International Journal of Radiation Biology, January–February 2012; 88(1–2): 200–208
© 2012 Informa UK, Ltd.
ISSN 0955-3002 print / ISSN 1362-3095 online
DOI: 10.3109/09553002.2011.627977

informa
healthcare

An implementation to read and write IAEA phase-space files in GEANT4-based simulations

Miguel Antonio Cortés-Giraldo[1], José Manuel Quesada[1], María Isabel Gallardo[1] & Roberto Capote[2]

[1]*Department of Atomic, Molecular and Nuclear Physics, University of Seville, Seville, Spain and* [2]*NAPC – Nuclear Data Section, International Atomic Energy Agency, Vienna, Austria*

Abstract

Purpose: **To develop a stand-alone code to make any application coded with the GEANT4 (GEometry ANd Tracking, version 4) toolkit capable of reading and writing phase-space (phsp) files in the format created by the IAEA (International Atomic Energy Agency), so that the exchange of phsp files between other validated Monte Carlo (MC) codes and GEANT4 is possible.** ***Methods*****: We present a stand-alone code, written in C++ object-oriented language, developed in a way that ensures the compatibility with future versions of the IAEA phsp format. The aim of the reader part is to get the information from a given IAEA phsp file and create the primary particles in a GEANT4 user application. On the other hand, the writer part of the code is the responsible for writing the IAEA phsp files during a run of the GEANT4 application.** ***Results*****: A testing simulation was written with GEANT4 to verify the performance of this code, with satisfactory results. An example of use in a GEANT4 application which simulates the treatment head of a radiotherapy linear electron accelerator (linac) is also shown, comparing dose calculations with experimental data.** ***Conclusions*****: This stand-alone package, which can be used in any GEANT4 application, allows the exchange of validated phsp files between different MC codes and the use of phsp data from many different accelerators and fields in dosimetry studies. Furthermore, it also offers additional utilities of interest in medical applications.**

Keywords: GEANT4, phase-space file, IAEA phsp format, radiotherapy

Introduction

The use of Monte Carlo (MC) techniques has become a standard tool in recent years for the computer simulation of the clinical components of linear accelerators (linacs), notably beam generation and collimation, and for the study of the interaction of radiation beams with patients and dosimetric materials (Rogers 2006). The improvement of the accuracy of radiotherapy treatment plannings requires a well characterised radiation source. As there is a limited number of commercial therapy units, in most of the calculations some parts of the linac treatment head are repeated (e.g., target, primary collimator and flattening filter). For this reason, along with the need of saving CPU (Central Processing Unit) time, the capability of using phase-space (phsp) files obtained and validated previously is extremely useful. A phsp file is a collection of data specifying the particle position, direction, energy, type and extra variables for each particle crossing a plane defined as phase-space plane (or scoring plane). Its use as a primary generator in a simulation avoids the need of a detailed description of machine geometry and materials, which are often a commercial secret and are seldom shared with users. The availability of phsp data also allows an easy use of different beam qualities in dosimetry studies, such as detector characterisation or development of dose protocols.

The International Atomic Energy Agency (IAEA) has promoted a project aimed to build a database to disseminate representative phsp files of linear accelerators and Co-60 units used in external radiotherapy by compiling existing data that have been properly validated (see the IAEA phsp project web site at http://www-nds.iaea.org/phsp). The IAEA phsp format has been designed and agreed by an international expert committee for its use in medical applications (Capote et al. 2006). Such a format has been implemented in recent releases of general purpose MC codes like BEAMnrc/EGSnrc ('EGS' stands for 'Electron Gamma Shower' and 'nrc' stands for 'National Research Council of Canada'; 'BEAMnrc' is just the name of a user code of EGSnrc) (Rogers et al. 1995, Kawrakow 2000, Kawrakow et al. 2009) and PENELOPE (PENetration and Energy LOss of Positrons and Electrons) (Salvat et al. 2008), which are considered the state-of-the-art for the coupled electron-photon transport in medical applications. The work described here is the implementation of two classes, written in C++ object-oriented language, to read and write IAEA phsp files in GEANT4 (GEometry ANd Tracking, version 4) (Agostinelli et al. 2003, Allison et al. 2006) user applications. In object-oriented programming, a class is an expanded concept of a data structure which, instead of holding only data, holds both data and functions, usually known as methods.

Correspondence: Miguel Antonio Cortés-Giraldo, Dep. Física Atómica, Molecular y Nuclear, University of Sevilla, PO Box 1065, Sevilla 41080, Spain.
Tel: + 34 954550928. Fax: + 34 954554445. E-mail: miancortes@us.es

(Received 30 March 2011; revised 17 September 2011; accepted 23 September 2011)

The GEANT4 toolkit is a MC code for the simulation of the passage of particles through matter. Although GEANT4 was originally created for applications in high-energy physics, it has been used in many other fields nowadays, namely nuclear physics, medical and space applications. The GEANT4 validation for electromagnetic and medical applications has been undertaken by several groups (Larsson et al. 2005, Poon and Verhaegen 2005, Faddegon et al. 2008a, 2008b, 2009, Cirrone et al. 2010, Lechner et al. 2010); furthermore, the GEANT4 toolkit is being extensively used in proton therapy applications (Paganetti et al. 2004, 2008, Aso et al. 2005, Newhauser et al. 2005). Therefore, built-in capabilities to store proton and/or secondary electron phsp files in the IAEA format are welcome, as they allow the exchange of phsp data which could be used for charged particle detector studies and for dosimetry in proton, electron and photon irradiations, among other applications.

We present a code created for reading and writing IAEA phsp files with GEANT4. This code has been endorsed by the IAEA as a general and portable implementation. The classes presented in this work use directly the public functions of the IAEA classes (a public function can be invoked out of the scope of the class where it is implemented), without a reimplementation of the IAEA format in the GEANT4 classes. This approach ensures the compatibility with future versions of the IAEA phsp format as long as public function calls are maintained. The capability to read and write IAEA phsp files was included in the GEANT4-based framework called GAMOS (Geant4-based Architecture for Medicine-Oriented Simulations), which has been developed to simulate medical devices without the need of coding in C++ (Arce et al. 2008), but the classes developed to read and write IAEA phsp files are not available as an independent package.

The next section presents a description of the classes including instructions regarding its use in a GEANT4 application. The third section shows results obtained with a test simulation and with a GEANT4 application which simulates the treatment head of a real radiotherapy linac. Finally, the conclusions are drawn in the fourth section.

