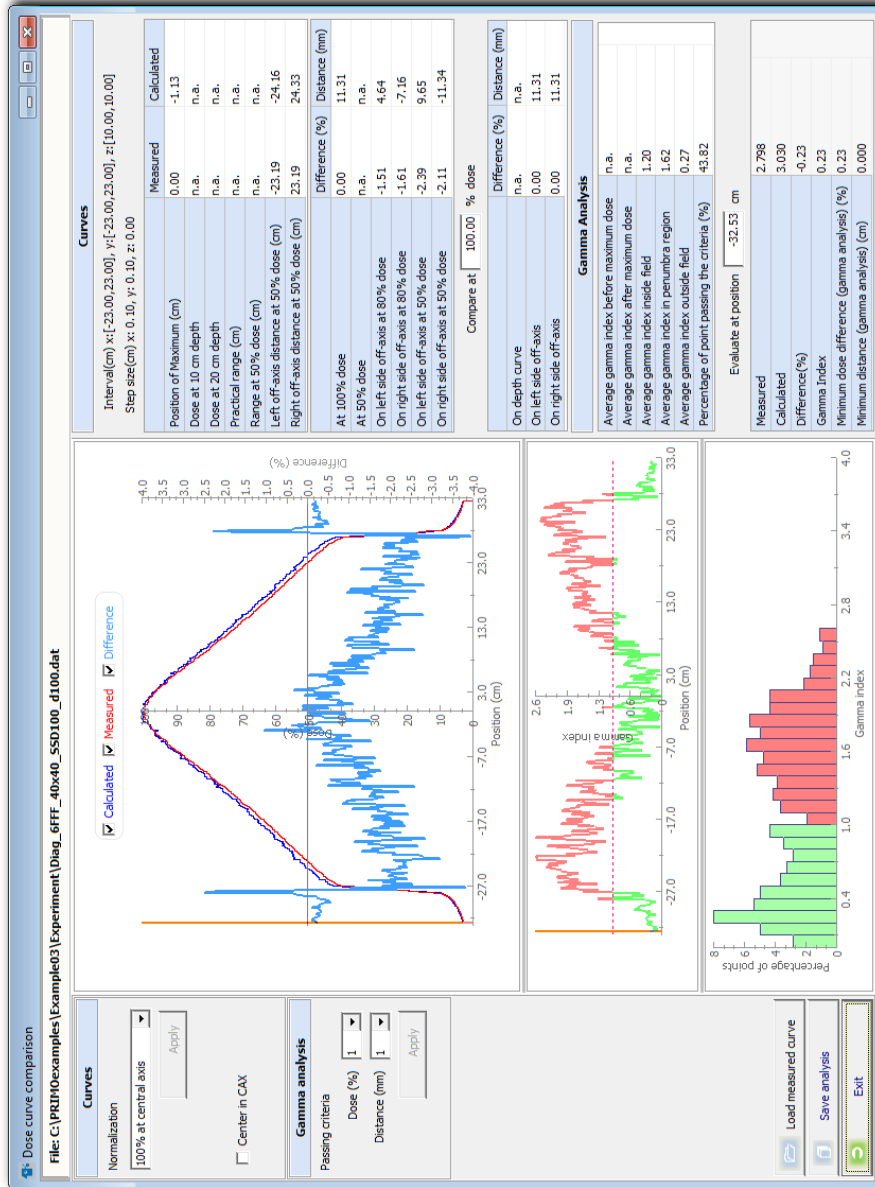


Figure 5.25: Comparison of the experimental and simulated data through a diagonal lateral profile.



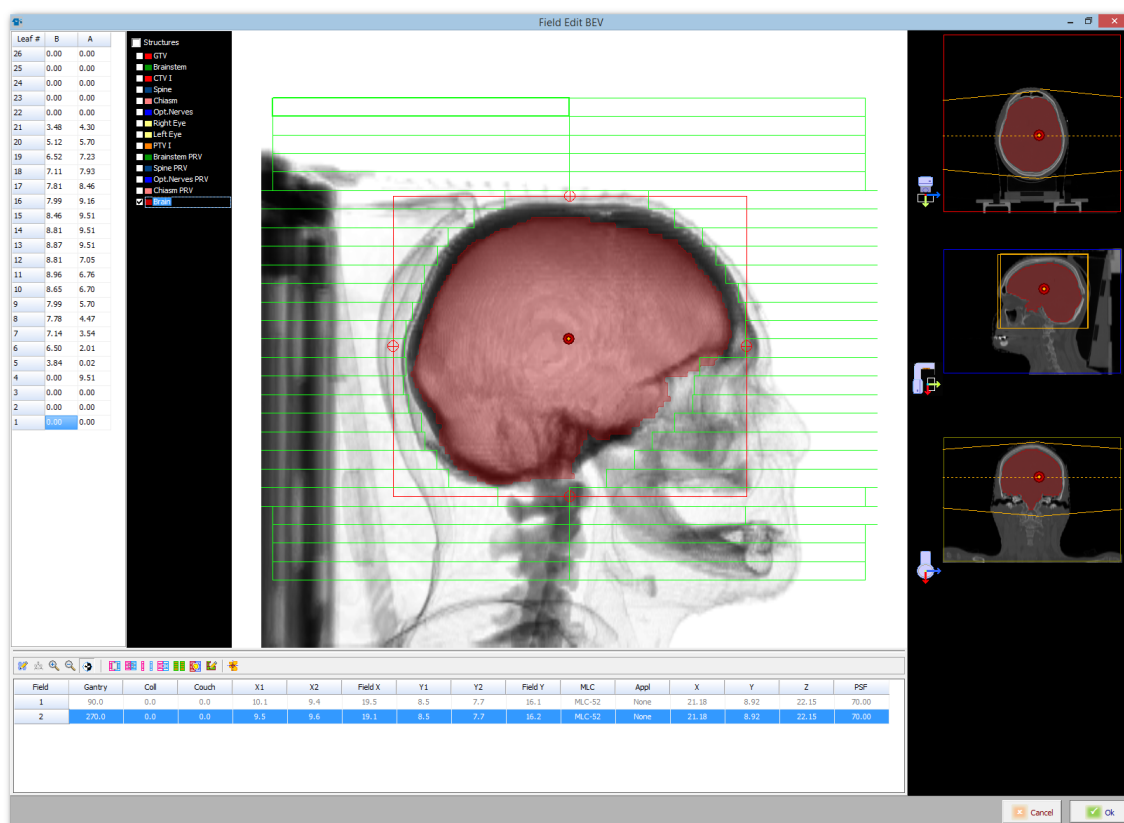


Figure 5.26: Parameters of the two parallel-opposed fields as seen in the BEV Edit dialog.

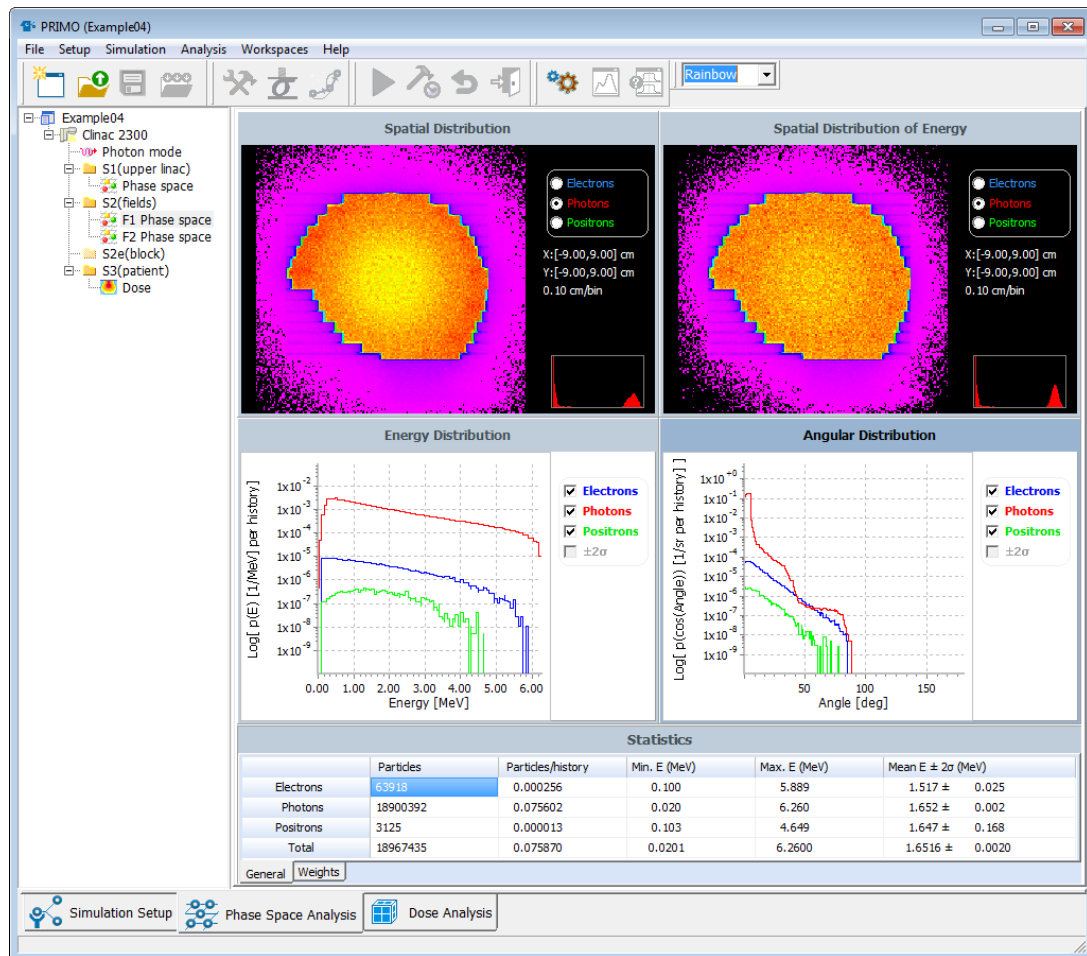


Figure 5.27: Analysis of the phase space created on the simulation of field 1.

