Rabat, Morocco

Published date : 20/10/2017

G4Linac_MT version 1.0

User Manual

Version1.0 Revision 2.0

Written By : Dr. Jaafar EL Bakkali

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

Table of Contents

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 documentations since this manual has been built to be self-consistent as well as possible. However, anyone who want to contribute in developed of this code should 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 "**G**eant4 Application **For** Modelling **Linac** with support of **M**ulti **T**hreading 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 of 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 Monte Carlo simulations at the same time and finally it merges output 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 given Linac.
- The code can accurately modeling the dose distribution either in homogeneous or either in heterogeneous water phantoms.
- The code uses script language as the material and geometry specification are read from .geom file (GEANT4 GEOMETRY FROM TEXT FILE version 1.0).

- It uses a new format of PhaseSpace file based on HDF5 file format by the mean of the HDF5 C++ library that I was developped in earlier years. For more information please consult this paper " Investigation of the HDF5 C++ Library in Development of

New Phase-Space I/O for Radiotherapy Simulation Using Monte Carlo Geant4 Code".

- In order to avoiding unnecessary time spent by writing data to the physical disk, those data are stored in RAM memory and finally are dumped to the physical disk at the end of a Run.
- 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 a Geant4 model of a typical Linac treatment head geometry that includes:

- 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 multi-leaf system.

The construct of a model geometry is simply taken place by reading Geant4 Geometry From Text code. G4Linac_Head module allows one to start its Monte Carlo simulation in multi-threading mode. Indeed, in each thread it will create an h5PhaseSpace file at the end of a thread-run, after that all H5PhaseSpace files will be finally merged in a single one, thus without user intervention and is done at the end of a given Run. It should be noted here, that this code is the only one which uses a new phase-space file format based on HDF5 file format and so called h5PhaseSpace file. Regarding variance reduction technique, the module uses the technique called BREMSPE for photons

bremsstrahlung splitting.

- *G4Linac_DoseCal*

This module lets user perform Monte Carlo simulation of 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 whi are produced produced by each thread in a binary file which can be furherly analyzed by G4Linac_DataAnalysis module.

- *G4Linac_DataAnalysis*

This module can be used for data analysis and histogramming tasks. The data can be either h5PhaseSpace data or either DosimetricData. For h5PhaseSpace data, this tool can be used for histogramming ten physics quantities:

Visualizing photon energy spectrum

```
./G4Linac_DataAnalysis PhaseSpaceData PhaseSpaceFileName.h5 photon_energy_spectrum
```

Visualizing electron energy spectrum

```
./G4Linac_DataAnalysis PhaseSpaceData PhaseSpaceFileName.h5 electron_energy_spectrum
```

Visualizing photon angular distribution

```
./G4Linac_DataAnalysis PhaseSpaceData PhaseSpaceFileName.h5 photon_angular_distribution
```

Visualizing electron angular distribution

```
./G4Linac_DataAnalysis PhaseSpaceData PhaseSpaceFileName.h5 electron_angular_distribution
```

Visualizing photon spatial distribution in plane XY

```
./G4Linac_DataAnalysis PhaseSpaceData  Phase_Space_fileName.h5
photon_spatial_distribution_xy
```

Visualizing electron spatial distribution in plane XY

```
./G4Linac_DataAnalysis PhaseSpaceData  Phase_Space_fileName.h5
photon_spatial_distribution_xy
```

Visualizing particle spatial distribution in plane XY

```
./G4Linac_DataAnalysis PhaseSpaceData  Phase_Space_fileName.h5 spatial_distribution_xy
```

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
```

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
```

Visualizing particle spatial distribution in plane XY in function of its kinetic energies

```
./G4Linac_DataAnalysis PhaseSpaceData  Phase_Space_fileName.h5
spatial_distribution_xy_with_kinetic
```

For DosimetricData, this tool can be used for performing histogramming tasks of six physics quantities:

Visualizing percentage depth dose curve

```
./G4Linac_DataAnalysis DosimetricData  DosimetricData_fileName.dat
Percentage_Depth_Dose_Curve  x_voxel_id  y_voxel_id
```

Visualizing Cross-plane Dose Profile

```
./G4Linac_DataAnalysis DosimetricData  DosimetricData_fileName.dat  Cross-plane_Dose_Profile
y_voxel_id  z_voxel_id
```

Visualizing In-plane Dose Profile

```
./G4Linac_DataAnalysis DosimetricData  DosimetricData_fileName.dat  In-plane_Dose_Profile
x_voxel_id  z_voxel_id
```

Visualizing Dose Heat Map in plane YZ

```
./G4Linac_DataAnalysis DosimetricData  DosimetricData_fileName.dat  2D_Dose_HeatMap  x
x_voxel_id
```

Visualizing Dose Heat Map in plane XZ

```
./G4Linac_DataAnalysis DosimetricData  DosimetricData_fileName.dat 2D_Dose_HeatMap  y
y_voxel_id
```

Visualizing Dose Heat Map in plane XY

```
./G4Linac_DataAnalysis DosimetricData  DosimetricData_fileName.dat 2D_Dose_HeatMap z
z_voxel_id
```

This tool also allows one to draw Relative Standard Deviation curves for three dosimetric

quantities:

Visualizing rsd curve associated to percentage depth dose

```
./G4Linac_DataAnalysis DosimetricData DosimetricData_fileName.dat pdd_statisticals
x_voxel_id y_voxel_id
```

Visualizing rsd curve associated to Cross-plane Dose Profile

```
./G4Linac_DataAnalysis DosimetricData DosimetricData_fileName.dat Cross-
plane_Dose_Profile_Rsd y_voxel_id z_voxel_id
```

Visualizing rsd curve associated to In-plane Dose Profile

```
./G4Linac_DataAnalysis DosimetricData DosimetricData_fileName.dat In-plane_Dose_Profile_Rsd
x_voxel_id z_voxel_id
```

- ***H5Phase-Space data* format**

h5PhaseSpace is a new PhaseSpace file format that stores all particle data at a given plane. I have developed this PhaseSpace file format since 2015 where its first version can be worked only in a sequential mode. Todays, a new version has been developed that exploits the multi-threading capabilities of Geant4 Monte Carlo code. The h5PhaseSpace data consists of two distinct file formats:

- h5PhaseSpace_name.h5: an hdf5 file which keeps all information about recorded particles that include: the number of simulated histories, particle kinetic energy, particle statistical weight, particle PDGE code, particle position, and particle momentum direction. It stores also Monte Carlo simulation information include number of active events, event unique ID of a number of particles created at the same Geant4 event unit, number of these particles, number of original histories and the z-coordinate of h5PhaseSpace plane.

- h5PhaseSpace_name.summary: an ASCII text file generated at the end of a Geant4 Run unit which has an extension of ".summary". It stores meaningful statistical data about a given Monte Carlo simulation shuch as number of simulated histories, percent of active events, max-mean-min energy and weight of all recorded particles, percent of each of them, CPU time spent by a Monte Carlo simulation and date of creation of an h5PhaseSpace file. It should be noted here, that all of these data are self-explanatory.

- *DosimericData data* format

DosimericData is a binary file that keeps all information about dosimetric and phantom data, such as number of voxels along x axis, number of voxels along y axis, number of voxels along z axis, voxalized phantom dimensions along x, y and z axis, number of events recorded in each voxel, absorbed dose and associated relative standard deviation in a given voxel. At the end of a given Geant4 Run unit, these useful information are dumped to an ASCII text file which can be furthmore readable by users. This data will be used as input parameters for G4Linac_DataAnalysis module in order to perform histogramming tasks of dosimeric functions such as 2d dose heat map along a given axis which is a common visualizing method for plotting data in a graphical manner.

2. Getting started

Getting the source of G4Linac_MT

G4Linac_MT it hosted on github and it can be obtained freely from the following link: https://github.com/EL-Bakkali-Jaafar/G4Linac_MT

This by clicking on "clone or download" button.
G4Linac_MT has been tested in Ubuntu 14.05 and Ubuntu 16 but it should work with other Linux distributions as well. G4Linac_MT has been tested on a Lenovo Workstation having the following characheristics:Intel ® Xenon(R) CPU W3690 @ 3.47 GHZ x 12, OS TYPE : 64 bit Ubuntu 16.04 LTS.

