

Rabat, Morocco

G4Linac_MT version 1.0

Published date : 03/09/17

G4Linac_MT version 1.0

User Manual

Written By : Dr. Jaafar EL Bakkali

Webpage : https://github.com/EL-Bakkali-Jaafar/G4Linac_MT

Table of Contents

1. Introduction	01
• About this document	01
• Quick Introduction to G4Linac	01
• Modules of G4Linac	02
2. Getting started	04
• Getting the source of G4Linac	04
• Prerequisite softwares and libraries	04
• Installing G4Linac	05
• Running an example	06
3. Troubleshooting.....	12
• Some common errors	12
• I need a help	12
4. For developer	12
5. license	12

1. Introduction

▪ About this Document

This document refers to G4Linac_MT version 1.0 and contains some instructions and tips which may help user to successfully building their Linac simulations using G4Linac_MT code which is built on the top of the last known stable version of Geant4 Monte Carlo code (at the time of writing this document) namely Geant4.10.03. Users don't need to have a deep knowledge of Geant4 Monte Carlo code nor to consult further documentation since this manual has been built to be self-consistent as well as possible. However, anyone want to contribute in developed of this code must have advanced knowledge of Geant4 code and are welcome.

Anyone have a question about this document, don't hesitate to send me an email at:
bahmedj@gmail.com

▪ Quick Introduction to G4Linac_MT

The word G4Linac_MT is an acronym of “ Geant4 For Linac with Multi Threading mode ”, as its name indicates, this code is dedicated to accurately modeling medical linear accelerator used in radiotherapy to treat cancer diseases by employing ionizing radiations. This code has been developed by Dr. Jaafar EL Bakkali, which is actually working as Assistant Professor in Nuclear Physics in Morocco. G4Linac_MT has been written in C++ language from scratch to perform all mandatory tasks required for validating the Monte Carlo Geant4 code for such Linac operating at photon or electron mode. The code supports the following features:

- The code was built on the top of the last known stable version of Geant4 (namely Geant4 .10.3).
- It uses multi-threading option allowing it to start multiple simulations at the same time and it merges data delivered by different threads at the end of a run.
- The code can accurately modeling the physics of all components of treatment head of a Linac.
- The code can accurately modeling the dose distribution either in homogeneous or either in heterogeneous water phantoms.
- The code can read material and geometry specification from .geom file (GEANT4 GEOMETRY FROM TEXT FILE version 1.0) or they can be specified directly by implementing a hard C++ code by the mean of Geant4 APIs.
- It uses a new format of PhaseSpace file based on HDF5 file format by the mean of the HDF5 C++ library and it integrates a set of h5PhaseSpace I/O routines for writing and reading H5Phase-Space files.
- It also stores the dosimetric data on a file with binary format, this for each thread. At the end of a run all dosimetric files are merged together in order to produce a single one.
- The code contains a tool that can be serve at extracting and visualizing data, two kind of files are supported namely: h5PhaseSpace and DosimetricData.

▪ **Modules of G4Linac_MT**

G4Linac_MT code consists of following three modules:

• ***G4Linac_Head***

This module lets user construct model geometry related to all components of a given Linac treatment head include:

- Invariant elements: primary collimator, output window, monitors chamber, mirror and mesh.
- Elements which strongly depend on the selected irradiation energy, namely X-ray target and flattening filter.
- Others that depend on the shape of the beam, namely removable jaws and multileaf system.

The construct of model geometry can be either set directly by implementing required C++ or can be read from a macro file that contains a Geant4 Geometry From Text code. G4Linac_Head module allows one to start its simulation in multi-threading mode, in each thread it will create a h5PhaseSpace file at the end of thread-run, after that all phase-space file will be merged in single one without user intervention, this at the end of a given Run. This code is the only which uses a new phase-space file format based on HDF5 file format and called h5PhaseSpace file. Regarding variance reduction technique, the module uses the technique called BREMSPE for photons bremsstrahlung splitting.

• ***G4Linac_DoseCal***

This module lets user simulate doses deposition in either homogeneous or heterogeneous water phantoms. The user can run multiple simulation at the same time, since this code supports multi-threading features. In aim to reduce RAM memory usage, all thread share some copy of h5PhaseSpace file which is used as an input to the Monte Carlo dose calculations. At the end of execution of each run, this module merges DosimetricData per thread in binary file which can be analyzed by G4Linac_DataAnalysis module.