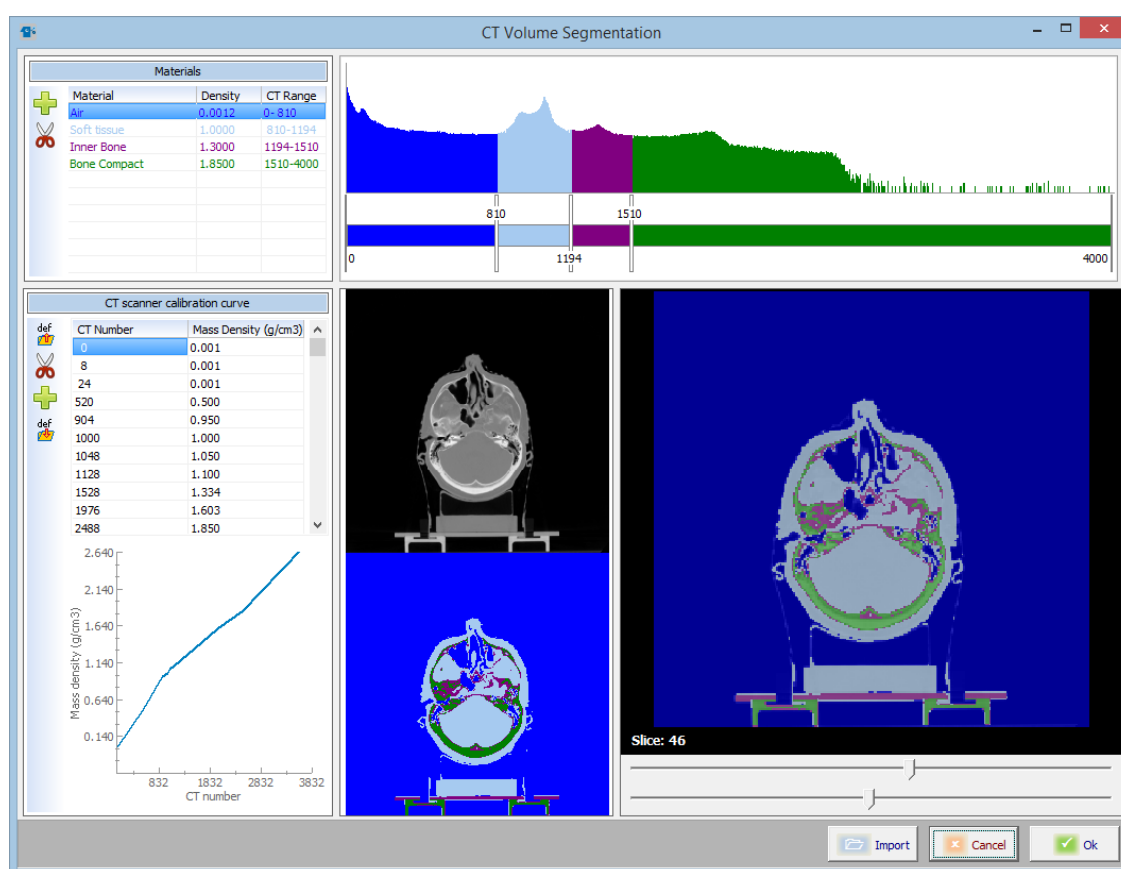


Figure 5.28: Construction of the voxelized geometry. Materials and densities used are shown.

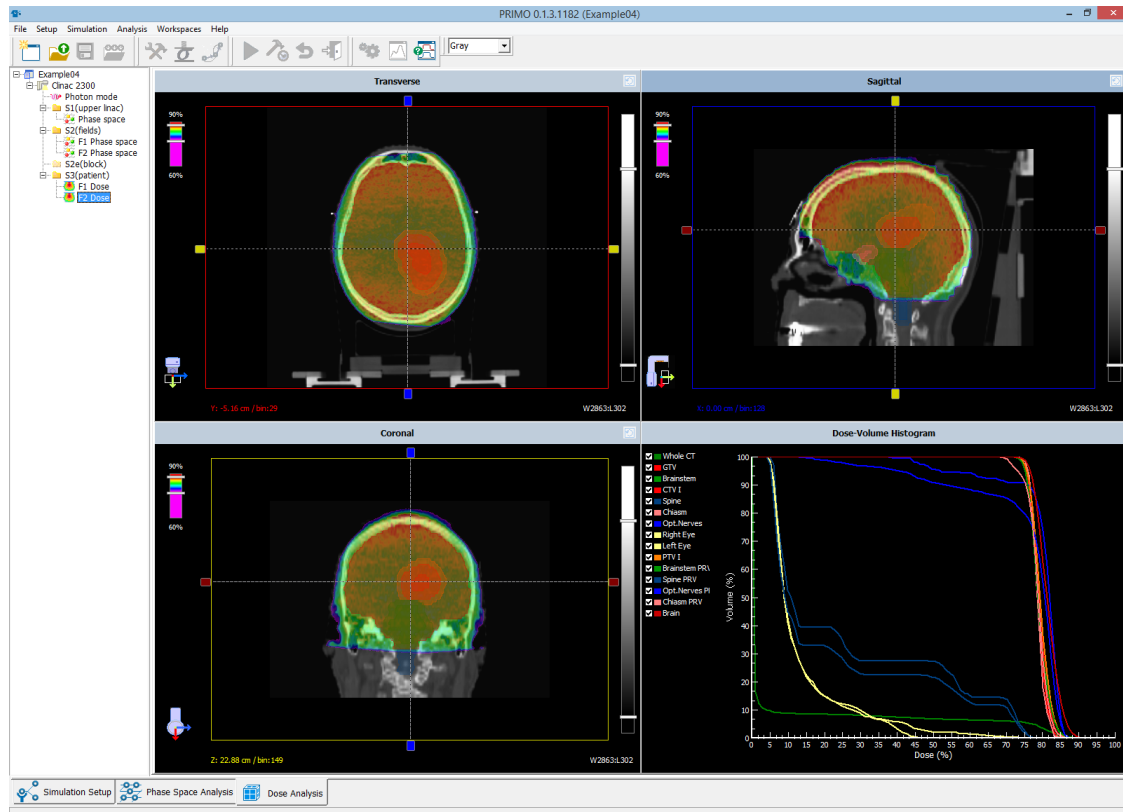


Figure 5.29: Dose distribution in the CT volume.

#### 5.4.4 Dose calculation

For the simulation of segment s3, splitting with a factor of 30 was used. The stopping condition of the simulation was determined by the relative uncertainty of the dose, which was set at 4% ( $2\sigma$ ). The dose distribution and the DVHs are shown in figure 5.29.

## Bibliography

### Books

- [Bru12] L Brualla. *Simulation of medical linear accelerators with PENELOPE*. Edited by G García Gómez-Tejedor and MC Fuss. Springer, 2012, pages 313–325 (cited on page 9).
- [Cap+06] R Capote et al. *Phase-Space Database for External Beam Radiotherapy*. Volume INDC (NDS)-0484. INDC International Nuclear Data Committee. Vienna: International Atomic Energy Agency, Jan. 2006 (cited on pages 9, 37).
- [JNR98] MT Jenkins, WR Nelson, and A Rindi, editors. *Monte Carlo Transport of Electrons and Photons*. City: Plenum Press, Jan. 1998 (cited on page 9).
- [SFS11] F Salvat, JM Fernández-Varea, and J Sempau. *PENELOPE 2011—A code system for Monte Carlo simulation of electron and photon transport*. Issy-les-Moulineaux: OECD Nuclear Energy Agency, 2011 (cited on pages 9, 34).
- [SV13] J Seco and F Verhaegen, editors. *Monte Carlo Techniques in Radiation Therapy*. 1st edition. Taylor & Francis, Mar. 2013 (cited on page 9).