IAEA phsp classes for GEANT4

A design scheme of the IAEA phsp implementation for GEANT4, following the UML (Unified Modelling Language) notation, is shown in Figure 1. The public functions of the IAEA classes (available at the IAEA phsp project web site), are used directly in our classes coded for GEANT4, called *G4IAEAphspReader* and *G4IAEAphspWriter*. Since it was not necessary to reimplement the IAEA functions in GEANT4, our work is protected against future internal changes or upgrades of the IAEA phsp format, provided that the IAEA public functions never change their names and/or arguments.

To use our classes in a GEANT4 user application, one needs to compile it together with our classes and the IAEA files. Figure 1 also shows the relationship of use between the user actions (UA) of the GEANT4 specific application and the classes described in this work. Details of such relationships are given in the following subsections.

Table I presents a list of the public functions defined in the IAEA phsp classes. Their descriptions can be found in the IAEA phsp source files. These functions are used in our classes as follows. In the right column, an 'R' stands for IAEA functions used in the reader class (*G4IAEAphspReader*), whereas a 'W' stands for IAEA functions invoked in the writer class (*G4IAEAphspWriter*). The reader interested in further details about the IAEA phsp format may find additional information at the IAEA phsp project website (http://www-nds.iaea.org/phsp).

Reading phase-space files

There are several primary particle generators available in GEANT4, being those defined in *G4ParticleGun* and

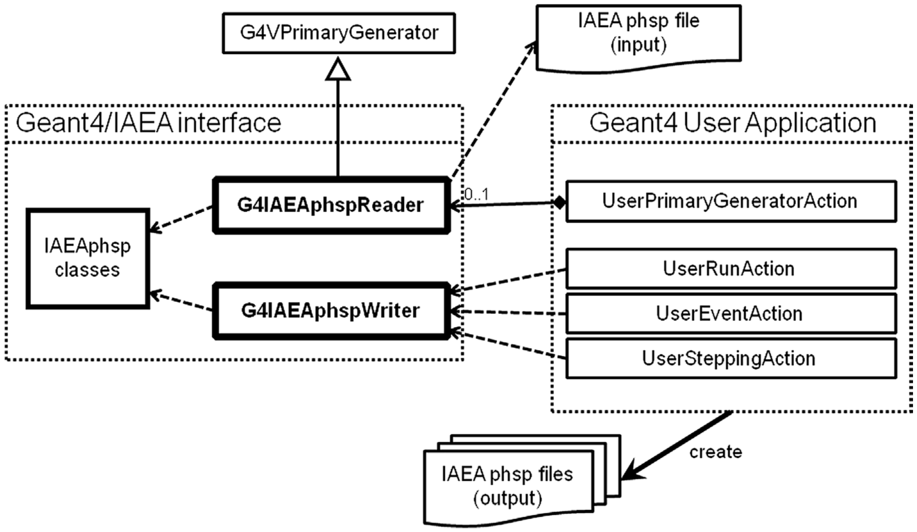

Figure 1. Scheme in UML notation of the main relationship between our classes (*G4IAEAphspReader* and *G4IAEAphspWriter*), classes of a GEANT4 user application and the IAEAphsp classes. *G4IAEAphspReader* class is derived from *G4VPrimaryGenerator* class of the GEANT4 toolkit (solid line with a triangular empty arrowhead), uses methods implemented in the IAEAphsp classes and accesses an IAEA phsp file (dashed arrows); further, a pointer to a *G4IAEAphspReader* object must be included among the data members of the *UserPrimaryGeneratorAction* class (solid line with a black diamond on the right side and an arrowhead on the left side). *G4IAEAphspWriter* is used by three UA of the GEANT4 application and uses methods of the IAEAphsp classes as well (dashed arrows). With this configuration, the GEANT4 application is able to create IAEA phsp files (solid arrow at the bottom).

Table I. Usage of the public functions defined in the IAEA classes.

IAEA function	Class
iaea_check_file_size_byte_order()	R
iaea_copy_header()	W
iaea_destroy_source()	R/W
iaea_get_constant_variable()	R
iaea_get_extra_numbers()	R
iaea_get_max_particles()	R
iaea_get_maximum_energy()	R
iaea_get_particle()	R
iaea_get_total_original_particles()	R
iaea_get_type_extra_variables()	R
iaea_get_used_original_particles()	R
iaea_new_source()	R/W
iaea_print_header()	W
iaea_set_constant_variable()	W
iaea_set_extra_numbers()	W
iaea_set_paralell()	R
iaea_set_record()	R
iaea_set_total_original_particles()	W
iaea_set_type_extrafloat_variable()	W
iaea_set_type_extralong_variable()	W
iaea_update_header()	W
iaea_write_particle()	W

G4GeneralParticleSource classes the most widely used. Both are derived from *G4VPrimaryGenerator* abstract base class (an abstract base class contains at least one method which must be implemented in a derived class) and can be used in the *UserPrimaryGeneratorAction* of the GEANT4 concrete application. *G4IAEAphspReader* class was also derived from *G4VPrimaryGenerator* with the goal of retrieving the information from an IAEA phsp file and create the corresponding primary particles for the GEANT4 simulation.

Among the utilities that *G4IAEAphspReader* class has, the main ones are:

- To perform a statistical analysis properly, it is mandatory to take correlations between particles into account. To keep correlations, this class generates all the particles that may be correlated in the same event of the simulation. Two different types of correlations are considered: (a) When different particles come from the same original history, and (b) when the same particle is reused (recycled) several times to increase statistics. In other words, all the particles belonging to the same original history are created as primaries at the same event, and each one is repeated (recycled) the number of times set in the GEANT4 simulation. Although it has been documented that recycling of particles should be avoided if possible (Walters et al. 2002), recycling of particles stored in a phsp file is often used in Monte Carlo simulations. The number of times that each particle is recycled can be set by means of *SetTimesRecycled(G4int n)* method, where n indicates that each particle will be used $(n+1)$ times. Furthermore, in cases for which the phsp presents symmetry around the central axis, it is possible to perform recycling taking into account such symmetry by randomizing the azimuthal angle.
- The IAEA phsp format offers the possibility of defining an extra-integer variable called *incremental history number* (also known as *n_stat*), which can be used in EGSnrc and PENELOPE simulations. This variable indicates the number of original histories employed to get a given particle after the one stored previously in the phsp. Thus, $n\_stat = 0$ means that both previous and current particle come from the same original history (and hence are correlated). In our class, this variable is kept if it is defined in the IAEA phsp file which is being read, with the aim of book-keeping the number of original histories simulated.
- The phsp file can be divided into fragments with *SetTotalParallelRuns(G4int)* method, where obviously *G4int* refers to the integer variable passed as argument, which sets the number of fragments. Therefore, one can prepare parallel runs with the same phsp file by using different fragments for different runs. The particular fragment can be selected with *SetParallelRun(G4int)* method.
- Translations and rotations can be applied to the positions and momenta stored in the phsp file. Rotations can be performed around the Cartesian axes of the global reference frame in any order. In addition, it is possible to define the position of the isocentre and the vector of the rotation axis of the gantry and the treatment head of the medical linac which created the phsp file, so that rotations around the isocentre can be applied in a straightforward way.