• ***G4Linac_DataAnalysis***

This module can be used for data analysis, the data can be either h5PhaseSpace data or either DosimetricData.

For h5PhaseSpace data, this tool can be used for histogramming height physics quantities:

- *Visualizing photon energy spectrum*

/G4Linac_DataAnalysis PhaseSpaceData PhaseSpaceFileName.h5 photon_energy_spectrum

root photon_energy_spectrum.C

- *Visualizing electron energy spectrum*

*/G4Linac_DataAnalysis PhaseSpaceData PhaseSpaceFileName.h5 electron_energy_spectrum
root electron_energy_spectrum.C*

- *Visualizing photon angular distribution*

*/G4Linac_DataAnalysis PhaseSpaceData PhaseSpaceFileName.h5 photon_angular_distribution
root photon_angular_distribution.C*

- *Visualizing electron angular distribution*

*/G4Linac_DataAnalysis PhaseSpaceData PhaseSpaceFileName.h5 electron_angular_distribution
root electron_angular_distribution.C*

- *Visualizing photon spatial distribution in plane XY*

*/G4Linac_DataAnalysis PhaseSpaceData Phase_Space_fileName.h5 photon_spatial_distribution_xy
root photon_spatial_distribution_xy.C*

- *Visualizing particle spatial distribution in plane XY*

*/G4Linac_DataAnalysis PhaseSpaceData Phase_Space_fileName.h5 spatial_distribution_xy
root spatial_distribution_xy.C*

- *Visualizing photon spatial distribution in plane XY in function of its kinetic energies*

*/G4Linac_DataAnalysis PhaseSpaceData Phase_Space_fileName.h5
photon_spatial_distribution_xy_with_kinetic
root photon_spatial_distribution_xy_with_kinetic.C*

- *Visualizing electron spatial distribution in plane XY in function of its kinetic energies*

/G4Linac_DataAnalysis PhaseSpaceData Phase_Space_fileName.h5 electron_

spatial_distribution_xy_with_kinetic
root electron_spatial_distribution_xy_with_kinetic.C

For DosimetricData, this tool can be used for histogramming four physics quantities:

- *Visualizing percentage depth dose curve*

/G4Linac_DataAnalysis DosimetricData DosimetricData_fileName.dat pdd x_center_id y_center_id
root pdd.C

- *Visualizing percentage x profile*

/G4Linac_DataAnalysis DosimetricData DosimetricData_fileName.dat xprofile y_center_id z_center_id
root xprofile.C

- *Visualizing percentage y profile*

/G4Linac_DataAnalysis DosimetricData DosimetricData_fileName.dat yprofile x_center_id z_center_id
root yprofile.C

- *Visualizing dose heat map*

/G4Linac_DataAnalysis DosimetricData DosimetricData_fileName.dat dose_heatmap x_center_id
z_center_id
root dose_heatmap.C

2. Getting started

▪ Getting the source of G4Linac_MT

G4Linac_MT is hosted on github and it can be obtained freely from the following link:

https://github.com/EL-Bakkali-Jaafar/G4Linac_MT

by clicking on “clone or download” button.

▪ Obtaining, compiling and installing prerequisite softwares and libraries

G4Linac_MT was built on the top of Geant4.10.03, so users must have it installed in their operating system. Geant4.10.03 can be downloaded from the following link:

http://geant4.web.cern.ch/geant4/support/source/geant4_10_03_p02.zip

Users must also download all require data and must follow Geant4 installation guide.

G4Linac also needs ROOT data analysis framework (version 6.10) for histogramming, which can be downloaded from:

https://root.cern.ch/download/root_v6.10.04.source.tar.gz

This tool must be configured, compiled and finally installed by following ROOT installation guide.

G4Linac uses HDF5 C++ API for writing and reading h5PhaseSpace files that are based on HDF5 file format. The last version of hdf5 can be downloaded from this link:

<https://support.hdfgroup.org/ftp/HDF5/current/src/hdf5-1.10.1.tar>

Then, user must configure this tool as follows:

```
./configure --enable-cxx --enable-threadsaf --prefix=/opt/hdf5 --enable-unsupported
```

after that this tool must be compiled (by typing make) and installed (by typing make install).