### Articles

- [BS06] A Badal and J Sempau. “A package of Linux scripts for the parallelization of Monte Carlo simulations”. In: *Comput Phys Commun* 175 (2006), pages 440–450 (cited on page 34).
- [Bar+95] J Baró et al. “PENELOPE: an algorithm for Monte Carlo simulation of the penetration and energy loss of electrons and positrons in matter”. In: *Nucl Instrum Meth B* 100 (1995), pages 31–46 (cited on page 9).
- [BSP09] L Brualla, F Salvat, and R Palanco-Zamora. “Efficient Monte Carlo simulation of multileaf collimators using geometry-related variance-reduction techniques”. In: *Phys Med Biol* 54 (2009), pages 4131–4149 (cited on pages 10, 35).
- [BS10] L Brualla and W Sauerwein. “On the efficiency of azimuthal and rotational splitting for Monte Carlo simulation of clinical linear accelerators”. In: *Radiat Phys Chem* 79 (2010), pages 929–932 (cited on pages 10, 35).

- [Fad+08] B A Faddegon et al. “Benchmarking of Monte Carlo simulation of bremsstrahlung from thick targets at radiotherapy energies”. In: *Med Phys* 35 (2008), pages 4308–4317 (cited on page 10).
- [Fad+09] B A Faddegon et al. “The accuracy of EGSnrc, Geant4 and PENELOPE Monte Carlo systems for the simulation of electron scatter in external beam radiotherapy”. In: *Phys Med Biol* 54 (2009), pages 6151–6163 (cited on page 10).
- [Low+98] D Low et al. “A technique for the quantitative evaluation of dose distributions”. In: *Med Phys* 25 (1998), pages 656–661 (cited on page 51).
- [Rey+07] N Reynaert et al. “Monte Carlo treatment planning for photon and electron beams”. In: *Radiat Phys Chem* 76 (2007), pages 643–686 (cited on page 9).
- [RSB12] M Rodriguez, J Sempau, and L Brualla. “A combined approach of variance-reduction techniques for the efficient Monte Carlo simulation of linacs”. In: *Phys Med Biol* 57 (2012), pages 3013–3024 (cited on pages 10, 35).
- [SF09] F Salvat and J M Fernández-Varea. “Overview of physical interaction models for photon and electron transport used in Monte Carlo codes”. In: *Metrologia* 46 (2009), S112–S138 (cited on page 10).
- [SA06] J Sempau and P Andreo. “Configuration of the electron transport algorithm of PENELOPE to simulate ion chambers”. In: *Phys Med Biol* 51 (2006), pages 3533–3548 (cited on page 35).
- [SBB11] J Sempau, A Badal, and L Brualla. “A PENELOPE-based system for the automated Monte Carlo simulation of clinacs and voxelized geometries—application to far-from-axis fields”. In: *Med Phys* 38 (2011), pages 5887–5895 (cited on pages 9, 10, 18, 37).
- [Sem+97] J Sempau et al. “An algorithm for Monte Carlo simulation of coupled electron-photon transport”. In: *Nucl Instrum Meth B* 132 (1997), pages 377–390 (cited on page 9).
- [Sem+01] J Sempau et al. “Monte Carlo simulation of electrons beams from an accelerator head using PENELOPE”. In: *Phys Med Biol* 46 (2001), pages 1163–1186 (cited on page 17).
- [Sem+03] J Sempau et al. “Experimental benchmarks of the Monte Carlo code PENELOPE”. In: *Nucl Instrum Meth B* 207 (2003), pages 107–123 (cited on page 10).
- [SL08] E Spezi and G Lewis. “An overview of Monte Carlo treatment planning for radiotherapy”. In: *Radiat Prot Dosim* 131 (2008), pages 123–129 (cited on page 9).



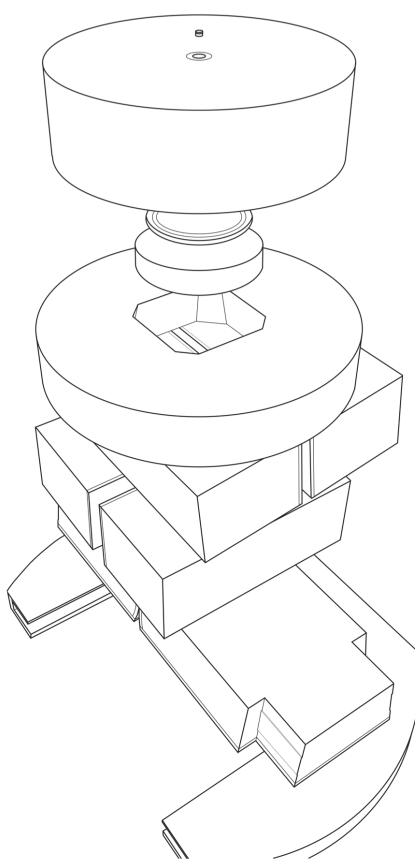
## Index

- Absolute dose, 17
- Absorbed dose simulation, 17
- Acknowledgments, 5
- Analysis of results, 41
- Analysis of the results
  - dose analysis, 43
    - comparing to experimental data, 46
    - comparison by gamma analysis, 51
    - inspecting the dose distribution, 46
  - phase-space file analysis, 41
- Anatomical structure, 25
- Beam parameters, 24
- Calibration curve, 31
- computerized tomography, 26
  - coordinate system, 28
  - import, 28
- Contouring, 32
- Copyright, 3
- CT numbers, 31
- CT scanner, 31
- DICOM, 28
- Disclaimer, 3
- Dose
  - analysis, 43
- Elekta
  - jaw positions, 73
- Example 01, 53
  - analysis of results, 59
  - comparison with another simulation, 68
  - comparison with experiment, 62
  - simulation setup and execution, 53
- Example 02
  - analysis of results, 74
  - electron reference field, 70
  - simulation setup and execution, 70
- Example 03, 74
  - TrueBeam, 74
- Example 04
  - construction of the voxelized geometry, 77
  - dose calculation, 82
  - dose calculation in a computerized tomography, 77
  - field setup, 77
  - simulation of segment s1, 77
- Execution, 38
  - changing parameters at runtime, 40
- Field conformation, 24
- Field edit, 25
- Gamma analysis, 51
- Genesis, 10
- GLASS, 9
- Histories and particles, 17
- Hounsfield numbers, 31
- IAEA format, 9
- Import structures, 32
- Installation, 13, 14
  - examples, 14
  - requirements, 13
- Introduction, 9

- isocenter, 25
- Linac models, 22
- Linac simulation, 15
- List of citable references, 11
- List of examples, 14
- Mass density, 31
- Material and density assignment, 30
- penEasy, 9, 40
- penEasyLinac, 9, 40
- PENELOPE, 9
- Phase-space file, 17
  - analysis, 41
  - import, 75
- Photon reference field, 53
- Reset button, 39
- Scope, 10
- Segments
  - separate simulation, 27
- Self-Installation, 14
- Simulation segments, 18
- Simulation setup, 21
  - beam parameters (s1), 24
  - dose tallying (s3), 26
  - field conformation (s2), 24
  - field Edit BEV, 25
  - material and density assignment, 30
  - multiple fields, 24
  - new project, 21
  - segments, 24
  - tab, 22
- SSD, 73
- Statistical uncertainty, 17
- Uninstall, 14
- Varian Clinac 2100 C/D, 53
- Variance-reduction techniques, 9, 10, 18, 37
  - rotational splitting, 54
  - splitting roulette, 54
- Version of PRIMO, 11
- Water phantom, 26
  - coordinate system, 27
- Webpage and resources, 10
- Whence to obtain PRIMO, 13



# PRIMO Quick Start Guide



Brualla · Rodriguez · Sempau

Copyright © 2010–2014 L. Brualla, M. Rodriguez, J. Sempau

PRIMO QUICK START GUIDE  
SOFTWARE VERSION 0.1.5.1202

WWW.PRIMOPROJECT.NET

About the authors:

Lorenzo Brualla  
*Universität Duisburg-Essen*  
*Universitätsklinikum Essen*  
*NCTeam, Strahlenklinik*  
*Hufelandstraße 55, D-45122 Essen, Germany*

Miguel Rodriguez and Josep Sempau  
*Universitat Politècnica de Catalunya*  
*Institut de Tècniques Energètiques*  
*Diagonal 647, E-08028 Barcelona, Spain*

lorenzo.brualla@uni-duisburg-essen.de  
miguel.lazaro.rodriguez@upc.es  
josep.sempau@upc.es

Typeset by the authors with the L<sup>A</sup>T<sub>E</sub>X Documentation System. The *Legrand Orange Book* template downloaded from <http://www.LaTeXTemplates.com> was used to write this guide.