Prerequisite softwares and libraries

G4Linac_MT was built on the top of Geant4.10.03, so users must have installed its in their operating system. Geant4.10.03 can be downloaded freely from the following link: http://geant4.web.cern.ch/geant4/support/source/geant4_10_03_p02.zip

Users should also download all required data and should follow Geant4 installation guide. G4Linac also needs ROOT data analysis framework (version 6.10) for performing histogramming task, which can be downloaded from the following link: 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 lasted version of hdf5 tool can be downloaded from this link:

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

Then, users need to configure this tool as follows:

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

After that this tool should 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, this by untaring it with any known untar tool. After that users probably must see three separated 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
```

3. Running simple examples

▪ Dose simulation in water phantom irradiated by a Mono-Field

The G4Linac_MT code comes by default with a sample Linac geometry implemented in macro file called "LinacX6.geom". It's about a sample Linac operating at 6 MV configuring a 10 x 10 cm^2 radiation field and assembled by the following components: x-ray target, primary collimator, flattening filter, ionization chamber and jaws. Structurally, G4Linac_MT code comes with three separated folders namely: G4Linac_Head, G4Linac_DoseCal and G4Linac_DataAnalysis. Users should performing three mandatory tasks in order to validate G4Linac_MT code for a such Linac:

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

create a temporary directory beside G4Linac_Head folder, which we denote as "G4Linac_Head_build", then open a 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
make
```

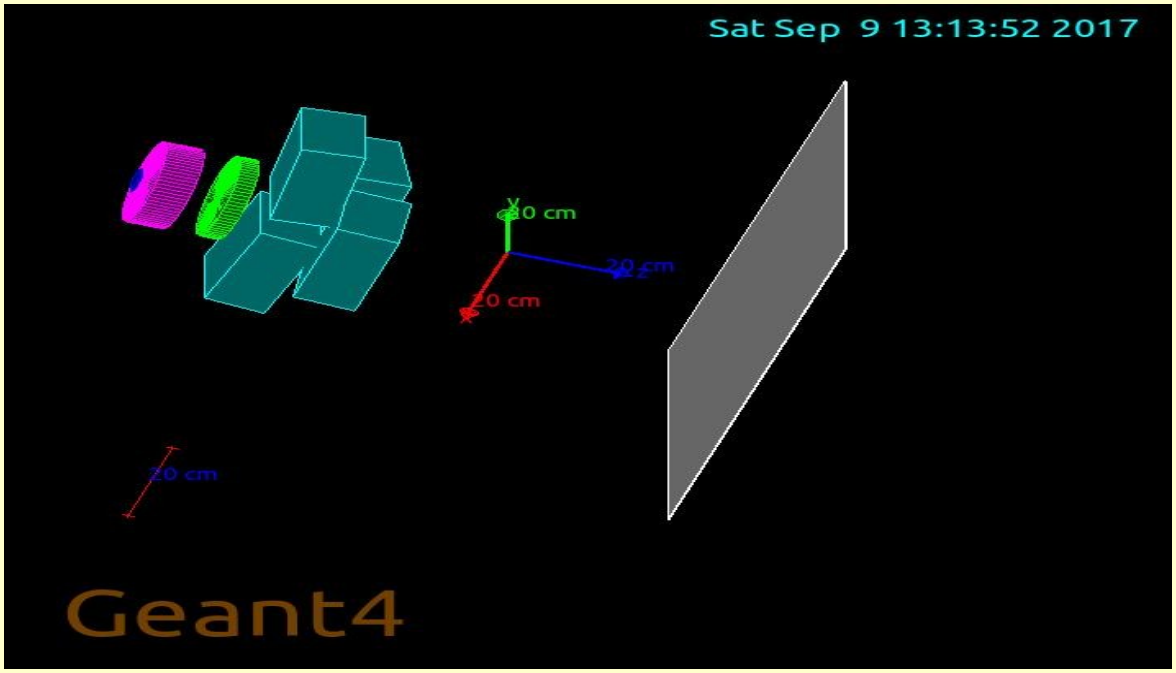
and run the program by typing:

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

Users can display Linac geometry by typing the following line:

```
./G4Linac_Head
```

The results should be like this:



User can set the number of events per thread, this by only changing the following line in "run.mac" file:

```
/Parameters/NumberOfEventsPerThread  _numberOfEventsPerThread
```

At the end of a Geant4 Run unit, the Monte Carlo simulation output is composed by the two files PhaseSpace.h5 and PhaseSpace.summary. The content of PhaseSpace.summary should be like this:

```
#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#
G4LINAC_MT version 1.O: a Geant4-based application for Medical Linear Accelerator
Developed by Dr.Jaafar EL Bakkali, Assistant Prof. of Nuclear Physics, Rabat, Morocco,  10/09/ 2017
Webpage :https://github.com/EL-Bakkali-Jaafar/G4Linac_MT
#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#
   @DATE_OF_CREATION: Sun Oct  1 17:06:54 2017
   @ELAPSED_TIME: 8067.72 seconds.
   @PHASE_SPACE_NAME: PhaseSpaceX6.h5
   @Z_STOP: 500 mm.
   @REDUCTION VARIANCE TECHNIQUE: BREMSPE
   @SPLIT NUMBER: 100
   @TOTAL_NUMBER_OF_SIMULATED_HISTORIES: 9600000
   @TOTAL_NUMBER_OF_ACTIVE_EVENTS: 1024713
   @ACTIVE_EVENTS_PERCENT: 10.6741 %
   @TOTAL_NUMBER_OF_PARTICLES : 2419220
   @TOTAL_NUMBER_OF_PARTICLES_PER_SECONDE : 299
   @TOTAL_NUMBER_OF_PHOTONS: 2412231
```

```
@TOTAL_NUMBER_OF_ELECTRONS: 6851
@TOTAL_NUMBER_OF_POSITRONS: 138
@PHOTONS_PERCENT: 99.7111 %
@ELECTRONS_PERCENT: 0.28319 %
@POSITRONS_PERCENT: 0.00570432
@PHOTONS_MAX_ENERGY: 6.613 MeV
@PHOTONS_MIN_ENERGY: 0.0501649 MeV
@PHOTONS_MEAN_ENERGY: 1.44539 MeV
@ELECTRONS_MAX_ENERGY: 5.80573 MeV
@ELECTRONS_MIN_ENERGY: 0.012435 MeV
@ELECTRONS_MEAN_ENERGY: 1.20346 MeV
@POSITRONS_MAX_ENERGY: 4.33367 MeV
@POSITRONS_MIN_ENERGY: 0.0292144 MeV
@POSITRONS_MEAN_ENERGY: 1.73146 MeV
@PHOTONS_MAX_WEIGHT: 1
@PHOTONS_MIN_WEIGHT: 0.01
@PHOTONS_MEAN_WEIGHT: 0.0100283
@ELECTRONS_MAX_WEIGHT: 0.01
@ELECTRONS_MIN_WEIGHT: 0.01
@ELECTRONS_MEAN_WEIGHT: 0.01
@POSITRONS_MAX_WEIGHT: 0.01
@POSITRONS_MIN_WEIGHT: 0.01
@POSITRONS_MEAN_WEIGHT: 0.01
```

Here a H5PhaseSpace file having a size of 182.4 MB and storing 2419220 different 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
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 a Monte Carlo simulation, a binary file named “dose.dat” will be created which contains dosimetric data and can be analyzed by G4Linac_DataAnalysis module. These Data are also dumped to an ASCII text file called “dose.txt” which can be then opened by any known text editor. Its content should be like this:

```
#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#
G4LINAC_MT version 1.O: a Geant4-based application for Medical Linear Accelerator
Developed by Dr.Jaafar EL Bakkali, Assistant Prof. of Nuclear Physics,
Rabat, Morocco,  10/09/ 2017
Webpage :https://github.com/EL-Bakkali-Jaafar/G4Linac_MT
#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#=#
@TOTAL NUMBER OF SIMULATED EVENTS: 144000000
@THREAD NUMBERS: 12
@REDUCTION VARIANCE TECHNIQUE: PARTICLE RECYCLING
@SPLITTING/RECYCLING FACTOR:10
@NUMBER_VOXELS_ALONG_X: 39
@NUMBER_VOXELS_ALONG_Y: 39
@NUMBER_VOXELS_ALONG_Z: 40
@PHANTOM_SIZE_X: 19.5 cm
@PHANTOM_SIZE_Y: 19.5 cm
@PHANTOM_SIZE_Z: 20 cm
@VOXEL_SIZE_X: 0.5 cm
@VOXEL_SIZE_Y: 0.5 cm
@VOXEL_SIZE_Z: 0.5 cm
@VOXEL_VOLUME: 0.125 cm_3
@GLOBAL_RSD (%): 0.619279
@DOSE UNIT IS: (MEV/GRAMME/NUMBER_OF_SIMULATED_EVENTS)
@ACTIVE EVENTS TO TOTAL HISTORIES IN H5_PHASE_SPACE RATIO: 0.0803563
@ELAPSED TIME IN SECONDS: 66457.1
@ESTIMATED CPU TIME FOR ACHIEVING GLOBAL_RSD ~1%: [ 25486.7 SECONDS <=> 7.07964
HOURS ].
@ESTIMATED NUMBER OF REQUIRED EVENTS FOR ACHIEVING GLOBAL_RSD ~1%:  55224907
PARTICLES.
@ESTIMATED CPU TIME FOR ACHIEVING MAX_RSD ~1%: [ 302284 SECONDS <=> 83.9679
HOURS ].
-----------------------------------------------------------------------------------------
Voxel_X_ID    Voxel_Y_ID    Voxel_Z_ID        DOSE         RSD(%)    NEVENTS_IN_VOXEL
   0               0              0        6.76182e-07    1.61529       15930
   0               0              1        3.6871e-07     2.06948        8980
   0               0              2        2.15069e-07    1.92214        8821
   0               0              3        2.30996e-07    1.83568        9528
   0               0              4        2.27958e-07    1.79731        9925
```

```
0                    0                    5            2.55189e-07    1.72732        10448
……..
```

As any one can clearly see, the code contains an useful method for estimating CPU time required for achieving a global relative standard deviation less than one per cent.