In order to use an IAEA phsp as a primary generator in a GEANT4 concrete application, the following requirements must be fulfilled in the Primary Generator class of such an application:

1) Declare a pointer to a *G4IAEAphspReader* object in the header file as a data member of the Primary Generator class.
2) In the Primary Generator class constructor: A *G4IAEAphspReader* object must be instantiated with the proper argument (name of the IAEA phsp file). Only one IAEA phsp file can be used for a generator.
3) In the GeneratePrimaries() method of Primary Generator class: The pointer to the G4IAEAphspReader object must be used to invoke the GeneratePrimaryVertex() method of our class.

A reference guide for this class can be found in the user's guide of our classes (Cortés-Giraldo et al. 2009b) as well as examples on the use of the *G4IAEAphspReader* utilities.

Writing phase-space files

The writing of IAEA phsp files is managed by *G4IAEAphspWriter* class, which can create several IAEA phsp files during the same run. It has been designed following a *singleton* class pattern, since the same object of this class must be used in three different UA of the GEANT4 application. The singleton is accessed with a static method called *GetInstance()*. In software engineering, a singleton is a design pattern which restricts the instantiation of a class to only one object; because of this restriction, the object must be accessed by means of a static method, which is a type of method that acts at the class level instead of at the instance level, so that it can be used even when no instantiations have taken place yet.

As it is explained in the IAEA documentation (Capote et al. 2006), the phsp is written in a binary file of extension "*.IAEAphsp*". For each particle crossing the phsp plane, it stores the energy E, statistical weight w, the Cartesian components of the position (x, y, z), the direction cosines of the linear momentum (u, v, w) and extra variables, either of integer (*extraints*) or floating point (*extrafloats*) type. To avoid particles passing multiple times through the phsp plane (multiple passers), *G4IAEAphspWriter* class stores each particle only the first time it crosses the phsp plane. Certain variables can be declared as constant to save space in disk. This class defines z as constant and includes the incremental history number as an integer extra variable. In total, each record is allocated in 33 bytes with the default configuration. The requested IAEA phsp files are generated at the end of the run.

To write IAEA phsp files with a given GEANT4 application, *G4IAEAphspWriter* class must be used as indicated below:

(1) The following three UA must be defined: "Run Action" class (derived from *G4UserRunAction*), "Event Action" class (derived from *G4UserEventAction*) and "Stepping Action" class (derived from *G4UserSteppingAction*).
(2) In the "Run Action" class:
  - In *BeginOfRunAction()* method: The *SetZStop()* method of the singleton should be called once for each scoring plane, passing the z value as argument. Finally, the particular method of the singleton, also called *BeginOfRunAction()*, must be used.
  - In *EndOfRunAction()* method: The *EndOfRunAction()* method of the singleton must be invoked.

(3) In the *BeginOfEventAction()* method of the "Event Action" class, the *BeginOfEventAction()* method of *G4IAEAphspWriter* singleton must be used.
(4) Similarly, in the *UserSteppingAction()* method of the "Stepping Action" class, the *UserSteppingAction()* method of *G4IAEAphspWriter* singleton must be invoked.

The reference guide and further explanations on how to use *G4IAEAphspWriter* class are provided in the user's guide as well (Cortés-Giraldo et al. 2009b).

Results

Testing

Our developed classes have been tested with phsp files obtained from the IAEA database (see IAEA phsp project web site). It is a good test to verify that the IAEA phsp file created with our code is identical to the original IAEA phsp file read in the GEANT4 application. With this purpose, we coded a very simple GEANT4 application. The geometry consists of a 1-micrometer-thick slab made of vacuum placed perpendicular with respect to the z axis at $z = z_{\text{phsp}}$, where z_{phsp} is the source-to-surface distance (SSD) recorded in the original IAEA phsp file; therefore, the particles read from the IAEA phsp file were generated exactly in the central plane of the vacuum slab. The physics list, which is the name used in the GEANT4 terminology for the class where the set of physics models and processes is defined, included only transport of particles, since our only interest was the transport of particles through the scoring plane. Thus, the generated particles were transported up to the boundary of the slab (at $z = z_{\text{phsp}} + 0.5$ μm) in one step. By means of *G4IAEAphspWriter* class, we defined a scoring plane at $z_{\text{score}} = z_{\text{phsp}} + 0.25$ μm, so that we ensured that all the particles crossed the scoring plane during their first step. The analysis of the results obtained was realized using the data processing framework ROOT (see ROOT website), developed at CERN (in French, *Centre Européen pour la Recherche Nucléaire*, Geneva, Switzerland).

This testing application was run for two phsp files, available in the IAEA database, corresponding to a 3×3 cm^2 and a 10×10 cm^2 radiation field, respectively. Both were generated by a BEAMnrc/EGSnrc application which simulates the treatment head of a PRIMUS linac (Siemens AG, Munich, Germany) working at the 6-megavolt (MV) photon mode. The phsp plane was placed at SSD = 95 cm and directional