29<sup>th</sup> December 2014

## Disclaimer and Copyright

PRIMO is furnished 'as is'. No warranties, express or implied, that this software is free of error, or is consistent with any particular standard of merchantability, or that it will meet your requirements for any particular application, is made. No responsibility for any mathematical or technical limitations of the procedures and functions which make up this software is accepted. This software should not be relied on for solving a problem whose incorrect solution could result in injury to a person or loss of property. The authors of PRIMO shall not in any event be liable for any damages, whether direct or indirect, special or general, consequential or incidental, arising from use of this software. Your use of this software is entirely at your own risk.

This software is sole property of its authors. Permission to use this software for any purpose is hereby granted without fee. This software, or any part of it, cannot be sold, modified or re-distributed unless a written consent from the authors is obtained. Reverse-engineering is forbidden on any of the distributed files and on the files generated during the execution of the code.



## Important information

### Purpose of this guide

The scope of this *Quick Start Guide* is to give users the possibility to quickly get a glimpse of the basic features of PRIMO. However, this guide is not intended as a substitute for the *PRIMO User's Manual*, whose reading is necessary in order to use PRIMO adequately.

### What is PRIMO?

PRIMO is an automated, self-contained, fully Monte Carlo-based linac simulator and dose calculator. It features a user-friendly graphical interface. The Monte Carlo engine of PRIMO is PENELOPE 2011. PRIMO can estimate the absorbed dose in water phantoms and computerized tomography studies written in DICOM format. It is possible to delineate structures using the tools provided by the software or, if desired, structures in DICOM-RT STRUCT format can be imported.

### Available linacs

The linacs that can be simulated are listed in table 1. PRIMO can import phase-space files written in the International Atomic Energy Agency (IAEA) format. Phase-space files written by PRIMO also comply with the IAEA standard.

### Legal matters in a nutshell

PRIMO is free software. However, it is not open-source and reverse-engineering of any distributed or generated file from the software is expressly forbidden.

Although PRIMO is mainly conceived as research software, it finds multiple applications in the routine clinical practice. For example, it can be used as an independent quality assurance tool. However, it must be stressed that PRIMO is not medical software and it does not have any certification or warranty.

<sup>1</sup>For simulating this linac it is necessary to obtain the phase-space files distributed by Varian. These phase-space files are currently only available to TrueBeam owners.

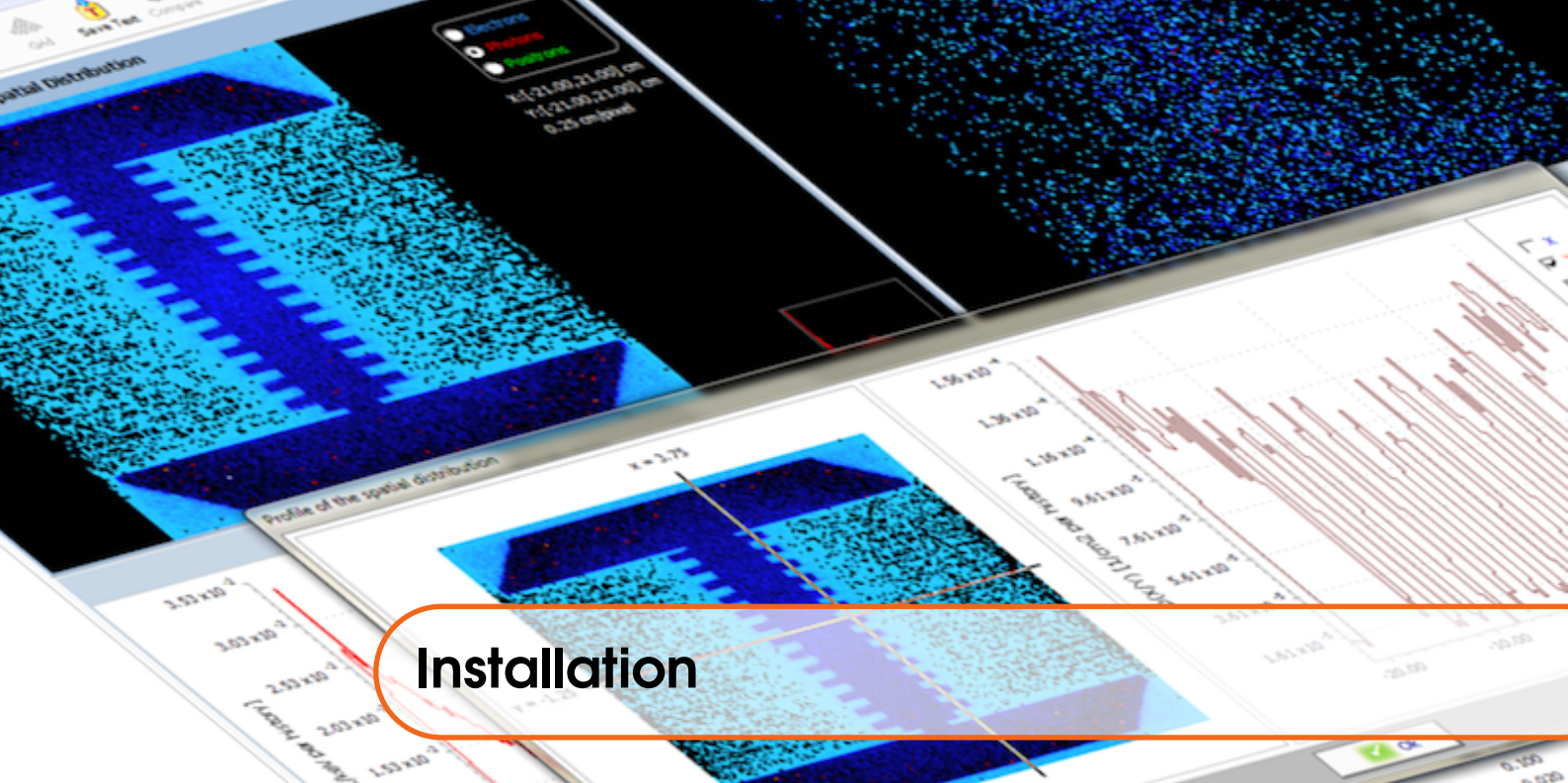
<sup>2</sup>Experimentally based geometry of TrueBeam developed in-house

PRIMO	commercial
Elekta SL	SL series
Elekta MLCi	SLi Plus, Axesse, Affinity, Synergy, Precise
Varian Clinac 600C	Clinac 600 C
Varian Clinac 600CD	Clinac 600 C/D
Varian Unique	Unique
Varian Clinac 2100	Clinac C series, TrueBeam <sup>1</sup>
Varian Clinac 2300	Clinac 2300 C/D
FakeBeam	TrueBeam 6- and 10-FFF beams <sup>2</sup>

Table 1: The column ‘PRIMO’ indicates the name given to the available linacs in the program. The column ‘commercial’ indicates the different commercial names that the same linac in PRIMO might have.

PRIMO can be downloaded from the project webpage <http://www.primoproject.net>. It is forbidden to redistribute copies of this software.

Please refer to the disclaimer and copyright statements for further details.



## Installation

### Tested hardware and software configurations

We have tested and successfully run the code in the following hardware and software configurations:

- Computer with Intel 64 bits processor
- Windows 64 bits<sup>3</sup> operating system<sup>4</sup>
- Administrator rights
- Recommended 1 GB RAM per computing core. For example, a computer with 2 CPUs each with 4 computing cores requires about 8 GB RAM.<sup>5</sup>
- The hard drive must be either local or accessible through a high speed connection (at least 6 GB/s).
- PRIMO occupies less than 100 MB of disk space. Owing to the fact that large files might be generated during execution a minimum of 100 GB of free disk space is recommended for a normal usage of the software.
- Minimum screen resolution 1280 × 960 pixels
- The default font size in Windows (smaller size) must be used.

### Whence to obtain PRIMO

1. Visit the webpage <http://www.primoproject.net>.
2. Enter the section 'Download' and enter your name, email address and affiliation. You will receive an email with a link for downloading the software. PRIMO is distributed as a Windows installer msi file.
3. With the same link sent for downloading PRIMO it is possible to download already simulated examples. Each example is distributed as a compressed zip file.

<sup>3</sup>PRIMO has been successfully tested on Windows XP, Windows 7, Windows 8, Windows Server 2008 and Windows Server 2012. The classical view of the Windows desktop does not permit to see the full graphical quality of the program's icons.

<sup>4</sup>PRIMO may give problems on virtual machines (Parallels, VirtualBox, VMWare, etc.) and it does not run on Windows emulators (Wine, CrossOver, etc.)

<sup>5</sup>This 1 GB RAM rule also applies to logical computing cores in case of using hyper-threading. However, use of hyper-threading is not recommended.

## How to install/uninstall PRIMO

### Installer msi file

1. Execute the msi file.
2. The installation program will guide you through the installation process. The default installation folder is `c:\PRIMO` but it can be changed during the installation process.