- **Third task: data histogramming**

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:

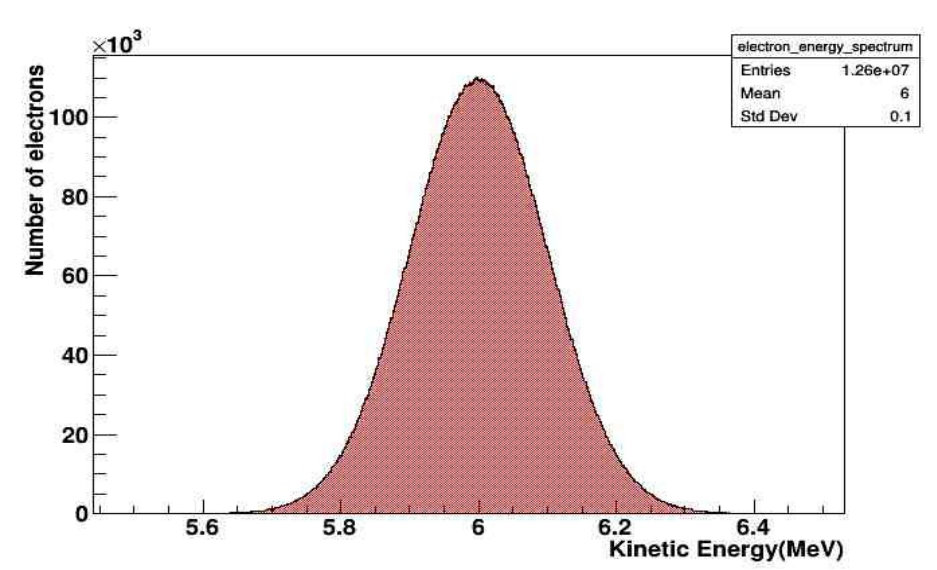
```
cmake -DGeant4_DIR=/home/user/Geant4/geant4.10.3-install ../G4Linac_DataAnalysis
make
```

After that, users need to copie "h5Phase-Space.h5" file and "dose.dat" to this emplacement in order to perform analysis task. Here we share a number of histogramms produced for a number of physical quantities thereby extracting data from h5Phase-Space file:

- **Energy spectrum of 6 MeV electron source**

In order to capture particles just leaving the electron sourece, the h5PhaseSpace Zstop parameter must be set to -60 cm. Then users should run G4Linac_Head module in order to produce an "h5PhaseSpace.h5" file which need to be moved to build directory of G4Linac_DataAnalysis. The following line must be executed in order to obtain the desired histogramm:

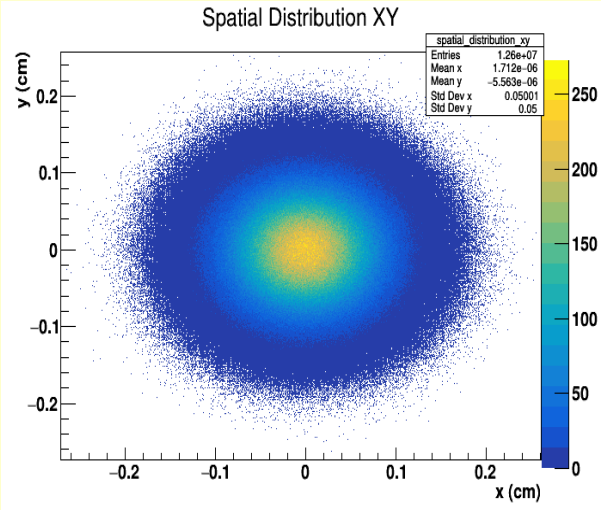
```
./G4Linac_DataAnalysis PhaseSpaceData PhaseSpace.h5 electron_energy_spectrum
```



Users should obtain picture like below.

- **Spatial bidistribution of 6 MeV electron source**

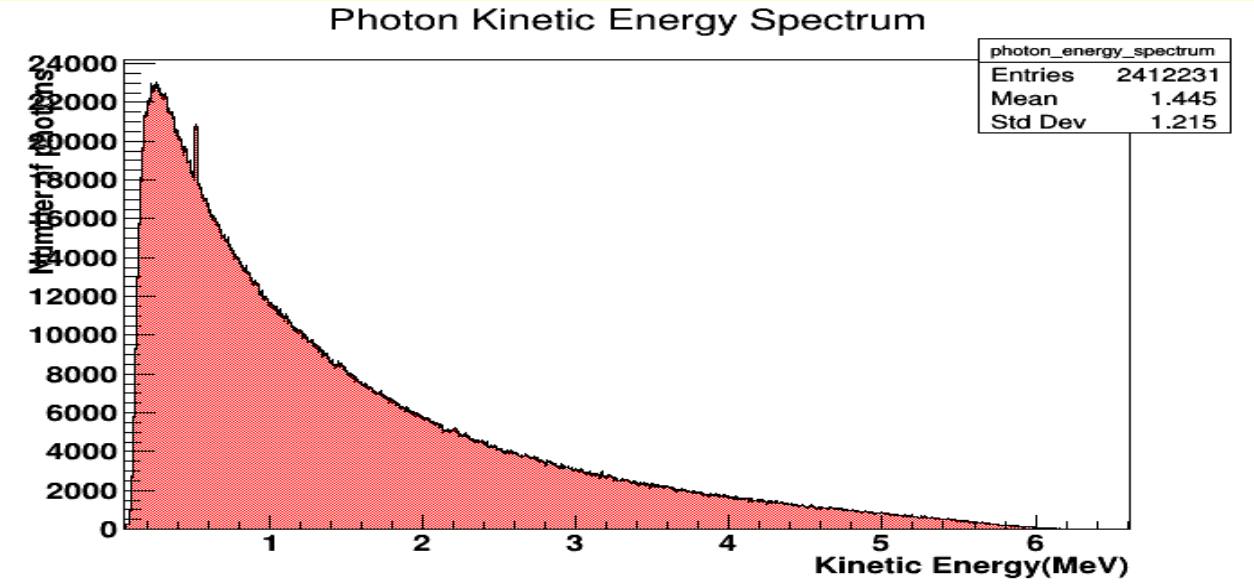
```
./G4Linac_DataAnalysis PhaseSpaceData PhaseSpace.h5 electron_spatial_distribution_xy
```



- **6 MV**

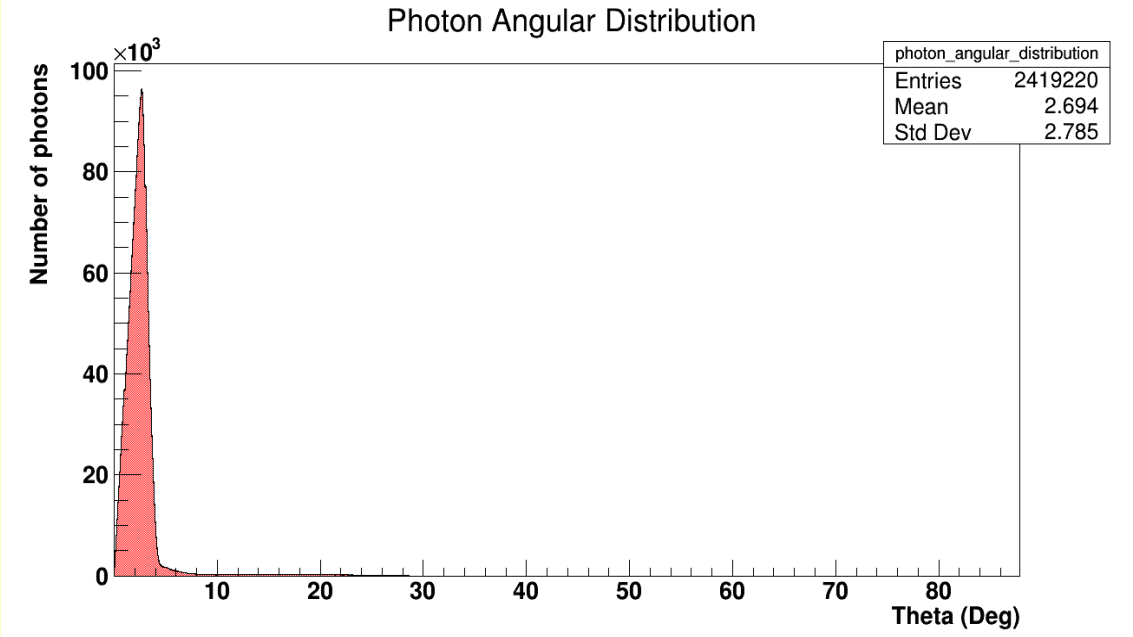
Bremsstrahlung photon energy spectrum

```
./G4Linac_DataAnalysis PhaseSpaceData PhaseSpace.h5 photon_energy_spectrum
```