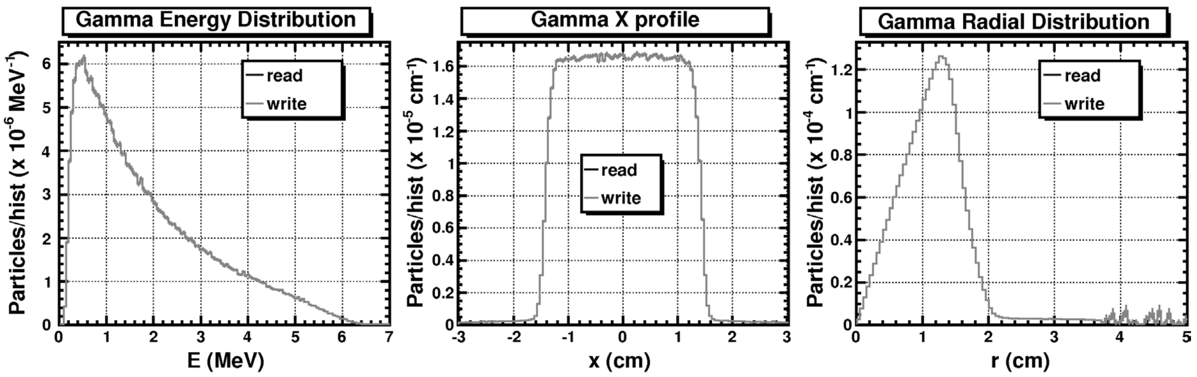

Figure 2. Histograms of particle weights obtained with a phsp file stored in the IAEA database (code PHSP 109) which characterises a 3×3 cm^2 field at SSD = 95 cm. Histograms of the original and output file are plotted in black and grey, respectively, but they are not distinguishable since, as expected, they exactly overlap. The histogram values are normalised to the number of original histories stored in the file. The statistical uncertainty associated to each bin is represented with error bars. Left: Energy distribution of photons with radial coordinate, r, lower than 1 cm; bin values are calculated in units of megaelectronvolt. Centre: Distribution of photons according to the x coordinate; each bin represents the integral of the fluence from $y = -5$ mm to $y = 5$ mm; values are presented in units of centimetre. Right: Radial coordinate histogram; the value of each bin corresponds to the integral of the fluence from the lower to the upper limit for all values of the azimuthal angle; the histogram values are presented in units of centimetre.

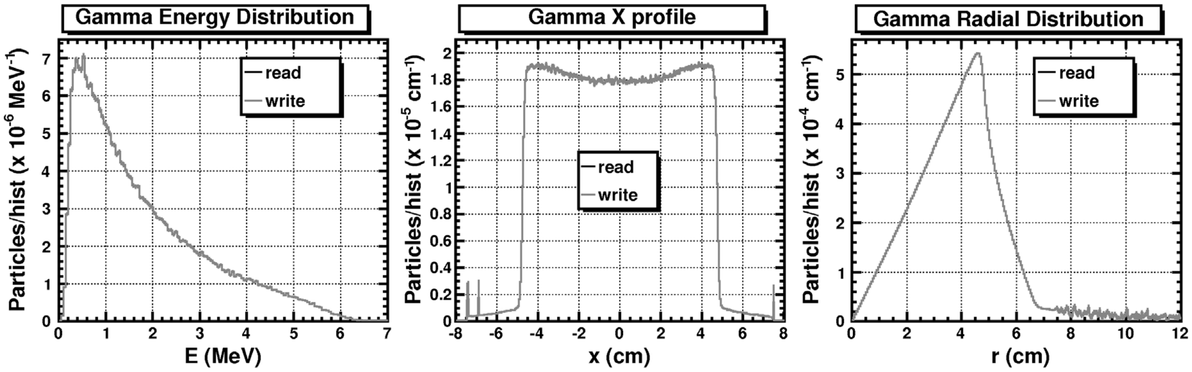

Figure 3. Histograms of particle weights obtained with a phsp file stored in the IAEA database (code PHSP 100) which characterises a 10×10 cm^2 field at SSD = 95 cm. Histograms of the original and output file are plotted in black and grey, respectively, but they are not distinguishable since, as expected, they exactly overlap. The histogram values are normalised to the number of original histories stored in the file. The statistical uncertainty associated to each bin is represented with error bars. Left: Energy distribution of photons with radial coordinate, r, lower than 1 cm; bin values are calculated in units of megaelectronvolt. Centre: Distribution of photons according to the x coordinate; each bin represents the integral of the fluence from $y = -5$ mm to $y = 5$ mm; values are presented in units of centimetre. Right: Radial coordinate histogram; the value of each bin corresponds to the integral of the fluence from the lower to the upper limit for all values of the azimuthal angle; the histogram values are presented in units of centimetre.

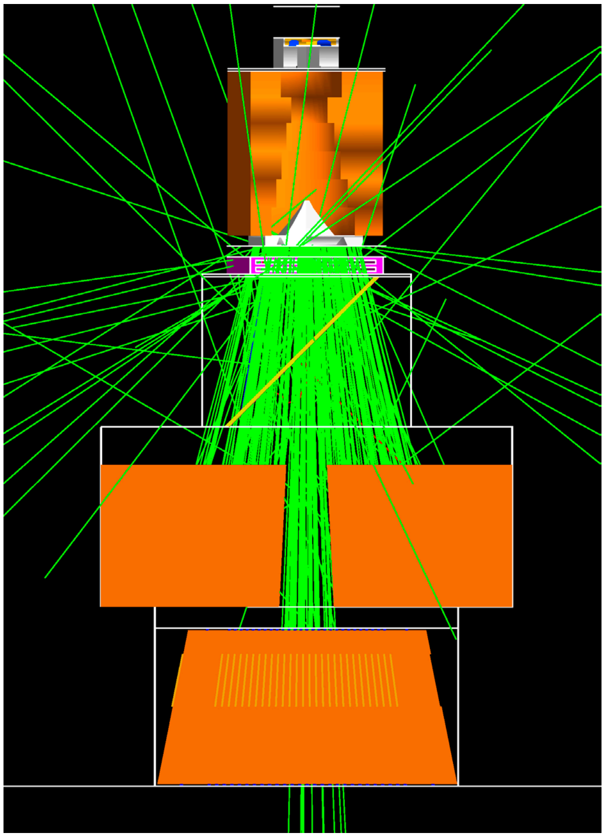

Figure 4. Geometry model of the Siemens PRIMUS linac treatment head, 6-MV photon mode, installed at the Virgen Macarena Hospital (Seville, Spain). It is composed by the following component modules (from top to bottom): tungsten target (gray), primary collimator (orange-brown), flattening filter (white cone), monitor chamber (purple), tilted mirror (yellow), jaws (orange) and multi-leaf collimator (orange). Each component module is placed within a mother box (white lines) to achieve a good hierarchy in the geometry model. The electron beam enters from the upper side. The trajectories of gammas recorded in a phsp file defined by a scoring plane placed perpendicular to the central axis below the flattening filter (SSD = 9.97 cm) are also shown (green lines); a total of 100 original histories are plotted. This figure is reproduced in colour in the online version of *International Journal of Radiation Biology*.

bremsstrahlung splitting (DBS) was applied. DBS is a variance reduction technique which splits each photon created by bremsstrahlung and pointing to a previously defined region of interest into a certain number of photons, N_{split}, of statistical weight $1/N_{\text{split}}$. In this case, the region of interest (splitting field) was a circle placed at the isocentre plane ($SSD = 100$ cm) and centred at the intersection point with the central axis. For the 3×3 cm^2 radiation field the splitting radius was 4 cm and for the 10×10 cm^2 field the splitting radius was 8 cm.