**R** Errors during simulation execution could occur when PRIMO is installed in the Program Files folder of a server computer running Windows Server 2012. In that case it is advisable to select a different installation folder, such as `c:\PRIMO`.

### Uninstall

Use the Windows Start Menu → Control Panel → Programs and Features to uninstall any version of PRIMO that was installed from an msi file.

**R** Uninstall any previous version before installing a new one.

### How to install the examples

1. Create a directory called `PRIMOexamples` under `c:\`.
2. Unpack the downloaded files inside the just created directory `c:\PRIMOexamples`. The unpacked files will each one create a directory called `Example $mn$` , with  $mn$  a two digits number.

### List of examples

**Example01** Photon reference field from a Varian Clinac 2100 C/D. Nominal energy 6 MV. Field size  $10 \times 10 \text{ cm}^2$ . Tallied results: phase-space files and dose distribution.

**Example02** Electron reference field from an Elekta SL15. Nominal energy 15 MeV. Electron applicator  $20 \times 20 \text{ cm}^2$ . Tallied results: phase-space file and dose distribution.

**Example03** Photon reference field from a TrueBeam STx. Nominal energy 6 MV (free flattening filter). Field size  $40 \times 40 \text{ cm}^2$ . Tallied result: dose distribution.

**Example04** Brain irradiation with two fields from a Varian Clinac 2100 C/D conformed with a MLC 120 High Definition. Nominal energy 6 MV. Tallied results: phase-space files and dose distribution.

Please notice the following remark:

**R** Examples 01, 02 and 03 contain subdirectories named `Experiment` where experimental data files are stored for comparison with simulated results.





## Running PRIMO

PRIMO has been designed with an intuitive graphical user interface. It is assumed that the user has the basic operating knowledge of a medical physicist experienced with treatment planning systems. For the correct interpretation of the simulated results some basic knowledge of the Monte Carlo method is useful. Although the program is flexible enough to satisfy the requirements of advanced users, it always provides reasonable default parameters and hints.

**R** To launch PRIMO double-click on the PRIMO.exe file icon.

### Using the examples

Before launching your own simulation it is recommended to load the examples and touring through the screens of PRIMO. Each of the distributed examples exploits different features of the software. The following instructions are valid for any of the examples.














1. Launch PRIMO
2. Go to File → Open
3. Open the project file (\*.ppj) stored inside the chosen example directory. Immediately after opening a project PRIMO shows its main screen with the Simulation Setup tab selected.
4. All examples but number 03 contain phase-space files. If a phase-space file is available, this can be analyzed by clicking the Phase Space Analysis tab, then dragging the desired phase-space file icon on the left logical tree and dropping it in the blank screen area. Alternatively, right click on the icon and select the option *Analyze* in the appearing pop-up menu.
5. Similarly to phase-space files, dose distributions can be analyzed by clicking the Dose Analysis tab and then dragging the desired dose icon on the left logical tree and dropping it in the blank screen area. Alternatively, right click on the dose icon and select the option *Analyze* in the appearing pop-up menu.
6. Simulation properties of both phase-space files and dose distributions can be viewed by right-clicking the corresponding icons on the logical tree.

**R** Most of the screens showing plots give further options if the right mouse button is clicked over the plot.

It is recommended to navigate through all the screens of the examples and to click all buttons to familiarize with the available features. In case of inadvertently damaging any example, it can be restored by erasing its directory and installing it again.

## Simulating a linac in photon mode

This section covers how to simulate a reference field of a Varian Clinac 2100 C/D in photon mode. It is a summary of the steps presented in section 5.1 of the *PRIMO User's Manual*.

1. Launch PRIMO
2. Go to File → New or click the  button.
3. In the New Project window enter a name for your simulation project. Then from the drop-down menu choose Clinac 2100 and click the radio button Photon to choose photon mode.
4. Click the s2 tab from the Segment Setup panel in the main PRIMO screen. Click the  button. In this window the field size can be defined. For the purpose of this example it will be left as  $10 \times 10 \text{ cm}^2$ . Click *Ok*.
5. Click the s3 tab from the Segment Setup panel in the main PRIMO screen. Click the  button. Enter in the Source to phantom surface distance (SSD) field the value of 95 cm. Click *Ok*.
6. Click the  button. Select the Splitting-Roulette radio button. Click *Ok*.
7. Click the  button. Check the Time checkbox. Uncheck the Histories checkbox. Enter the value 1800 seconds in the Time field for a 30 minutes simulation. In the Number of parallel processes field enter the maximum number of computing cores available in your computer. Click *Ok*.
8. Click the  button. Click the  button.
9. Once the simulation has finalized the open door button in the execution window will become active. Click it and the main PRIMO window will resume. Then check the Active checkbox next to s2 in order to simulate the second segment of the linac (jaws).
10. Click the  and  buttons in order to simulate the second segment.
11. Once the simulation has finalized click the  button in the execution window will become active. The main PRIMO window will resume. Then check the Active checkbox next to s3 in order to simulate the third segment of the linac (dose distribution).
12. Click the  and  buttons in order to simulate the third segment.
13. Once the simulation has finalized click the  button. The main PRIMO window will resume.

At this point a full simulation of the linac and the dose distribution obtained in water is completed. To analyze the obtained results proceed as described previously for the examples.



## REFERENCES

---

- [1] S. Agostinelli, J. Allison, K. Amako, J. Apostolakis, H. Araujo, P. Arce, M. Asai, D. Axen, S. Banerjee, G. Barrant, F. Behner, L. Bellagamba, J. Boudreau, L. Broglia, A. Brunengo, H. Burkhardt, S. Chauvie, J. Chuma, R. Chytracsek, G. Cooperman, G. Cosmo, P. Degtyarenko, A. Dell'Acqua, I. G. Depaola, D. Dietrich, R. Enami, A. Feliciello, C. Ferguson, H. Fesefeldt, G. Folger, F. Foppiano, A. Forti, S. Garelli, S. Giani, R. Giannitrapani, D. Gibin, and Gómez Cadenas J. Geant4—a simulation toolkit. *Nucl. Instr. Meth. Phys. Res. A*, 506:250–303, 2003.
- [2] K. Aljarrah, G. C. Sharp, T. Neicu, and S. B. Jiang. Determination of the initial beam parameters in Monte Carlo linac simulation. *Med. Phys.*, 33:850–858, 2006.
- [3] S. S. Almberg, J. Frengen, A. Kylling, and T. Lindmo. Monte Carlo linear accelerator simulation of megavoltage photon beams: Independent determination of initial beam parameters. *Med. Phys.*, 29:40–47, 2002.
- [4] P. Andreo. Dose to 'water-like media' or dose to tissue in MV photons radiotherapy treatment planning: still a matter of debate. *Phys. Med. Biol.*, 60:309–337, 2015.
- [5] A. Anhesjö. Collapsed cone convolution of radiant energy for photon dose calculation in heterogeneous media. *Med. Phys.*, 16:577–592, 1989.
- [6] A. Anhesjö and M. M. Aspradakis. Dose calculations for external photon beams in radiotherapy. *Phys. Med. Biol.*, 44:R99–R115, 1999.
- [7] M. R. Arnfield, C. H. Siantar, J. Siebers, P. Garmon, L. Cox, and R. Mohan. The impact of electron transport on the accuracy of computed dose. *Med. Phys.*, 27:1266–1274, 2000.
- [8] M. Åsell, S. Hyödynmaa, A. Gustafsson, and A. Brahme. Optimization of 3d conformal electron beam therapy in inhomogeneous media by concomitant fluence and energy modulation. *Phys. Med. Biol.*, 42:2083–2100, 1997.
- [9] H. F. Batho. Lung corrections in cobalt 60 beam therapy. *J. of the Canadian Association of Radiologists*, 15:39–83, 1964.
- [10] G. Battistoni, F. Cerutti, A. Fasso, A. Ferrari, S. Muraro, J. Ranft, S. Roesle, and P. R. Sala. The FLUKA code: description and