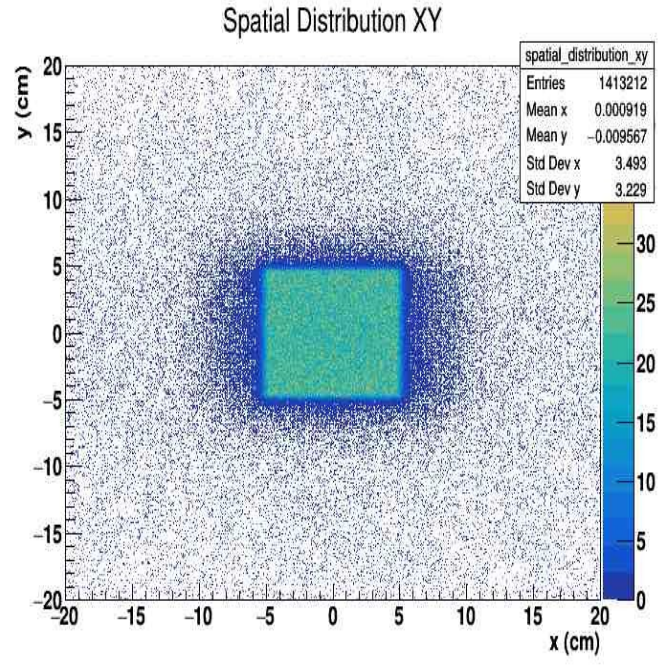
- **6MV bremsstrahlung Photon angular distribution**

./G4Linac_DataAnalysis PhaseSpaceData PhaseSpace.h5 photon_angular_distribution



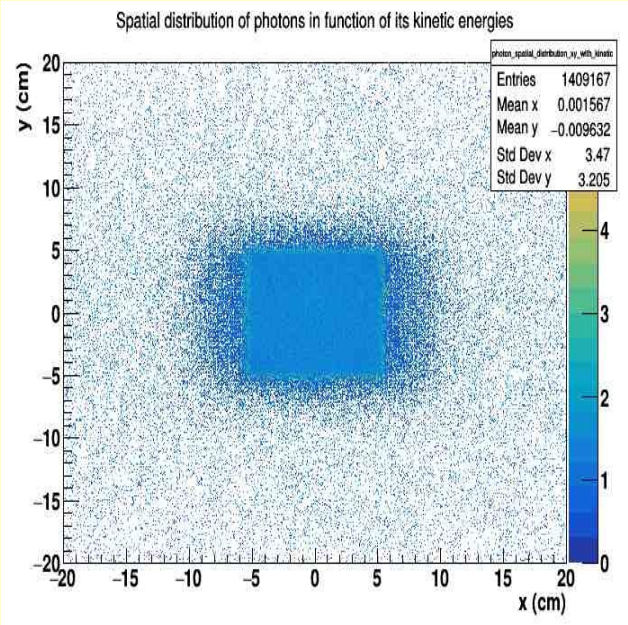
- **Particle spatial bidistribution**

./G4Linac_DataAnalysis PhaseSpaceData Phase_Space.h5 spatial_distribution_xy



- **6MV bremsstrahlung photon spatial bi-distribution in function of its kinetic energies**

./G4Linac_DataAnalysis PhaseSpaceData Phase_Space..h5
photon_spatial_distribution_xy_with_kinetic

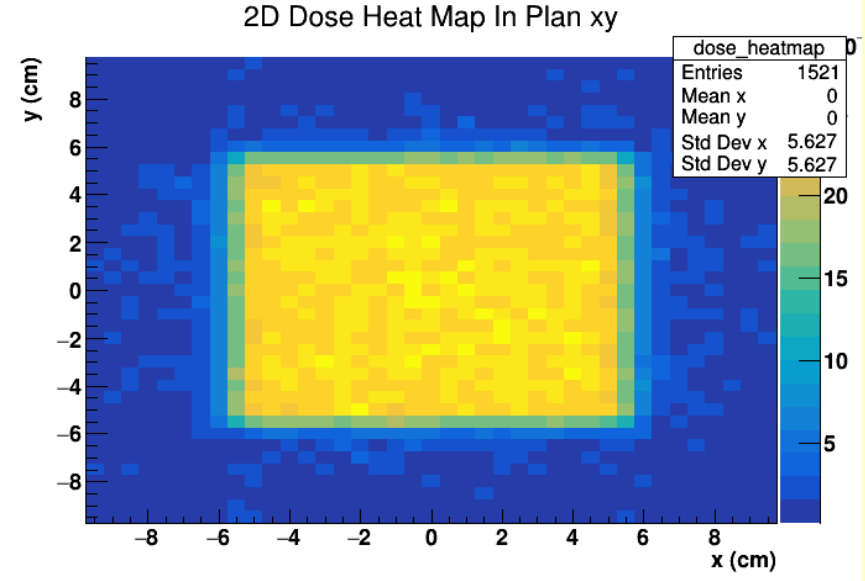


Here we can see a set of histogramms produced for a number of physical quantities thereby extracting

data from DosimetricData file called “dose.dat”.

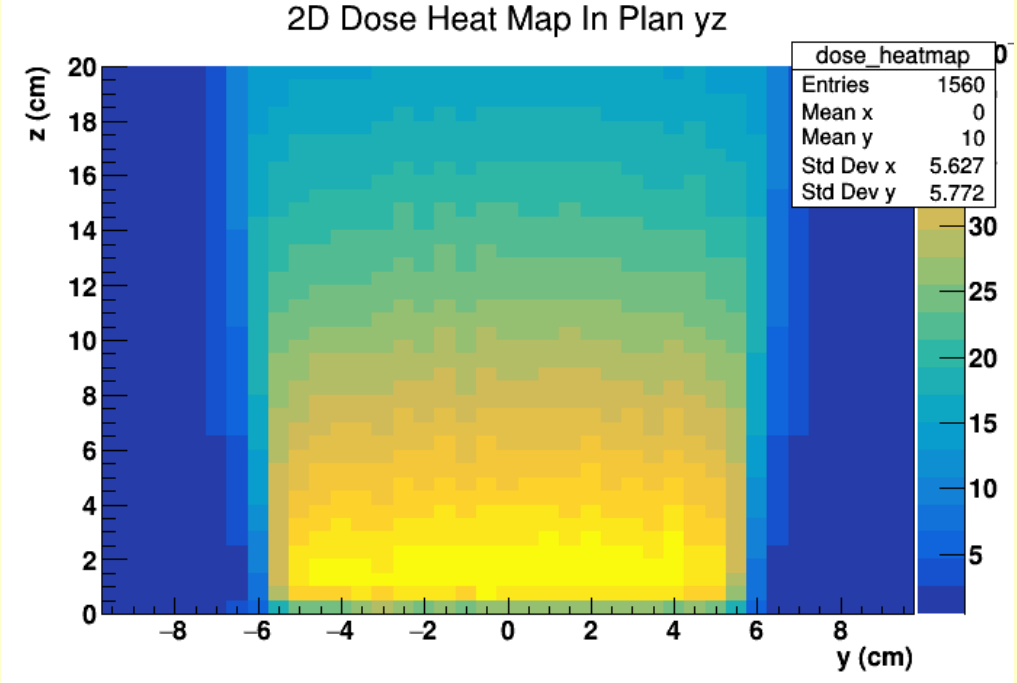
- **2D Dose heat map in plane xy**

```
./G4Linac_DataAnalysis DosimetricData dose.dat 2D_Dose_HeatMap z  39
```