▪ Installing G4Linac_MT

Installing G4Linac_MT is an easy task, after setup-ping all prerequisite libraries and after downloading the source of G4Linac_MT, users need to extracting the file, by untaring it with any known untar tool, after that users must see three folders namely: G4Linac_Head, G4Linac_DoseCal and G4Linac_DataAnalysis. Users must create a build folder for each ones change directory to the new folder and execute the cmake command. For example:

1. Installing G4Linac_Head module

```
mkdir G4Linac_Head
_ Build
cd ./G4Linac_Head

cmake -DGeant4_DIR=/home/user/Geant4/geant4.10.3-install ~/Geant4/WorkDir/G4Linac_Head

make
```

2. Installing G4Linac_DoseCal module

```
mkdir G4Linac_DoseCal_Build
cd ./G4Linac_DoseCal
cmake -DGeant4_DIR=/home/user/Geant4/geant4.10.3-install ~/Geant4/WorkDir/G4Linac_DoseCal
make
```

3. Installing G4Linac_Head module

```
mkdir G4Linac_DataAnalysis_Build
cd ./G4Linac_DataAnalysis
cmake -DGeant4_DIR=/home/user/Geant4/geant4.10.3-install
~/Geant4/WorkDir/G4Linac_DataAnalysis
make
```

▪ Running an example

• Sample Linac geometry implemented using a hard C++ code

The G4Linac_MT code comes as default with a sample Linac geometry implemented directly by using C++ programming code and Geant4 APIs. It's about a sample linac operating at 6 MV configuring a $10 \times 10 \text{ cm}^2$ radiation field and assembled by the following components: x-ray target, primary collimator, flattening filter, ionization chamber and jaws.

G4Linac_MT code comes with three separated folder namely: G4Linac_Head, G4Linac_DoseCal and G4Linac_DataAnalysis. Users must performing three mandatory tasks:

- First task: modeling medical linear accelerator head operating at photon mode X6

create new folder in same directory of G4Linac_Head, rename it to “G4Linac_Head_build”, then open new Linux terminal, cd to “G4Linac_Head_build” folder and execute the following command:

```
cmake -DGeant4_DIR=/home/user/Geant4/geant4.10.3-install ../G4Linac_Head
```

Then type:

```
make
```

and run the program by typing:

```
./G4Linac_Head run.mac
```

User can set number of events per thread by changing the following line in “run.mac” file:

```
/Parameters/NumberOfEventsPerThread _numberOfEventsPerThread
```

At the end of a run, the simulation output is composed by the two following files:

PhaseSpace.h5 and PhaseSpace.summary.

The content of PhaseSpace.summary is as follows:


```
#####
G4LINAC_MT v1.0: A GEANT4 BASED APPLICATION FOR MEDICAL LINEAR ACCELERATOR
```

DEVELOPED BY DR. JAAFAR EL BAKKALI, BAHMEDJ@GMAIL.COM, TETOUAN-MOROCCO, 08/ 2017

MULTI_THREADING SUPPORT: YES

```
#####
└ @DATE_OF_CREATION: Sun Sep 3 18:20:46 2017
```

└ @ELAPSED_TIME: 357.741 seconds.

└ @PHASE_SPACE_NAME:

└ @Z_STOP: 1000 mm.