- benchmarking, year=2007, volume=896, pages=31-49,. *AIP Conference Proceeding*.
- [11] H. Benmakhlouf, J. Johansson, I. Paddick, and P. Andreo. Monte Carlo calculated and experimentally determined output correction factors for small field detectors in Leksell Gamma Knife Perfexion beams. *Phys. Med. Biol.*, 60:3959-3973, 2015.
- [12] J. M. Berger. Monte Carlo calculation of the penetration and diffusion of fast charged particles. In B. Alder, S. Fernbach, and M. Rotenberg, editors, *Methods in Computational Physics*, volume 1, pages 135-215. Academic Press, New York, USA, 1963.
- [13] A. M. Bergman, K. Otto, and C. Duzenli. The use of modified single pencil beam dose kernels to improve IMRT dose calculation accuracy. *Med. Phys.*, 31:3279-3287, 2004.
- [14] T. E. Booth. *A Sample Problem for Variance Reduction in MCNP*. LA-10363-MS, Los Alamos National Laboratory, Los Alamos, USA, 1985.
- [15] T. Bortfeld, J. Burkelbach, R. Boesecke, and W. Schlegel. Methods of image reconstruction from projections applied to conformation radiotherapy. *Phys. Med. Biol.*, 35:1423-1434, 1990.
- [16] A. Brahme. Optimization of stationary and moving beam radiation therapy techniques. *Radiother. Oncol.*, 12:129-140, 1988.
- [17] A. Brahme, I. Lax, and P. Andreo. Electron beam dose planning using discrete gaussian beams. mathematical background. *Acta Radiol. Oncol.*, 20:147-158, 1981.
- [18] J. E. Bresenham. Algorithm for computer control of a digital plotter. *IBM Systems Journal*, 4:25-30, 1965.
- [19] L. Brualla, F. Salvat, and R. Palanco-Zamora. Efficient Monte Carlo simulation of multileaf collimators using geometry-related variance-reduction techniques. *Phys. Med. Biol.*, 54:4131-4149, 2009.
- [20] G. Bueno, O. Déniz, C. B. Carrascosa, J. M. Delgado, and L. Brualla. Fast Monte Carlo simulation on a voxelized human phantom deformed to a patient. *Med. Phys.*, 36:5162-5174, 2009.
- [21] F. Buffa and A. E. Nahum. Monte Carlo calculations and radiobiological modelling: analysis of the effect on the statistical noise of the dose distribution on the probability of tumor control. *Phys. Med. Biol.*, 45:3009-3023, 2000.
- [22] D. T. Burns. A new approach to the determination of air kerma using primary-standard cavity ionization chambers. *Phys. Med. Biol.*, 51:929-942, 2005.

- [23] K. Bush, S. F. Zavgorodni, and W. A. Beckham. Azimuthal particle redistribution for the reduction of latent phase-space variance in Monte Carlo simulations. *Phys. Med. Biol.*, 52:4345–4360, 2007.
- [24] R. Capote, R. Jeraj, C-M. Ma, D. W. O. Rogers, F. Sánchez-Doblado, J. Sempau, J. Seuntjens, and J. V. Siebers. *Phase-space database for external beam radiotherapy*. International Atomic Energy Agency, Nuclear Data Section, Report INDC(NDS)-0484, Vienna, Austria, 2006.
- [25] Memorial Sloan-Kettering Cancer Center. *A practical guide to Intensity-Modulated Radiation Therapy*. Medical Physics Publishing, 2003.
- [26] I. Chetty, J. J. DeMarco, and T. D. Solberg. A virtual source model for Monte Carlo modeling of arbitrary intensity distributions. *Med. Phys.*, 27:166–172, 2000.
- [27] I. Chetty, J. J. DeMarco, T. D. Solberg, A. R. Arellano, R. Fogg, and A. V. Mesa. A phase space model for simulating arbitrary intensity distributions for shaped radiosurgery beams using the Monte Carlo method. *Radiosurgery*, 3:41–52, 2000.
- [28] O. Chibani and C-M. Ma. On the discrepancies between Monte Carlo dose calculations and measurements for the 18 MV Varian photon beam. *Med. Phys.*, 34:1206–1216, 2007.
- [29] J. R. Clarkson. A note on depth doses in fields of irregular shape. *Br. J. Radiol.*, 14:265–268, 1941.
- [30] E. Conneely, A. Alexander, G. Stroian, J. Seuntjens, and M. J. Foley. An investigation into the use of MMCTP to tune accelerator source parameters and testing its clinical application. *J. Appl. Clin. Med. Phys.*, 14:3–14, 2013.
- [31] D. J. Convery and M. E. Rosenbloom. The generation of intensity-modulated fields for conformal radiotherapy by dynamic collimation. *Phys. Med. Biol.*, 37:1359–1374, 1992.
- [32] B. De Smedt, N. Reynaert, F. Flachet, M. Coghe, M. G. Thompson, and L. Paelinck. Decoupling initial electron beam parameters for Monte Carlo photon beam modelling by removing beam-modifying filters from the beam path. *Phys. Med. Biol.*, 50:5935–5951, 2005.
- [33] B. De Smedt, B. Vanderstraeten, N. Reynaert, W. De Neve, and H. Thierens. Investigation of geometrical and scoring grid resolution for Monte Carlo dose calculations for IMRT. *Phys. Med. Biol.*, 50:4005–4019, 2005.



- [34] J. J. DeMarco, T. D. Solberg, and J. B. Smathers. A CT-based Monte Carlo simulation tool for dosimetry planning and analysis. *Med. Phys.*, 25:1–11, 1998.
- [35] J. Deng, S. B. Jiang, T. Pawlicki, J. Li, and C-M. Ma. Derivation of electron and photon energy spectra from electron beam central axis depth dose curves. *Phys. Med. Biol.*, 46:1429–1449, 2001.
- [36] N. Dogan, J. V. Siebers, and P. J. Keall. Clinical comparison of head and neck and prostate IMRT plans using absorbed dose to medium and absorbed dose to water. *Phys. Med. Biol.*, 51:4967–4980, 2006.
- [37] F. C. du Plessis, A. Willemse, M. G. Lotter, and L. Goedhals. The indirect use of CT numbers to establish material properties needed for Monte Carlo calculation of dose distributions in patients. *Med. Phys.*, 25:1195–1201, 1998.
- [38] B. A. Faddegon and I. Blevis. Electron spectra derived from depth dose distributions. *Med. Phys.*, 27:514–526, 2000.
- [39] B. A. Faddegon, I. Kawrakow, Y. Kubyshev, J. Perl, J. Sempau, and L. Urban. The accuracy of EGSnrc, Geant4 and PENELOPE Monte Carlo systems for the simulation of electron scatter in external beam radiotherapy. *Phys. Med. Biol.*, 54:6151–6163, 2009.
- [40] J. M. Fernández-Varea, R. Mayol, J. Baró, and F. Salvat. On the theory and simulation of multiple elastic scattering of electrons. *Nucl. Instrum. Meth. B*, 73:447–473, 1993.
- [41] J. M. Fernández-Varea, P. Carrasco, V. Panettieri, and L. Brualla. Monte Carlo based water/medium stopping-power ratios for various ICRP and ICRU tissues. *Phys. Med. Biol.*, 52:6475–6483, 2007.
- [42] M. Fippel. Fast Monte Carlo dose calculation for photon beams based on the VMC electron algorithm. *Med. Phys.*, 26:1466–1475, 1999.
- [43] M. Fippel, F. Haryanto, O. Dohm, F. Nüsslin, and S. Kriesen. A virtual photon energy fluence model for Monte Carlo dose calculation. *Med. Phys.*, 30:301–311, 2003.
- [44] A. Fogliata, G. Nicolini, A. Clivio, E. Vanetti, and L. Cozzi. Dosimetric evaluation of Acuros XB Advanced Dose Calculation algorithm in heterogeneous media. *Rad. Oncol.*, 6:1–15, 2011.
- [45] S. García-Pareja, M. Vilches, and A. M. Lallena. Ant colony method to control variance-reduction techniques in the Monte Carlo simulation of clinical electron linear accelerators. *Nucl. Instr. Meth. Phys. Res. A*, 580:510–513, 2007.

- [46] T. Goorley, T. John, M. R. James, T. E. Booth, F. B. Brown, J. S. Bull, L. J. Cox, J. W. Jr. Durkee, J. S. Elson, M. L. Fensin, R. A. III Forster, J. S. Hendricks, H. G. III Hughes, R. C. Johns, B. C. Kiedrowski, R. L. Martz, S. G. Mashnik, G. W. McKinney, D. B. Pelowitz, R. E. Prael, J. E. Sweezy, L. S. Waters, T. Wilcox, and A. J. Zukaitis. Initial MCNP6 release overview – MCNP6 Version 1.0. In *LA-UR-13-22934*. Los Alamos National Laboratory, Los Alamos, USA, 2013.
- [47] S. Goudsmit and J. L. Saunderson. Multiple scattering of electrons. *Phys. Rev.*, 57:24–29, 1940.
- [48] S. Goudsmit and J. L. Saunderson. Multiple scattering of electrons. II. *Phys. Rev.*, 58:36–42, 1940.
- [49] B. Habib, B. Poumarede, F. Tola, and J. Barthe. Evaluation of PENFAST – A fast Monte Carlo code for dose calculations in photon and electron radiotherapy treatment planning. *Physica Medica: Eur. J. Med. Phys.*, 26:17–25, 2010.
- [50] O. K. Harling. Fission reactor based epithermal neutron irradiation facilities for routine clinical application in BNCT. *Applied Radiation and Isotopes*, 67:S7–S11, 2009.
- [51] G. T. Herman, J. Zheng, and C. A. Bucholtz. Shape-based interpolation. *IEEE; Computer Graphics and Applications*, 12:69–79, 1992.
- [52] S. Hyödynmaa. Electron beam dose computation using generalized Gaussian pencil beam algorithm with 3-D inhomogeneity correction and arbitrary field shapes. In *Proceedings of the Xth International Conference on the Use of Computers in Radiation Therapy*. Manchester, UK, 1994.
- [53] ICRP. Report of the Task Group on Reference Man. In *Annals of the International Commission on Radiological Protection*. Publication 23, 1975.
- [54] ICRU. Photon, electron, proton and neutron interaction data for body tissues. In *International Commission on Radiation Units and Measurements*. Report No. 46, .
- [55] ICRU. Prescribing, Recording, and Reporting Photon Beam Therapy. In *International Commission on Radiation Units and Measurements*. Report No. 50, .
- [56] ICRU. Prescribing, Recording and Reporting Photon Beam Therapy (Supplement to ICRU Report 50). In *International Commission on Radiation Units and Measurements*. Report No. 62, .