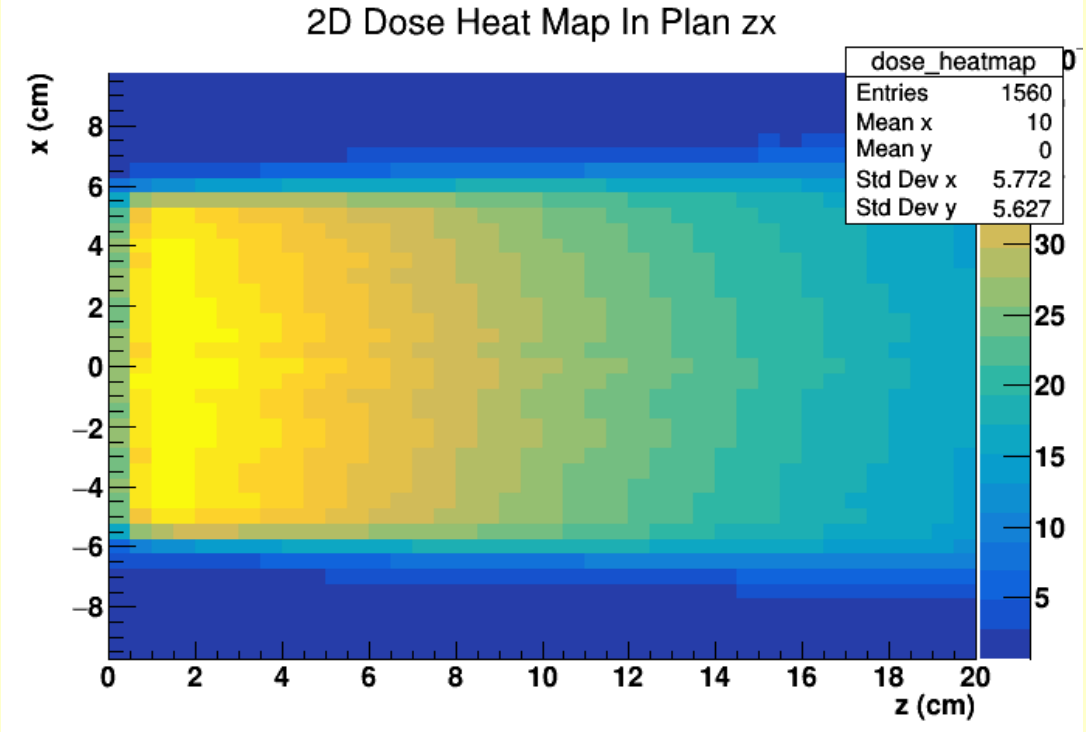
- **2D Dose heat map in plane yz**

```
./G4Linac_DataAnalysis DosimetricData dose.dat 2D_Dose_HeatMap  x  18
```



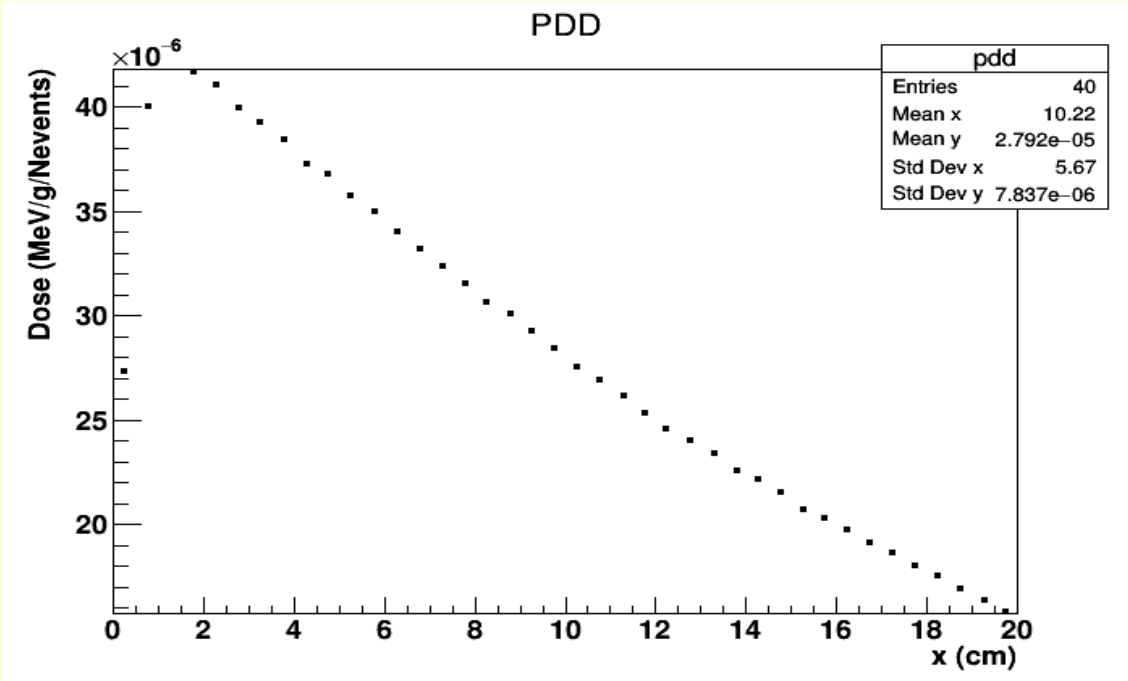
- **2D Dose heat map in plane zx**

```
./G4Linac_DataAnalysis DosimetricData dose.dat 2D_Dose_HeatMap  y  18
```



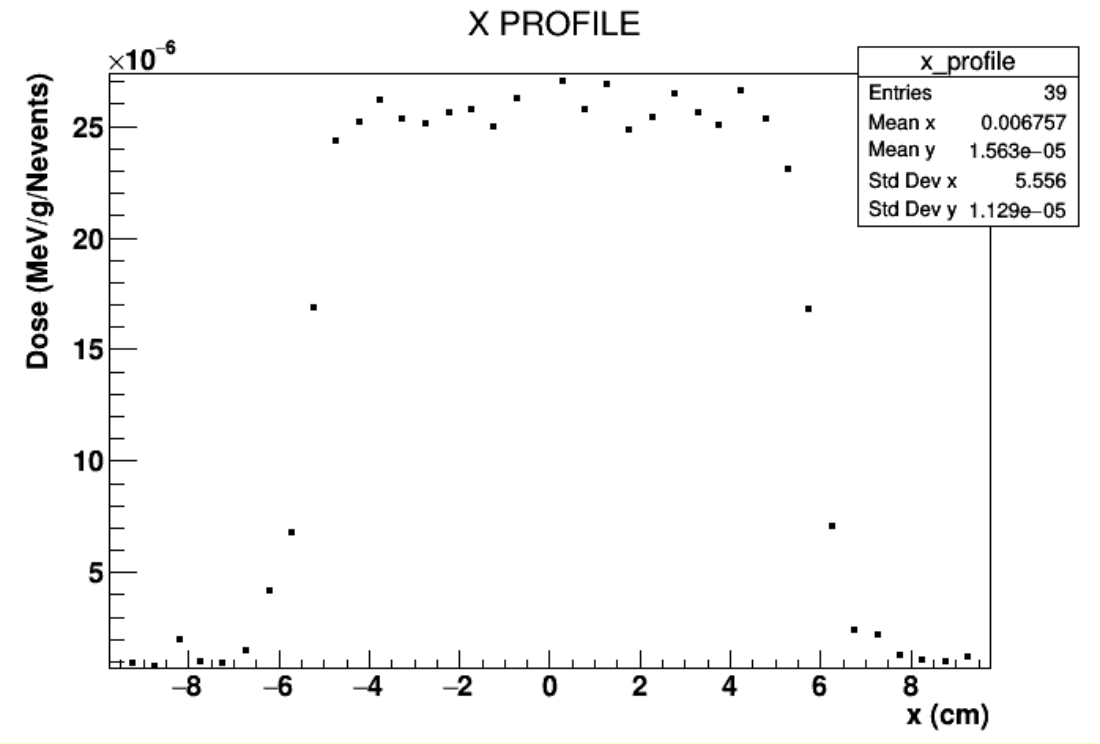
- **Percentage Depth Dose (PDD)**

```
./G4Linac_DataAnalysis DosimetricData dose.dat Percentage_Depth_Dose_Curve  18 18
```



- **Cross-plane Dose Profile**

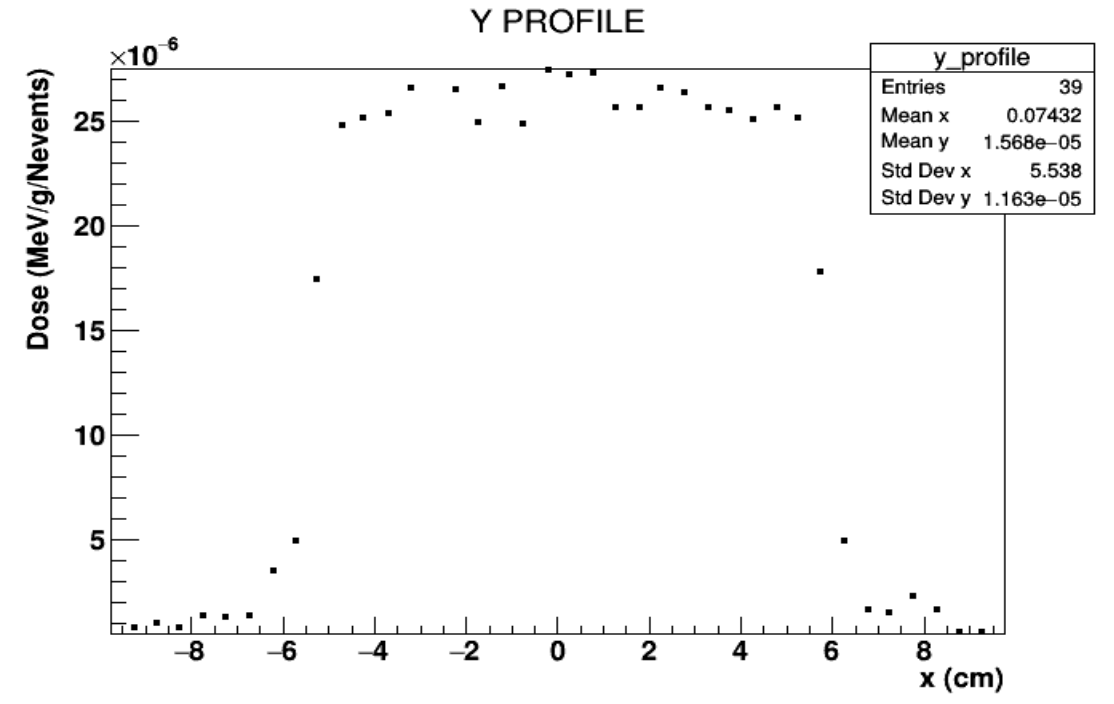
```
./G4Linac_DataAnalysis DosimetricData dose.dat  Cross-plane_Dose_Profile 18 39
```