└ @REDUCTION VARIANCE TECHNIQUE: BREMSPE

└ @SPLIT NUMBER: 500

└ @TOTAL_NUMBER_OF_SIMULATED_HISTORIES: 18000

└ @TOTAL_NUMBER_OF_ACTIVE_EVENTS: 8285

└ @ACTIVE_EVENTS_PERCENT: 46.0278 %

└ @TOTAL_NUMBER_OF_PARTICLES : 96036

└ @TOTAL_NUMBER_OF_PARTICLES_PER_SECONDE : 268

└ @TOTAL_NUMBER_OF_PHOTONS: 95929

└ @TOTAL_NUMBER_OF_ELECTRONS: 101

└ @TOTAL_NUMBER_OF_POSITRONS: 6

└ @PHOTONS_PERCENT: 99.8886 %

└ @ELECTRONS_PERCENT: 0.105169 %

└ @POSITRONS_PERCENT: 0.00624766 %

└ @PHOTONS_MAX_ENERGY: 6.1102 MeV

└ @PHOTONS_MIN_ENERGY: 0.910908 MeV

└ @PHOTONS_MEAN_ENERGY: 1.60081 MeV

└ @ELECTRONS_MAX_ENERGY: 5.04415 MeV

└ @ELECTRONS_MIN_ENERGY: 1.94522 MeV

└ @ELECTRONS_MEAN_ENERGY: 1.61613 MeV

└ @POSITRONS_MAX_ENERGY: 4.56933 MeV

└ @POSITRONS_MIN_ENERGY: 2.11287 MeV

└ @POSITRONS_MEAN_ENERGY: 2.61498 MeV

└ @PHOTONS_MAX_WEIGHT: 0.002

└ @PHOTONS_MIN_WEIGHT: 0.002

└ @PHOTONS_MEAN_WEIGHT: 0.002

Here a phase-space file created of 3.5 Mb that contains 96036 particles.

second task: modeling dose distribution in a homogeneous water phantom

Create a new folder in the same directory of G4Linac_DoseCal, rename it to “G4Linac_DoseCal_build”, then open new Linux terminal, cd to “G4Linac_Head_build” folder and execute the following command:

```
cmake -DGeant4_DIR=/home/user/Geant4/geant4.10.3-install ../G4Linac_DoseCal
```

Then type:

```
make
```

The h5Phase-Space file created previously need to be copied into this directory. Then open run.mac file and change Phase_Space word to the name of h5PhaseSpace file as follows:

```
/Parameters/H5PhaseSpaceName Phase_Space
```

After that, users should run this module by typing:

```
./G4Linac_DoseCal run.mac
```

At the end of the simulation, a binary file named dose.dat will be created which contains dosimetric data and can be analyzed by G4Linac_DataAnalysis module.

Third task: histograming data

Users should create a new folder in the same directory of G4Linac_DoseCal, rename it to “G4Linac_DoseCal_build”, then open new Linux terminal, cd to “G4Linac_Head_build” folder and execute the following command:

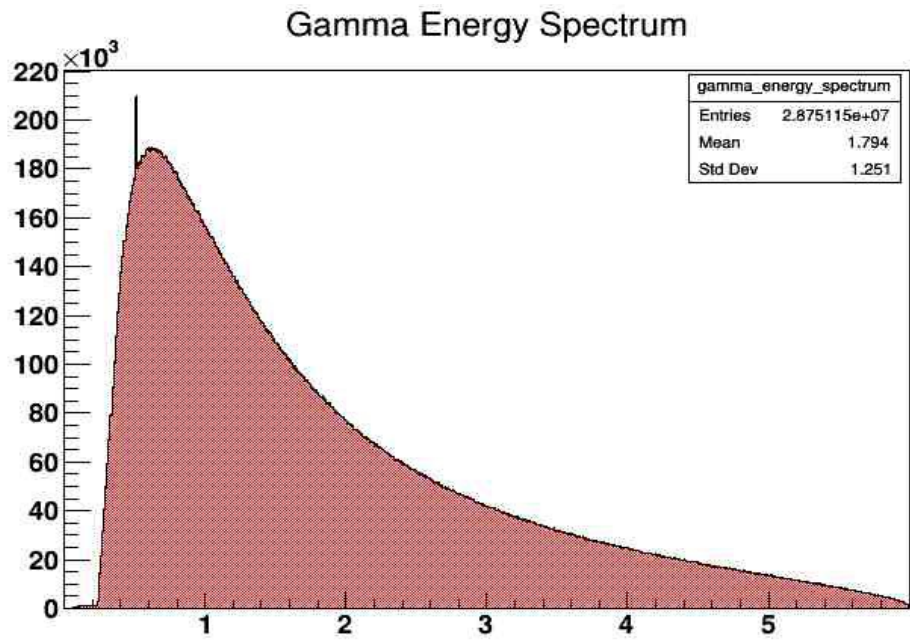
```
cmake -DGeant4_DIR=/home/user/Geant4/geant4.10.3-install ../G4Linac_DataAnalysis
```