- [57] ICRU. Prescribing, Recording, and Reporting Intensity-Modulated Photon-Beam Therapy (IMRT). In *International Commission on Radiation Units and Measurements*. Report No. 83, .
- [58] O. Jäkel, M. Krämer, C. P. Karger, and J. Debus. Treatment planning for heavy ion radiotherapy: clinical implementation and application. *Phys. Med Biol.*, 46:1101–1116, 2001.
- [59] O. Jäkel, D. Schulz-Ertner, C. P. Karger, A. Nikoghosyan, and J. Debus. Heavy ion therapy: status and perspectives. *Technology in Cancer Research and Treatment*, 2:377–388, 2003.
- [60] T. M. Jenkins, W. R. Nelson, and A. Rindi, editors. *Monte Carlo Transport of Electrons and Photons*. Plenum Press, New York, USA, 1988.
- [61] S. B. Jiang, A. Kapur, and C-M. Ma. Electron beam modeling and commissioning for Monte Carlo treatment planning. *Med. Phys.*, 27:180–191, 2000.
- [62] M. H. Kalos and P. A. Whitlock. *Monte Carlo methods*. Wiley-Blackwell Verlag GmbH & Co. KGaA, Weinheim, Germany, 2008.
- [63] C. J. Karzmark, A. Deubert, and R. Loevinger. Tissue-phantom ratios—and aid to treatment planning. *Br. J. Radiol.*, 38:58, 1965.
- [64] I. Kawrakow. Accurate condensed history Monte Carlo simulation of electron transport. I. EGSnrc, the new EGS4 version. *Med. Phys.*, 27:485–498, 2000.
- [65] I. Kawrakow and M. Fippel. VMC++, a MC algorithm optimized for electron and photon beam dose calculations for RTP. In *Proceedings of the 22nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Vol. 2, Chicago, 2000*.
- [66] I. Kawrakow and M. Fippel. VMC++, a fast MC algorithm for radiation treatment planning. In W. Schlegel and T. Bortfeld, editors, *The Use of Computers in Radiation Therapy, XIIIth International Conference*, pages 126–128. Springer-Verlag, Heidelberg, Germany, 2000.
- [67] I. Kawrakow and M. Fippel. Investigation of variance reduction techniques for Monte Carlo photon dose calculation using XVMC. *Phys. Med. Biol.*, 45:2163–2183, 2000.
- [68] I. Kawrakow and B. R. Walters. Efficient photon beam dose calculations using DOSXYZnrc with BEAMnrc. *Med. Phys.*, 33:3046–3056, 2006.

- [69] I. Kawrakow, M. Fippel, and K. Friedrich. 3D electron dose calculation using Voxel based Monte Carlo algorithm (VMC). *Med. Phys.*, 23:445–447, 1996.
- [70] I. Kawrakow, D. W. O. Rogers, and B. R. B. Walters. Large efficiency improvements in BEAMnrc using directional bremsstrahlung splitting. *Med. Phys.*, 31:2883–2898, 2004.
- [71] P. J. Keall, J. V. Siebers, R. Jeraj, and R. Mohan. The effect of dose calculation uncertainty on the evaluation of radiotherapy plans. *Med. Phys.*, 27:478–484, 2000.
- [72] T. Krieger and O. Sauer. Monte Carlo versus pencil beam/-collapsed cone dose calculation in a heterogeneous multi-layer phantom. *Phys. Med. Biol.*, 50:859–868, 2005.
- [73] L. Landau. On the energy loss of fast particles by ionization. *J. Phys. U.S.S.R.*, 8:201–205, 1944.
- [74] I Lax. *Development of a generalized Gaussian model for absorbed dose calculation and dose planning in therapeutic electron beams*. Ph.D. Thesis. Stockholm University, Stockholm, Sweden, 1986.
- [75] H. W. Lewis. Multiple scattering in an infinite medium. *Phys. Rev.*, 78:526–529, 1950.
- [76] H. H. Liu and P. Keall.  $D_m$  rather than  $D_w$  should be used in Monte Carlo treatment planning. *Med. Phys.*, 29:922–924, 2002.
- [77] D. Low, W. Harms, S. Mutic, and J. Purdy. A technique for the quantitative evaluation of dose distributions. *Med. Phys.*, 25:656–661, 1998.
- [78] C-M. Ma, E. Mok, A. Kapur, T. Pawlicki, D. Findley, S. Brain, K. Forster, and A. L. Boyer. Clinical implementation of a Monte Carlo treatment planning system. *Med. Phys.*, 26:2133–2143, 1999.
- [79] C-M. Ma, J. S. Li, S. B. Jiang, T. Pawlicki, W. Xiong, L. H. Qin, and J. Yang. Effect of statistical uncertainties on Monte Carlo treatment planning. *Phys. Med. Biol.*, 50:891–907, 2005.
- [80] T. R. Mackie, A. F. Bielajew, D. W. Rogers, and J. J. Battista. Generation of photon energy deposition kernels using the EGS Monte Carlo code. *Phys. Med. Biol.*, 33:1–20, 1988.
- [81] T. R. Mackie, T. Holmes, S. Swerdloff, P. Reckwerdt, J. O. Deasy, J. Yang, B. Paliwal, and T. Kinsella. Tomotherapy: A new concept for the delivery of dynamic conformal radiotherapy. *Med. Phys.*, 20:1709–1719, 1993.

- [82] N. L. Maidana, V. R. Vanin, V. Jahnke, J. M. Fernández-Varea, M. N. Martins, and L. Brualla. Efficiency calibration of x-ray HPGe detectors for photons with energies above Ge K binding energy. *Nucl. Instrum. Meth. A*, 729:371–380, 2013.
- [83] C. Martens, N. Reynaert, C. De Wagter, P. Nilsson, M. Coghe, H. Palmans, H. Thierens, and W. De Neve. Underdosage of the upper-airway mucosa for small fields as used in intensity-modulated radiation therapy: a comparison between radiochromic film measurements, Monte Carlo simulations, and collapsed cone convolution calculations. *Med. Phys.*, 29:1528–1535, 2002.
- [84] R. Mohan, C. Chui, and L. Lidofsky. Energy and angular distributions of photons from medical linear accelerators. *Med. Phys.*, 12:592–597, 1985.
- [85] R. Mohan, C. Chui, and L. Lidofsky. Differential pencil beam dose computation model for photons. *Med. Phys.*, 13:64–73, 1986.
- [86] J. Munzenrider, M. Pilepich, and J. Rene-Ferrero. Use of a body scanner in radiotherapy treatment planning. *Cancer*, 40:170–179, 1977.
- [87] W R Nelson, H Hirayama, and D W O Rogers. *The EGS4 Code System Stanford Linear Accelerator*. Report SLAC-265, Stanford, 1985.
- [88] K. Otto. Volumetric modulated arc therapy: IMRT in a single gantry arc. *Med. Phys.*, 35:310–317, 2008.
- [89] L. Paelinck, N. Reynaert, H. Thierens, W. De Neve, and C. De Wagter. Experimental verification of lung dose with radiochromic film: comparison with Monte Carlo simulations and commercially available treatment planning. *Phys. Med. Biol.*, 50:2055–2069, 2005.
- [90] J. Pena, D. M. González-Castaño, F. Gómez, F. Sánchez-Doblado, and G. H. Hartmann. Automatic determination of primary electron beam parameters in Monte Carlo simulation. *Med. Phys.*, 34:1076–1084, 2007.
- [91] P. L. Petti, M. S. Goodman, T. A. Gabriel, and R. Mohan. Investigation of buildup dose from electron contamination of clinical photon beams. *Med. Phys.*, 10:18–24, 1983.
- [92] E. B. Podgorsak, P. Metcalfe, and J. Van Dyk. Medical Accelerators. In J. Van Dyk, editor, *The Modern Technology of Radiation Oncology*. Medical Physics Publishing, Madison Wisconsin, USA, 1999.