- **In-plane Dose Profile**

```
./G4Linac_DataAnalysis DosimetricData dose.dat  In-plane_Dose_Profile 18 39
```

=



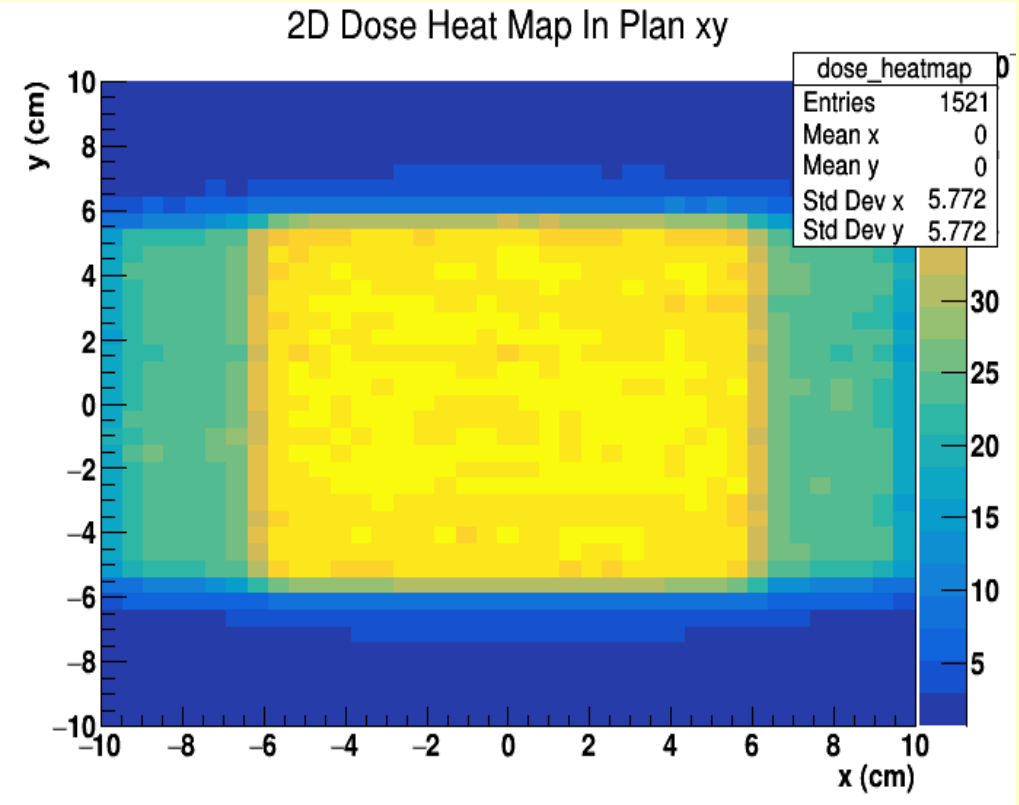
- **Dose simulation in water phantom irradiated by a 4-Fields**

For more realistic case, G4Linac_MT tool can be used to perform a 4-Fields conformational radiotherapy planning which is a common technique for treating pelvis anatomical region. In order to perform this job, users should enable h5Phase-Space multi-rotating option and they should set total rotation angle value. In this examplem, thhe case where the number of irradiation fields is 4, the running thread must be also the same number (numberOfThreads 4) and the total rotation angle will be fixed at 360 degree, so 4 rotation angles are considered (0°,90°,180° and 270°). The three following lines must be added to the "run.mac" macro file:

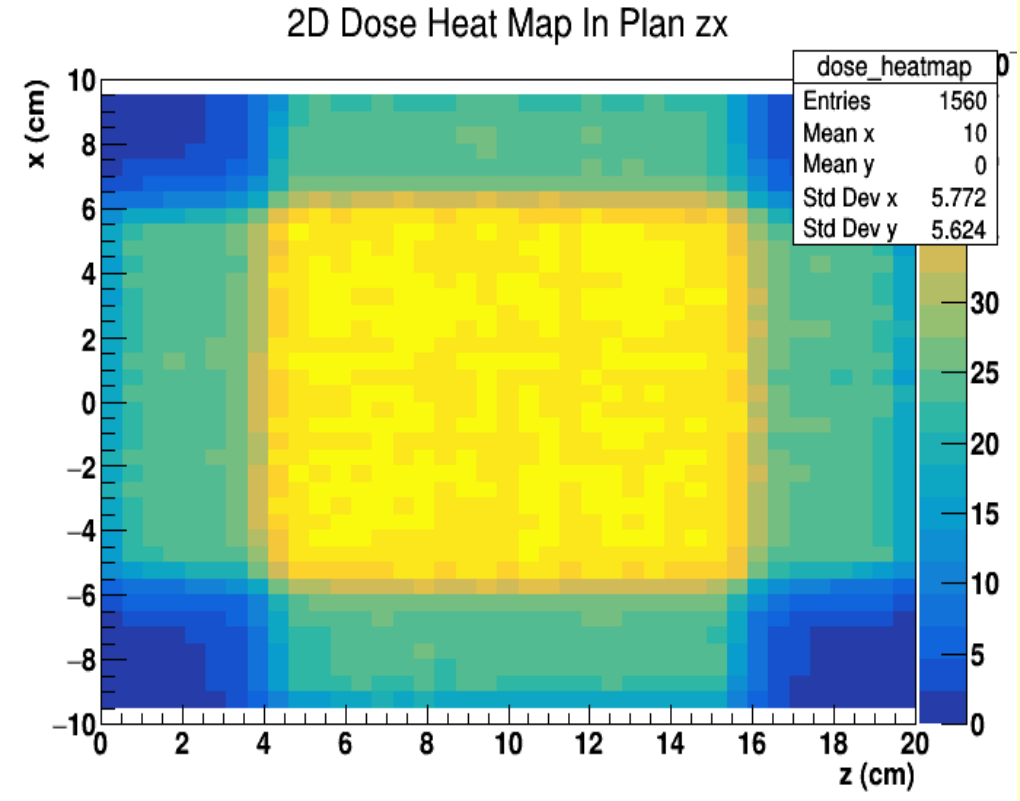
```
/Parameters/Rotate_h5PhaseSpace true
/Parameters/h5PhspRotationAngle 360 degree
/run/numberOfThreads 4
```

For N-Fields, we need to consider N threads and each h5Phase-Space executed in a given thread will be rotated with an angle of {(Total rotation angle / Number of threads)*(thread_ID+1)}. Here we present some simulation outputs concering a 4-Fields technique applied to a 6 MV photon beam configuring 10x10 cm^2 radiation field that irradiating a cubic water phantom of 20x20x20 cm3 dimensions. The rotation angles are 0, 90,180 and 270°.