Figure 2 shows three spectra characterizing the photon fluence of the 3×3 cm^2 field, obtained for both the phsp file available at the IAEA database and the phsp file created with our testing application. In these plots, the direction of the beam is oriented along the z axis. The left plot represents the energy distribution of photons crossing the scoring plane within $r < 1$ cm, being $r = (x^2 + y^2)^{1/2}$ the radial coordinate (distance to the central axis). The central plot represents the distribution in the x coordinate integrated from $y = -5$ mm to $y = +5$ mm (X profile). Both plots look exactly as expected. The right plot shows the distribution in the radial coordinate, where the value of each bin is given by the fluence integrated from its lower limit (r_{low}) to its upper limit (r_{high}) for all values of the azimuthal angle. It can be observed that the bins corresponding to $r > 4$ cm present larger statistical uncertainties. This effect occurs because DBS was not applied for these photons during the simulation (they are out of the splitting field); thus their statistical weights are much higher compared to those photons crossing with $r < 4$ cm; consequently the statistical uncertainty of the bins out of the splitting field increases. These photons with larger weights are usually referred as **fat** photons. The number of particles stored was the same for both the original and the IAEA phsp file created with our testing application (3,109,575), as well as all the statistical information (concerning total weight, energy and other variables) included in their corresponding IAEA header files.

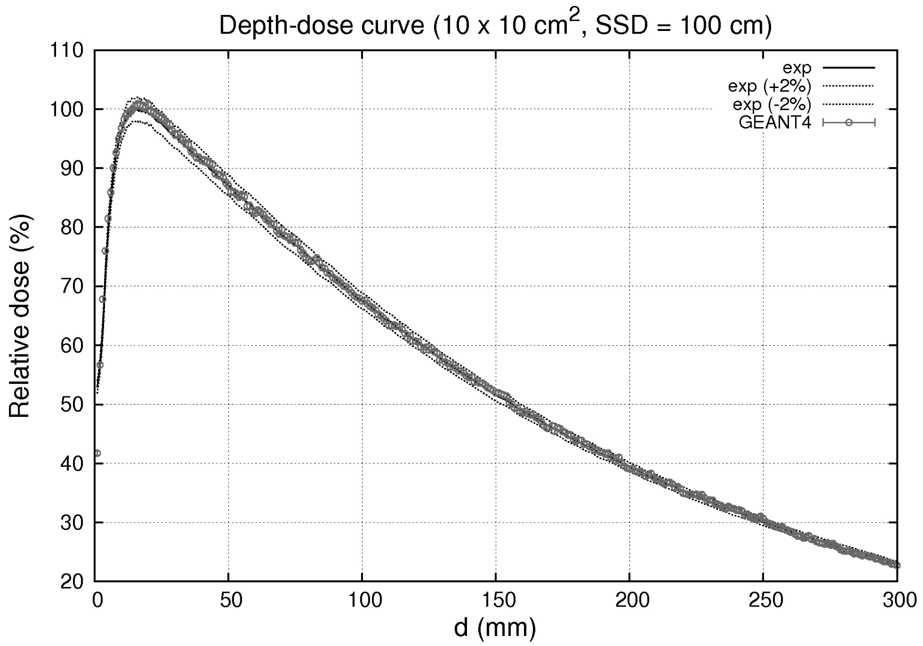

Figure 5. PDD curve obtained in water for a 10×10 cm^2 field ($SSD = 100$ cm). The dose normalised to the maximum value is plotted as a function of the depth, d, in water. The experimental values are depicted with a solid black line; dotted lines show the $\pm 2\%$ level of the experimental values of dose. GEANT4 calculations are represented with grey empty circles; error bars show the 1σ level of the statistical uncertainties of the MC calculations.

Figure 3 shows the same histograms as in Figure 2 for the simulation of a 10×10 cm^2 radiation field at $SSD = 95$ cm. The energy distribution and the X profile look as expected. In this case, the radius of the splitting field (8 cm) explains the appearance of bins with higher uncertainties for $r > 8$ cm, as it was previously discussed. The new phsp file created with the testing application stored the same number of particles (39,641,628) and was confirmed to be identical to the original phsp file as well.

Implementation in a GEANT4 radiotherapy application

Our classes were implemented in a GEANT4 application developed to simulate the treatment head of a Siemens PRIMUS linac used at the Virgen Macarena Hospital (Seville, Spain).

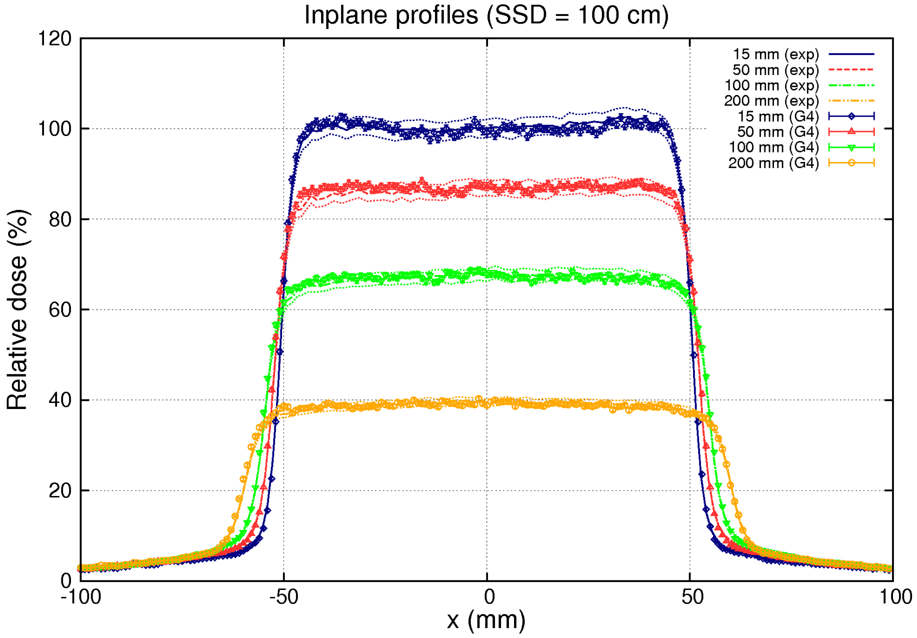

Figure 6. Lateral profiles (inplane direction) obtained in water for a 10×10 cm^2 field ($SSD = 100$ cm). The relative dose is plotted as a function of the x coordinate for depths in water of 1.5 cm (blue), 5.0 cm (red), 10.0 cm (green) and 20.0 cm (orange). For each case, the experimental values are depicted with a line (solid, dashed, dashed-dotted and dashed double-dotted, respectively); dotted lines show the $\pm 2\%$ level of the experimental values of dose. GEANT4 calculations are represented with empty symbols (diamonds, triangles, inverted triangles and circles, respectively); error bars show the 1σ level of the statistical uncertainties of the MC calculations. This figure is reproduced in colour in the online version of *International Journal of Radiation Biology*.