Then type:

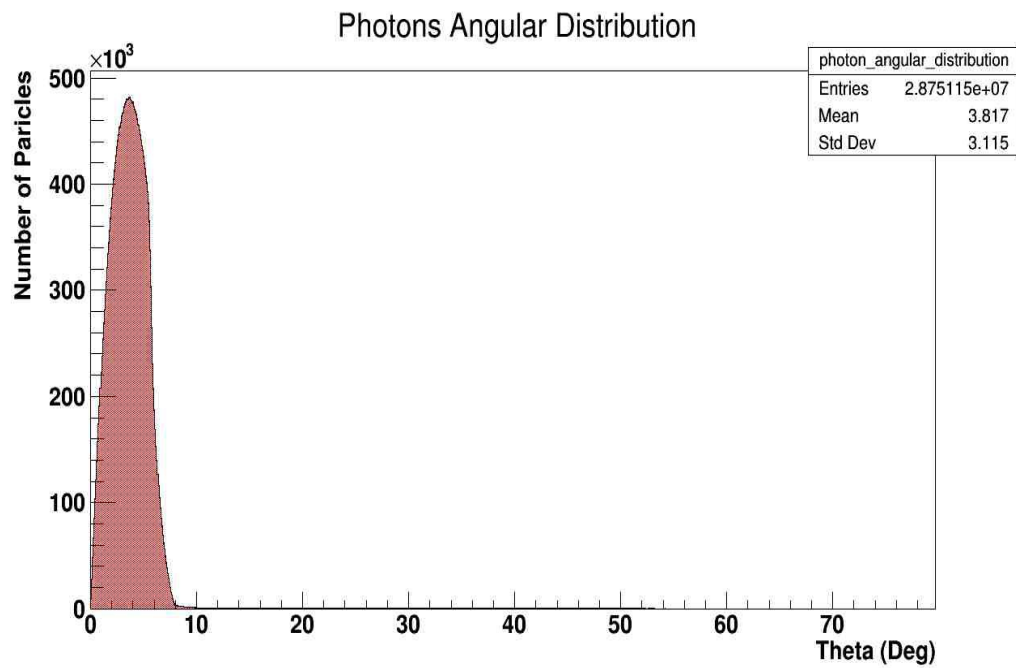
```
make
```

After that, users need to copy h5Phase-Space file and dose.dat to this emplacement in order to perform simulation analysis task. Here we can see a set of histogramms produced for a number of physical quantities thereby extracting data from h5Phase-Space file.

```
/G4Linac_DataAnalysis PhaseSpaceData PhaseSpace.h5 photon_energy_spectrum; root  
photon_energy_spectrum.C
```

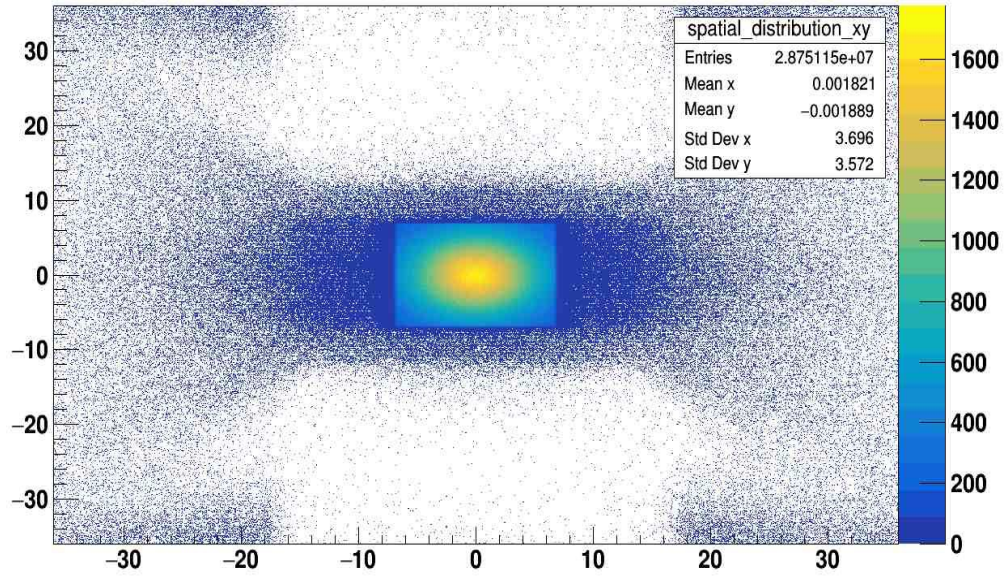


G4Linac_DataAnalysis PhaseSpaceData PhaseSpace.h5 photon_angular_distribution; root photon_angular_distribution.C