- [93] S. P. Raya and J. K. Udupa. Shape-based interpolation of multidimensional objects. *IEEE Transactions on Medical Imaging*, 9: 32–42, 1990.
- [94] N. Reynaert, M. Coghe, B. De Smedt, L. Paelinck, B. Vanderstraeten, and W. De Gersem. The importance of accurate linear accelerator head modelling for IMRT Monte Carlo calculations. *Phys. Med. Biol.*, 50:831–846, 2005.
- [95] M. Rodriguez, J. Sempau, and L. Brualla. A combined approach of variance-reduction techniques for the efficient Monte Carlo simulation of linacs. *Phys. Med. Biol.*, 57:3013–3024, 2012.
- [96] D. W. O. Rogers and R. Mohan. Questions for comparisons of clinical Monte Carlo codes. In W. Schlegel and T. Bortfeld, editors, *Proceedings of the 13th ICCR*, pages 120–122. Springer-Verlag, Heidelberg, Germany, 2000.
- [97] E. Rosenblatt, J. Izewska, Y. Anacak, Y. Pynda, P. Scalliet, M. Boniol, and P. Autier. Radiotherapy capacity in European countries: an analysis of the Directory of Radiotherapy Centers (DIRAC) database. *Lancet Oncol.*, 14:e79–e86, 2013.
- [98] R. Rubinstein and D. P. Kroese. *Simulation and the Monte Carlo method. 2nd Edition*. John Wiley & Sons, Hoboken New Jersey, USA, 2007.
- [99] F. Salvat, J. M. Fernández-Varea, and J. Sempau. *PENELOPE 2011—A Code System for Monte Carlo Simulation of Electron and Photon Transport*. OECD Nuclear Energy Agency, Issy de Molineaux, France, 2011.
- [100] A. Samuelsson, S. Hyödynmaa, and K. A. Johansson. Dose accuracy check of the 3d electron beam algorithm in a treatment planning system. *Phys. Med. Biol.*, 43:1529–1544, 1998.
- [101] A. E. Schach von Wittenau, L. J. Cox, P. M. Bergstrom, Jr. W. P. Chandler, C. L. Hartmann Siantar, and R. Mohan. Correlated histogram representation of Monte Carlo derived medical accelerator photon-output phase space. *Med. Phys.*, 26:1196–1211, 1999.
- [102] E. C. Schreiber and B. A. Faddegon. Sensitivity of large-field electron beams to variations in a Monte Carlo accelerator model. *Phys. Med. Biol.*, 50:769–778, 2005.
- [103] J. Seco and F. Verhaegen. *Monte Carlo techniques in radiation therapy*. CRC Press, Boca Raton, USA, 2013.

- [104] J. Sempau, S. Wilderman, and A. Bielajew. DPM, a fast, accurate Monte Carlo code optimized for photon and electron radiotherapy treatment planning dose calculations. *Phys. Med. Biol.*, 45: 2263–2291, 2000.
- [105] J. Sempau, A. Sánchez-Reyes, F. Salvat, H. Oulad ben Tahar, S. B. Jiang, and J. M. Fernández-Varea. Monte Carlo simulation of electron beams from an accelerator head using PENELOPE. *Phys. Med. Biol.*, 46:1163–1186, 2001.
- [106] J. Sempau, A. Badal, and L. Brualla. A PENELOPE-based system for the automated Monte Carlo simulation of clinacs and voxelized geometries—application to far-from-axis fields. *Med. Phys.*, 38:5887–5895, 2011.
- [107] D. Sheikh-Bagheri and D. W. O. Rogers. Sensitivity of megavoltage photon beam Monte Carlo simulations to electron beam and other parameters. *Med. Phys.*, 29:379–390, 2002.
- [108] D. Sheikh-Bagheri, D. W. Rogers, C. K. Ross, and J. P. Seuntjens. Comparison of measured and Monte Carlo calculated dose distributions from the NRC linac. *Med. Phys.*, 27:2256–2266, 2000.
- [109] G. W. Sherouse, K. Novins, and E. L. Chaney. Computation of digitally reconstructed radiographs for use in radiotherapy treatment design. *Int. J. Rad. Oncol. Biol. Phys.*, 18:651–658, 1990.
- [110] D. Shreiner, G. Sellers, J. Kessenich, and B. Licea-Kane. *OpenGL Programming Guide. The official guide to learning OpenGL version 4.3*. Addison-Wesley, Upper Saddle River, New Jersey, USA, 2013.
- [111] R. L. Siddon. Calculation of the radiological depth. *Med. Phys.*, 12:84–87, 1985.
- [112] J. V. Siebers, P. J. Keall, A. E. Nahum, and R. Mohan. Converting absorbed dose to medium to absorbed dose to water for Monte Carlo based photon beam dose calculations. *Phys. Med. Biol.*, 45: 983–995, 2000.
- [113] M. Sikora and M. Alber. A virtual source model of electron contamination of a therapeutic photon beam. *Phys. Med. Biol.*, 54:7329–7344, 2009.
- [114] M. R. Sontag and J. R. Cunningham. Corrections to absorbed dose calculations for tissue inhomogeneities. *Med. Phys.*, 4:431–436, 1977.
- [115] H. Steve. Trilinear interpolation. In Paul S. Heckbert, editor, *Graphics Gems IV*. Academic Press, London, UK, 1994.

- [116] J. Sulkimo and J. Vuoskoski. Particle tracking in sophisticated CAD models for simulation purposes. *Nucl. Instrum. Meth. Phys. Res., A* 371:434–438, 1996.
- [117] Varian Oncology Systems. *CadPlan 6.2 Manual. External Beam Modeling Physics*. Varian Oncology Systems, Palo Alto, California, USA, 2000.
- [118] Varian Oncology Systems. *Eclipse algorithms reference guide*. Varian Oncology Systems, Palo Alto, California, USA, 2010.
- [119] A. Tzedakis, J. E. Damilakis, M. Mazonakis, J. Stratakis, H. Varveris, and N. Gourtsoyiannis. Influence of initial electron beam parameters on Monte Carlo calculated absorbed dose distributions for radiotherapy photon beams. *Med. Phys.*, 31:907–913, 2004.
- [120] M. Udale. A Monte Carlo investigation of surface doses for broad electron beams. *Phys. Med. Biol.*, 33:939–953, 1988.
- [121] W. Ulmer and D. Harder. A triple Gaussian pencil beam model for photon beam treatment planning. *Z. Med. Phys.*, 5:25–30, 1995.
- [122] W. Ulmer, J. Pyyry, and Kaissl W. A 3D photon superposition/-convolution algorithm and its foundation on results of Monte Carlo calculations. *Phys. Med. Biol.*, 50:1767–1790, 2005.
- [123] J. Van Dyk. *The Modern Technology of Radiation Oncology*. Medical Physics Publishing, Madison Wisconsin, USA, 1999.
- [124] A. Van Esch, L. Tillikainen, J. Pyykkonen, M. Tenhunen, H. Helminen, S. Siljamäki, J. Alakuijala, M. Paiusco, M. Iori, and D. P. Huyskens. Testing of the analytical anisotropic algorithm for photon dose calculation. *Med. Phys.*, 33:4130–4148, 2006.
- [125] B. Vanderstraeten, P. W. Chin, M. Fix, A. Leal, G. Mora, N. Reynaert, J. Seco, M. Soukup, E. Spezi, W. De Neve, and H. Thierens. Conversion of CT numbers into tissue parameters for Monte Carlo dose calculations: a multi-centre study. *Phys. Med. Biol.*, 52:539–562, 2007.
- [126] F. Verhaegen and S. Devic. Sensitivity study for CT image use in Monte Carlo treatment planning. *Phys. Med. Biol.*, 50:937–946, 2005.
- [127] B. Walters, I. Kawrakow, and D. W. O. Rogers. DOSXYZnrc Users Manual. In *NRCC Report PIRS-794revB*. National Research Council of Canada, Ottawa, Canada, 2009.

- [128] B. R. B. Walters, I. Kawrakow, and D. W. O. Rogers. History by history statistical estimators in the BEAM code system. *Med. Phys.*, 29:2745–2752, 2002.
- [129] B. R. B. Walters, R. Kramer, and I. Kawrakow. Dose to medium versus dose to water as an estimator of dose to sensitive skeletal tissue. *Phys. Med. Biol.*, 55:4535–4546, 2010.
- [130] T. A. Wareing, J. M. McGhee, J. E. Morel, and S. D. Pautz. Discontinuous finite element  $S_n$  methods on three-dimensional unstructured grids. *Nucl. Sci. Engr.*, 138:256–268, 2001.
- [131] J. Yang, J. S. Li, L. Qin, W. Xiong, and C-M. Ma. A virtual photon energy fluence model for Monte Carlo dose calculation. *Phys. Med. Biol.*, 49:2657–2673, 2004.
- [132] C. X. Yu. Intensity-modulated arc therapy: principles, technologies and clinical implementation. *Phys. Med. Biol.*, 40:1435–1449, 1995.



UNIVERSITAT  
POLITÈCNICA  
DE CATALUNYA  
2015