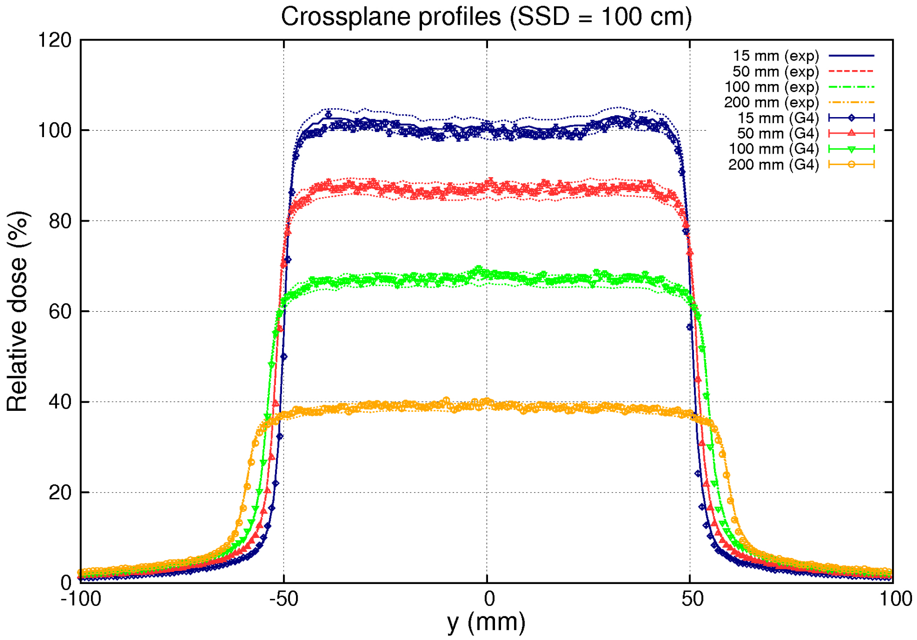

Figure 7. Lateral profiles (crossplane direction) obtained in water for a 10 × 10 cm^2 field (SSD = 100 cm). The relative dose is plotted as a function of the y coordinate for depths in water of 1.5 cm (blue), 5.0 cm (red), 10.0 cm (green) and 20.0 cm (orange). For each case, the experimental values are depicted with a line (solid, dashed, dashed-dotted and dashed double-dotted, respectively); dotted lines show the ±2% level of the experimental values of dose. GEANT4 calculations are represented with empty symbols (diamonds, triangles, inverted triangles and circles, respectively); error bars show the 1σ level of the statistical uncertainties of the MC calculations. This figure is reproduced in colour in the online version of *International Journal of Radiation Biology*.

The employed geometry model of the 6 MV photon mode of the linac is shown in Figure 4, where the different component modules can be distinguished; from top to bottom: tungsten target, primary collimator, flattening filter, monitor chamber, tilted mirror, jaws and multi-leaf collimator. In this model, the central axis is oriented along the z axis, the inplane direction (direction of movement of the jaws) along the x axis and the crossplane direction (direction of movement of the multi-leaf collimator) along the y axis. The tungsten target is placed at $z = 0$ and the beam propagates towards positive values of z.

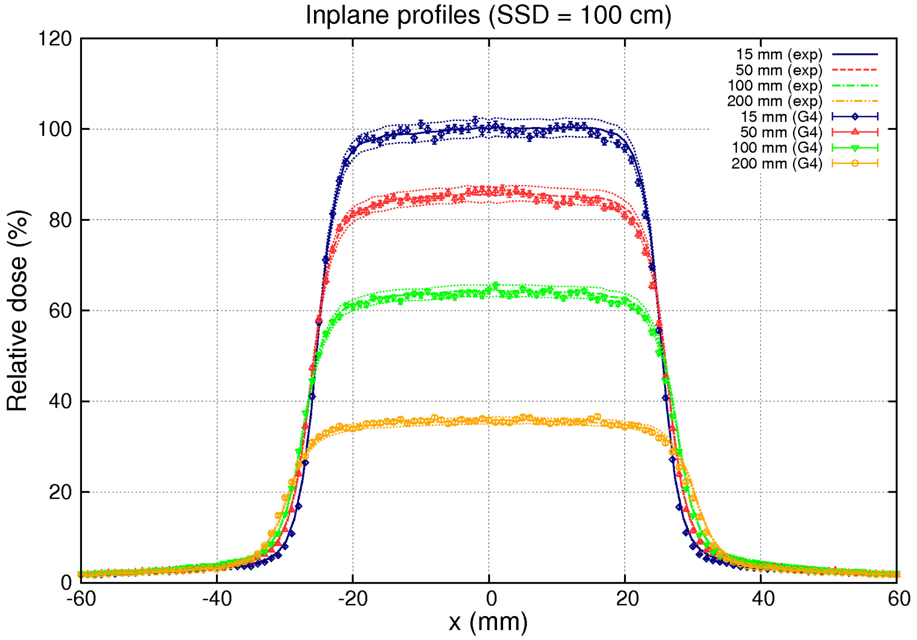

Figure 8. Lateral profiles (inplane direction) obtained in water for a 5 × 5 cm^2 field (SSD = 100 cm). The relative dose is plotted as a function of the x coordinate for depths in water of 1.5 cm (blue), 5.0 cm (red), 10.0 cm (green) and 20.0 cm (orange). For each case, the experimental values are depicted with a line (solid, dashed, dashed-dotted and dashed double-dotted, respectively); dotted lines show the ±2% level of the experimental values of dose. GEANT4 calculations are represented with empty symbols (diamonds, triangles, inverted triangles and circles, respectively); error bars show the 1σ level of the statistical uncertainties of the MC calculations. This figure is reproduced in colour in the online version of *International Journal of Radiation Biology*.