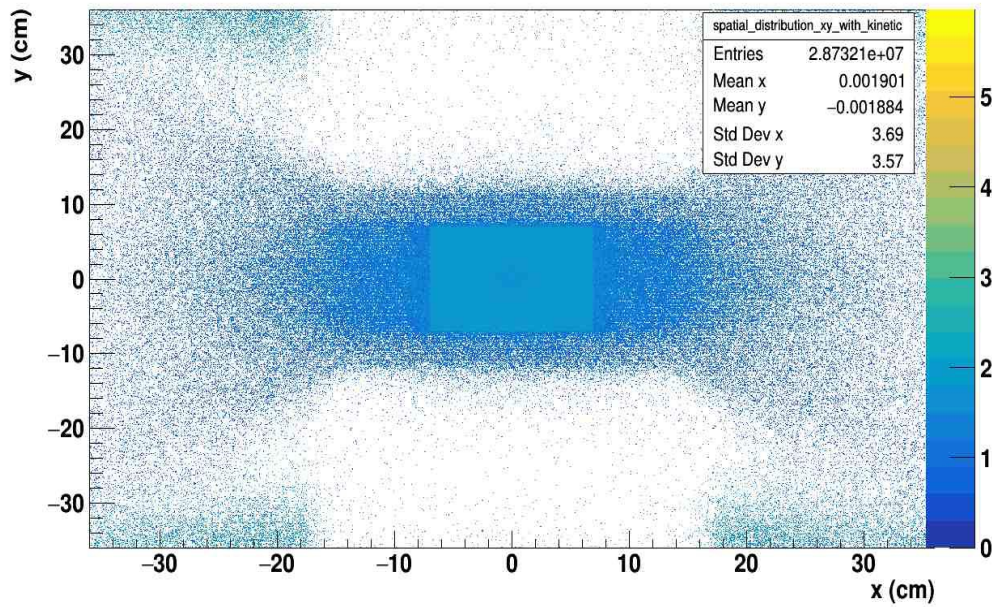
*./G4Linac_DataAnalysis PhaseSpaceData Phase_Space.h5 spatial_distribution_xy ;root
spatial_distribution_xy.C*

Spatial Distribution XY



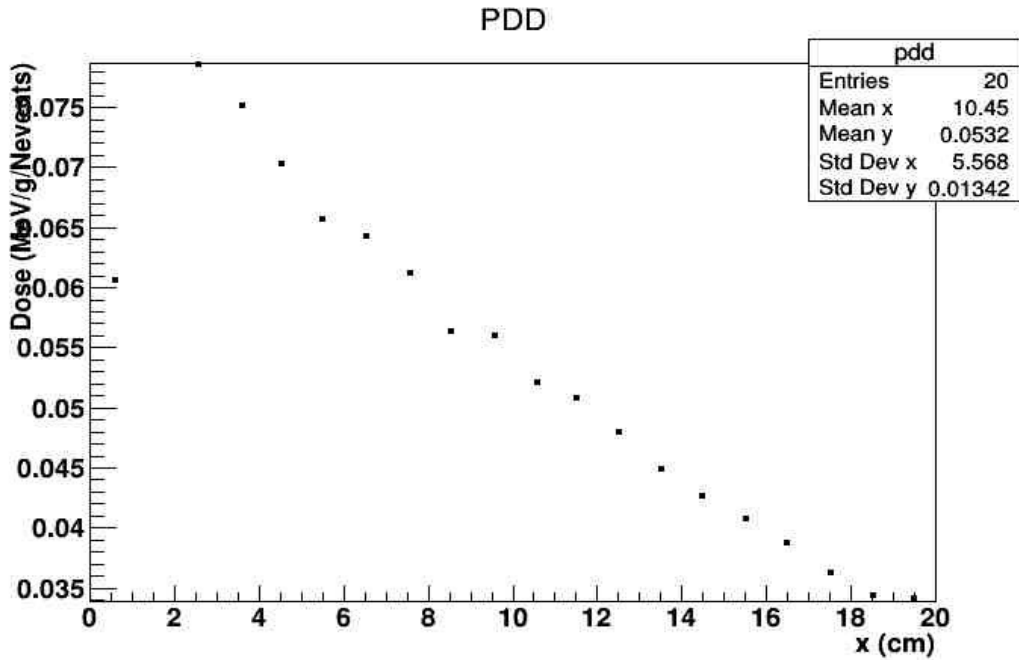
*./G4Linac_DataAnalysis PhaseSpaceData Phase_Space.h5 photon_spatial_distribution_xy_with_kinetic
root photon_spatial_distribution_xy_with_kinetic.C*