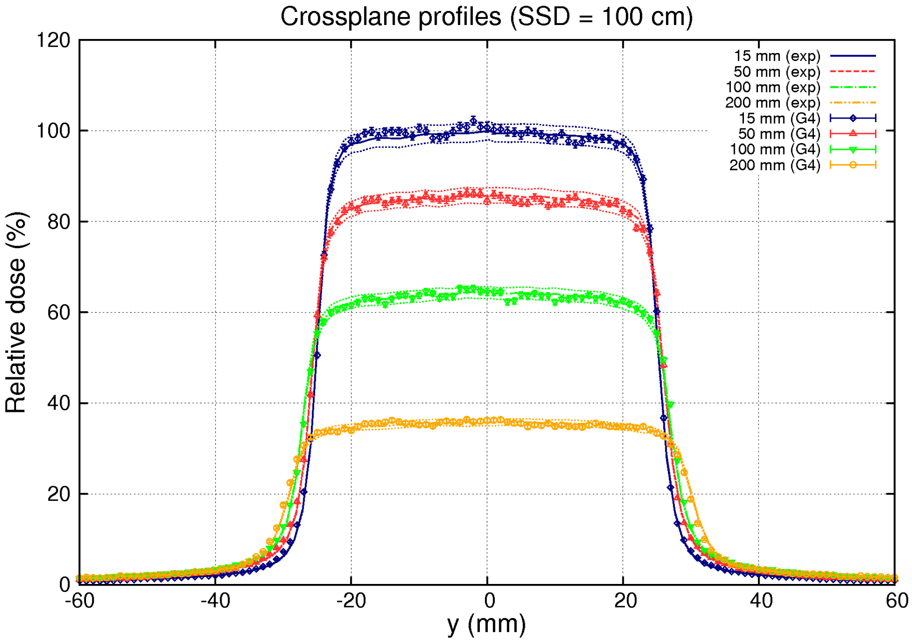

Figure 9. Lateral profiles (crossplane direction) obtained in water for a 5 × 5 cm^2 field (SSD = 100 cm). The relative dose is plotted as a function of the y coordinate for depths in water of 1.5 cm (blue), 5.0 cm (red), 10.0 cm (green) and 20.0 cm (orange). For each case, the experimental values are depicted with a line (solid, dashed, dashed-dotted and dashed double-dotted, respectively); dotted lines show the ±2% level of the experimental values of dose. GEANT4 calculations are represented with empty symbols (diamonds, triangles, inverted triangles and circles, respectively); error bars show the 1σ level of the statistical uncertainties of the MC calculations. This figure is reproduced in colour in the online version of *International Journal of Radiation Biology*.

Further details are described elsewhere (Cortés-Giraldo et al. 2009a). The MC simulation of the treatment head was performed in three steps:

1. Simulation of the motionless part of the treatment head devoted to characterise the fluence after the flattening filter; with this purpose, we defined a phsp plane at the end of it ($z = 9.97$ cm), which stored 4×10^9 original histories.
2. Simulation along the remaining part of the treatment head with the aim of producing phsp files to characterise the fluence at the exit of a given radiation field. The phsp stored during the first step is used here as primary generator with our code. Another scoring plane is defined at $z = 70$ cm to create the phsp files of the field. Each particle of the initial phsp file was recycled 5 times with rotational symmetry around the central axis.
3. Finally, the phsp file created at $z = 70$ cm is used to simulate the irradiation of a water tank used for dosimetry measurements. In this step, each particle was recycled 25 times.

The upper surface of the water tank (50 × 50 × 40 cm^3) was placed on the isocentre plane (SSD = 100 cm) for dose calculations. A percentage depth dose (PDD) curve and lateral profiles along inplane and crossplane directions were calculated and compared with experimental data. In our MC simulations, the dose was calculated using cylindrical water voxels with a radius of 4 mm and height of 1 mm, with its central axis oriented parallel to the direction of measurement. The experimental data were measured with a PTW 60008 Photon Dosimetry Diode (Radiation Products Design, Inc., Albertville, Minnesota, USA).

Figure 5 shows the experimental PDD curve obtained along the central axis for a 10 × 10 cm^2 radiation field and the corresponding MC calculations. The dose values were normalised to the maximum value, obtained at a depth, d, of 1.5 cm. The statistical uncertainty of the values calculated with MC is between 0.7% and 0.9% (1σ level). In the figure, it is clearly observed that the relative dose calculated with our GEANT4 application agrees within ±2% with the experimental data.

Lateral profiles along the inplane and crossplane directions at several depths (1.5 cm, 5.0 cm, 10.0 cm and 20.0 cm) are shown in Figures 6 and 7, respectively, for the same radiation field. In these figures, relative dose values were calculated with the normalisation factor used for the PDD. The statistical uncertainty (1σ) for the MC calculations at the centre of the profiles is between 0.7% ($d = 1.5$ cm) and 0.9% ($d = 20.0$ cm). Again, the agreement of our calculations with the experimental data is within a tolerance of 2%.

Finally, Figures 8 and 9 present inplane and crossplane profiles, respectively, obtained for a 5 × 5 cm^2 radiation field at the depths indicated previously (SSD = 100 cm). In this case, the relative dose values were calculated with respect to the dose value at the centre of the profiles at $d = 1.5$ cm. The statistical uncertainty of the MC calculated dose is similar to the previous case, as well as the agreement achieved between our GEANT4 calculations and the experimental values.

Summary and conclusion

A new code has been developed to make GEANT4 user applications capable of reading and writing IAEA phsp files.

The inheritance from *G4VPrimaryGenerator* abstract base class and a proper use of the public functions of the IAEA classes guarantee the performance and portability of the code presented in this work. *G4IAEAphspReader* class includes the possibility of applying any kind of spatial transformations, including rotations around the isocentre. The GEANT4 user has total freedom to define rotation axes and the isocentre position according to the simulated geometry. Correlations among particles in the phsp file coming from the same original history and correlations due to recycling are taken into account. This allows the GEANT4 user to make a rigorous statistical analysis to obtain an accurate uncertainty estimate of simulated physical quantities (e.g., dose). The IAEA phsp file can be divided into fragments for runs in parallel CPU.

The code described in this work is available online at the IAEA phsp project web site and aims to help GEANT4 users working in the field of dosimetry for radiotherapy. New aspects of interest and concern favour the use of GEANT4 applications in radiotherapy, namely the study of neutron contamination from high energy photon beams (Becker et al. 2005), as GEANT4 is capable to deal with the transport of all kind of particles through matter.

Acknowledgements

The authors are grateful to F. R. Palomo, from the School of Engineering, University of Sevilla (Seville, Spain), for his kind and useful suggestions during the preparation of this article and to Dr R. Arráns, from Virgen Macarena Hospital (Seville, Spain), for providing us with the experimental data used in this work. This work was supported by Junta de Andalucía and the Spanish Ministerio de Ciencia e Innovación under contracts P07-FQM-02894, FIS2008-04189 and FPA2008-04972-C03.

Declaration of interest

The authors report no conflicts of interest. The authors alone are responsible for the content and writing of the paper.

References

Agostinelli S, Allison J, Amako K, Apostolakis J, Araujo H, Arce P, Asai M, Axen D, Banerjee S, Barrand G, Behner F, Bellagamba L, Boudreau J, Broglia L, Brunengo A et al. (Geant4 Collaboration). 2003. GEANT4–a simulation toolkit. Nuclear Instruments and Methods in Physics Research A 506:250–303.

Allison J, Amako K, Apostolakis J, Araujo H, Arce Dubois P, Asai M, Barrand G, Capra R, Chauvie S, Chytracek R, Cirrone GAP, Cooperman G, Cosmo G, Cuttone G, Daquino GG et al. (Geant4 Collaboration). 2006. Geant4 developments and applications. IEEE Transactions in Nuclear Science 53(1):270–278.

Arce P, Rato P, Cañadas M, Lagares JI. 2008. GAMOS: A GEANT4-based easy and flexible framework for nuclear medicine applications. In: IEEE Nuclear Science Symposium Conference Record. 2008 October 19–25. Dresden, Germany. pp 3162–3168.

Aso T, Kimura A, Tanaka S, Yoshida H, Kanematsu N, Sasaki T, Akagi T. 2005. Verification of the dose distribution with GEANT4 simulation for proton therapy. IEEE Transactions on Nuclear Science 52(4):896–901.

Becker J, Brunckhorst E, Schmidt R. 2007. Photoneutron production of a Siemens Primus linear accelerator studied by Monte Carlo methods and a paired magnesium and boron coated magnesium ionization chamber system. Physics in Medicine and Biology 52:6375–6387.

Capote R, Jeraj R, Ma C-M, Rogers DWO, Sánchez-Doblado F, Sempau J, Seuntjens J, Siebers JV. 2006. Phase-space database for external beam radiotherapy. IAEA (International Atomic Energy Agency) Technical Report INDC(NDS)-0484. Available from the website: http://www-nds.iaea.org/phsp/documents/indc-nds-0484.pdf.

Cirrone GAP, Cuttone G, Di Rosa F, Pandola L, Romano F, Zhang Q. 2010. Validation of the Geant4 electromagnetic photon cross-sections for elements and compounds. Nuclear Instruments and Methods in Physics Research Section A 618:315–322.

Cortés-Giraldo MA, Quesada JM, Gallardo MI. 2009a. Geant4 application for the simulation of the head of a Siemens Primus linac. AIP Conference Proceeding Series 1231:209–210.

Cortés-Giraldo MA, Quesada JM, Gallardo MI, Capote R. 2009b. Geant4 interface to work with IAEA phase-space files. Available from: http://www-nds.iaea.org/phsp/Geant4/G4IAEAphsp HowTo.pdf.

Faddegon BA, Asai M, Perl J, Ross C, Sempau J, Tinslay J, Salvat F. 2008a. Benchmarking of Monte Carlo simulation of bremsstrahlung from thick targets at radiotherapy energies. Medical Physics 35: 4308–4317.

Faddegon BA, Kawrakow I, Kubyshin Y, Perl J, Sempau J, Urban L. 2009. The accuracy of EGSnrc, Geant4 and PENELOPE Monte Carlo systems for the simulation of electron scatter in external beam radiotherapy. Physics in Medicine and Biology 54(20):6151–6163.

Faddegon BA, Perl J, Asai M. 2008b. Monte Carlo simulation of large electron fields. Physics in Medicine and Biology 53:1497–1510.

GAMOS project website: http://fismed.ciemat.es/GAMOS/.

IAEA phsp project website: http://www-nds.iaea.org/phsp.

Kawrakow I. 2000. Accurate condensed history Monte Carlo simulation of electron transport. I. EGSnrc, the new EGS4 version. Medical Physics 27:485–498.

Kawrakow I, Mainegra-Hing E, Rogers DWO, Tessier F, Walters BRB. 2009. The EGSnrc code system: Monte Carlo simulation of electron and photon transport. National Research Council of Canada Technical Report PIRS-701. Available from: http://www.irs.inms.nrc.ca/inms/irs/EGSnrc/EGSnrc.html.

Larsson S, Svensson R, Gudowska I, Ivanchenko V, Brahme A. 2005. Radiation transport calculations for 50 MV photon therapy beam using the Monte Carlo code GEANT4. Radiation Protection Dosimetry 115(1–4):503–507.

Lechner A, Ivanchenko VN, Knobloch J. 2010. Validation of recent Geant4 physics models for application in carbon ion therapy. Nuclear Instruments and Methods in Physics Research B 268:2343–2354.

Newhauser W, Koch N, Hummel S, Ziegler M, Titt U. 2005. Monte Carlo simulations of a nozzle for the treatment of ocular tumours with high-energy proton beams. Physics in Medicine and Biology 50:5229–5249.

Paganetti H, Jiang H, Lee S-Y, Kooy HM. 2004. Accurate Monte Carlo simulations for nozzle design, commissioning and quality assurance for a proton radiation therapy facility. Medical Physics 32:2107–2118.

Paganetti H, Jiang H, Parodi K, Slopsema R, Engelsman M. 2008. Clinical implementation of full Monte Carlo dose calculation in proton beam therapy. Physics in Medicine and Biology 53:4825–4853.

Poon E, Verhaegen F. 2005. Accuracy of the photon and electron physics in GEANT4 for radiotherapy applications. Medical Physics 32:1696–1711.

Rogers DWO. 2006. Fifty years of Monte Carlo simulations for medical physics. Physics in Medicine and Biology 51:R287–301.

Rogers DWO, Faddegon BA, Ding GX, Ma C-M, We J, Mackie TR. 1995. BEAM: A Monte Carlo code to simulate radiotherapy treatment units. Medical Physics 22:503–524.

ROOT data analysis framework website: http://root.cern.ch.

Salvat F, Fernández-Varea JM, Sempau J. 2008. PENELOPE-2008: A code system for Monte Carlo simulation of electron and photon transport. In: Workshop Proceedings. Barcelona, 30 June to 3 July 2008. Available from: http://www.nea.fr/html/science/pubs/2009/nea6416-penelope.pdf.

Walters BRB, Kawrakow I, Rogers DWO. 2002. History by history statistical estimators in the BEAM code system. Medical Physics 29:2745–2752.