Spatial distribution of photons in function of its kinetic energies

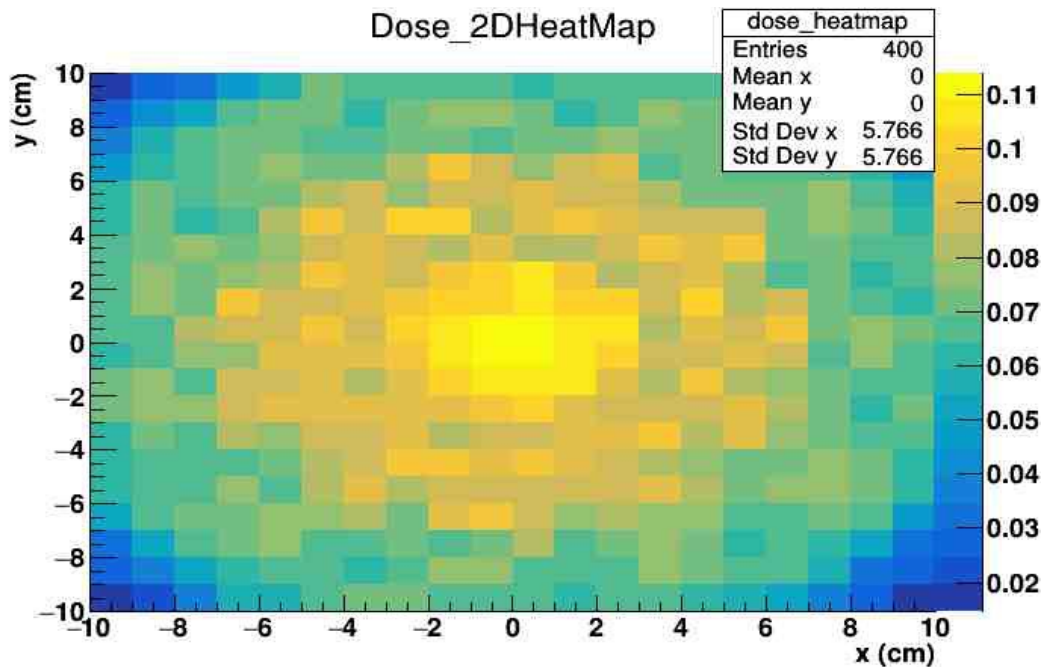


Here we can see a set of histogramms produced for a number of physical quantities thereby extracting data from DosimetricData file called dose.dat.

```
./G4Linac_DataAnalysis DosimetricData dose.dat pdd 4 2
; root pdd.C
```



```
./G4Linac_DataAnalysis DosimetricData dose.dat dose_heatmap 3; root dose_heatmap.C
```



3. Troubleshooting

- **Some common errors**

In order to determine the line of C++ code that causes a segmentation fault you should run the following command:

valgrind -tool=helgrind nameofModule run.mac

where nameofModule should be G4Linac_Head or G4Linac_DoseCal .

- **I need a help**

For any questions, users should contact me at bahmedj@gmail.com

4. For developers

Users who want to contribute in development of this code are welcome !

5. License

G4Linac_MT is an open-source code distributed under GPL liscence.