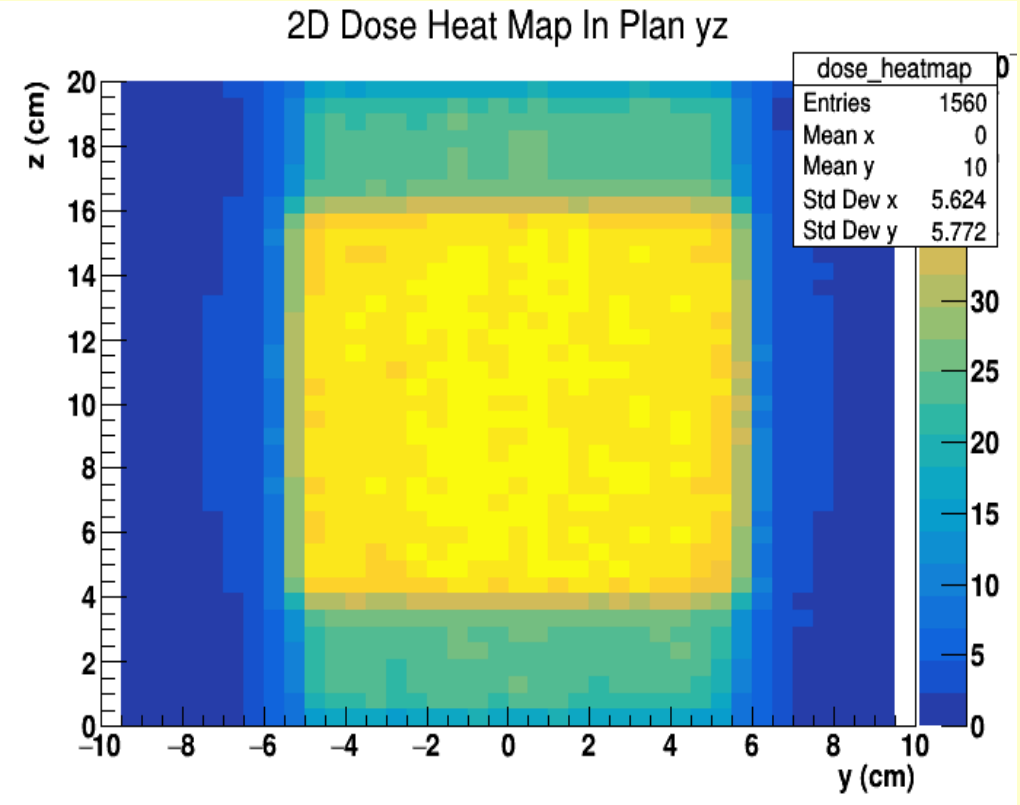
- **2D Dose heat map in plane xy**



2D Dose heat map in plane zx



- 2D Dose heat map in plane yz



- Dose simulation in a heterogeneous phantom irradiated by a Mono-Field

This example is identical as the one provided firstly, the only difference is that we introduce a heterogeneous slab into water phantom in order to have a heterogeneous medium. Indeed a slab of bone having dimensions of 3.5 X 3.5 X 4 cm^3 has been placed into water phantom at (0,0,20 cm). The description of materials and geometries of phantom considered in this example is as follows:

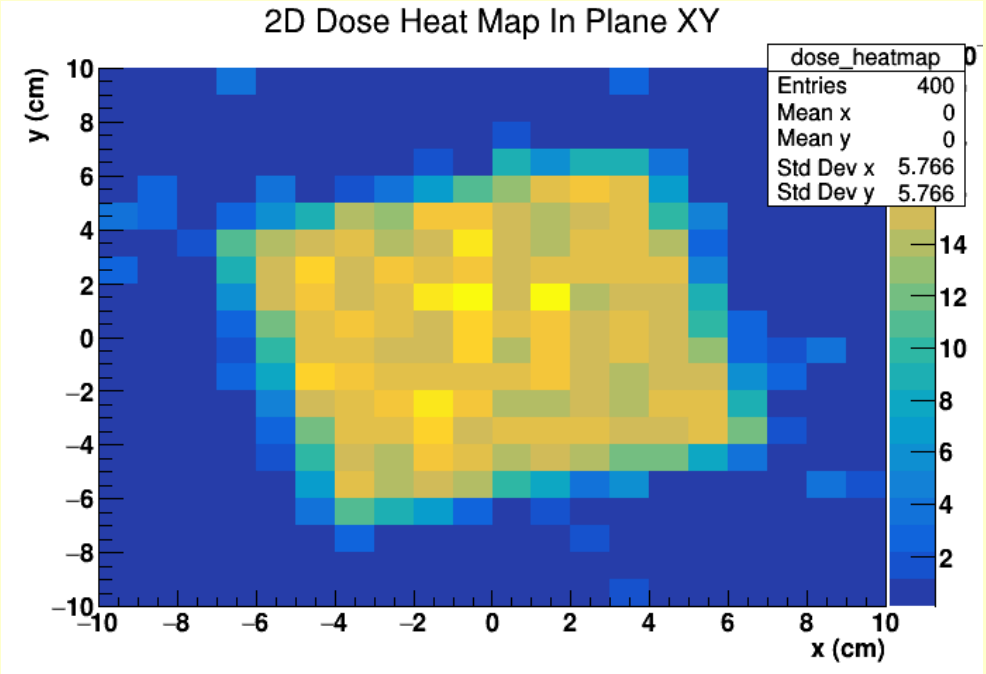
```
:ROTM RM0 0 0 0
:VOLU WORLD BOX 50*cm 50*cm 150*cm G4_Galactic
:VOLU phantom BOX 100  100 100 G4_WATER
:COLOR phantom  0 0.1 1
:PLACE phantom  0 WORLD RM0 0 0 20*cm
:SOLID BONE_SOLID BOX 1.75*cm 1.75*cm 2*cm
:VOLU BONE_LOGICAL BONE_SOLID G4_BONE_CORTICAL_ICRP
:PLACE BONE_LOGICAL 0 WORLD RM0 0 0 20*cm
```

- Dose simulation when rotating colimator by an angle of 15 degree

In this example we use the same h5Phase-Space produced previously but we rotate it along z axis by an angle of 15 degree. Here the code that must be implemented:

```
/Parameters/CollimatorRotAngleFlag  true
/Parameters/CollimatorRotAngle 15 degree
```

Dose heat map produced in plane xy is shown in the following figure:

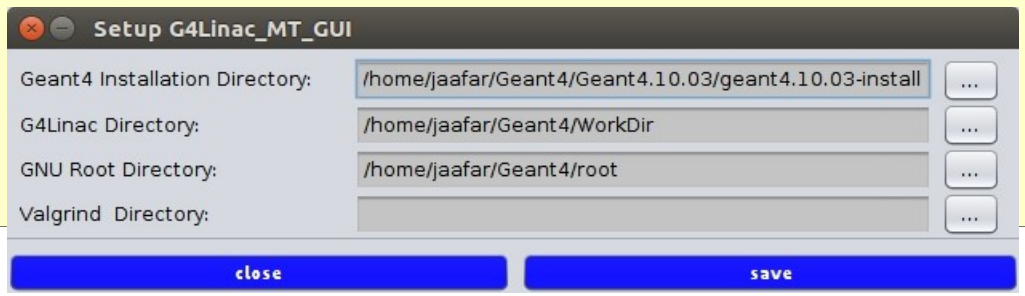


4. Faster building simulation with the help of G4Linac_MT_GUI

A Java-based GUI application has been developed called G4Linac_MT_GUI which make development of Linac simulation using G4Linac_MT tool much easier by avoiding time spent by writing by hand all macro files needed by a simulation. This Java-based application has an easy-to-use friendly graphical interface allowing users to code their simulation in a fast way. In addition, it contains a very interesting tool that handle geometry and material specification of such device which makeGeant4 Geometry From Text Language transparent to users, so they don't need to have knowledge of its roles.

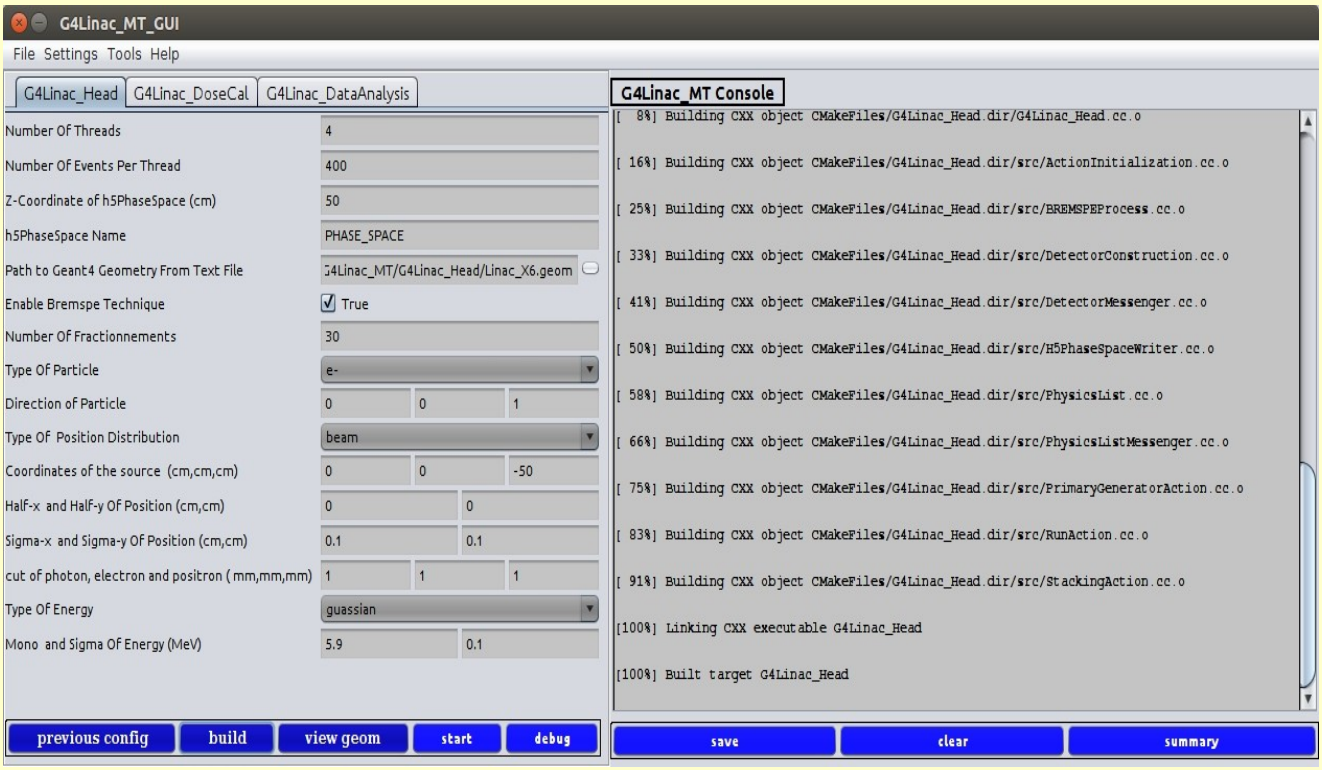
Firstly users should define some variables:

- Setuping G4Linac_MT_GUI



- **Linac treatment head modeling**

G4Linac_Head tab, allow users to specify all parameter related to a linac head simulation including: physics parameters, reduction variance techniques, PhaseSpace parameters, number of particles per thread and number of threads. Users should fill all parameters, then by clicking on build button the G4Linac_Head will be built.



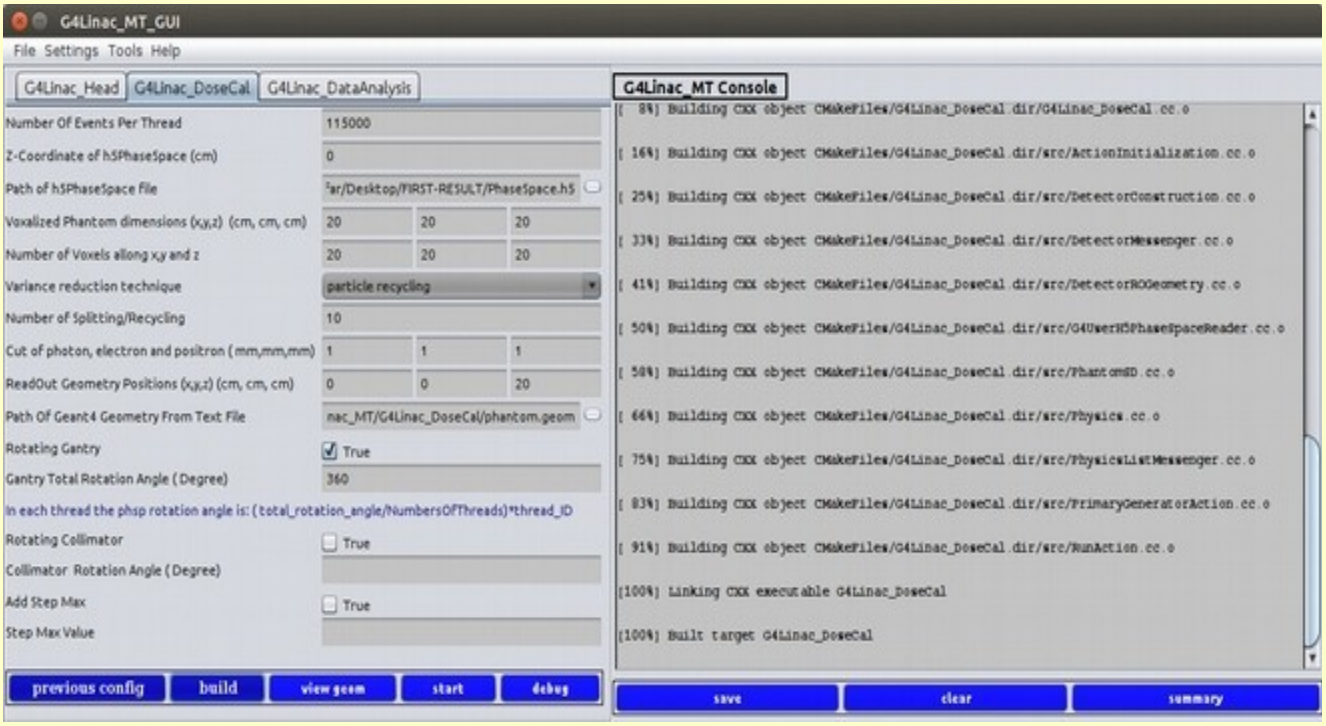
After that, users is free to choice between viewing geometry, starting or debuging a given simulation.

Go more in depth, the zone of text in the rigtht side of window is considerd as terminal emulator, the ouptut of a simulation is shown here, users can save it on a text

file by clicking on "save" button, the content of this zone of text can be cleaned by clicking on "clear" button. A summary of a such simulation will be dumped to the console this after the end of a Geant4 Run unit, if users clicking one "summary" button.

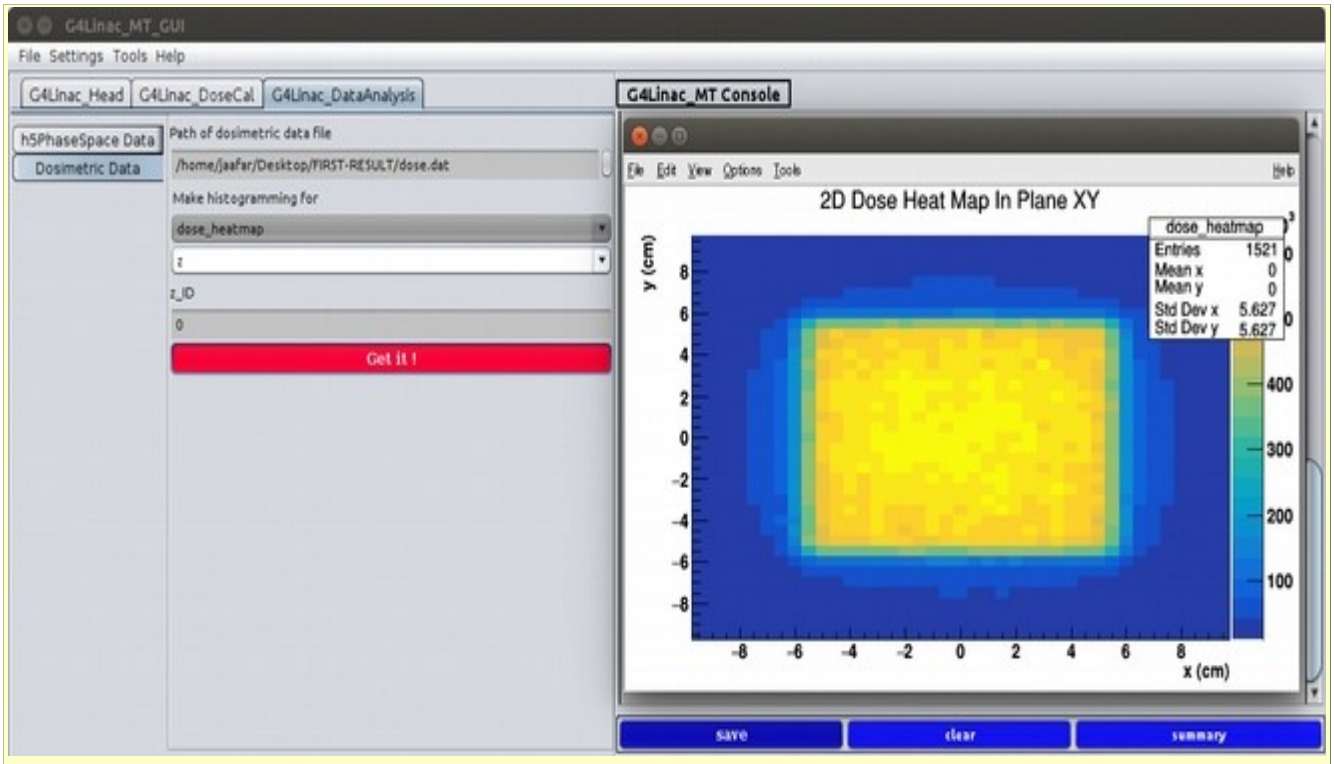
- **Modelling dose distribution in phantoms**

G4Linac_DoseCal, allow users to select all parameters needed by doses calculation module including: voxalized phantom sizes, number of voxels along x, y and z, number of particles per thread, physics parameters, rotation angle of gantry, rotation angle of collimator and many others.



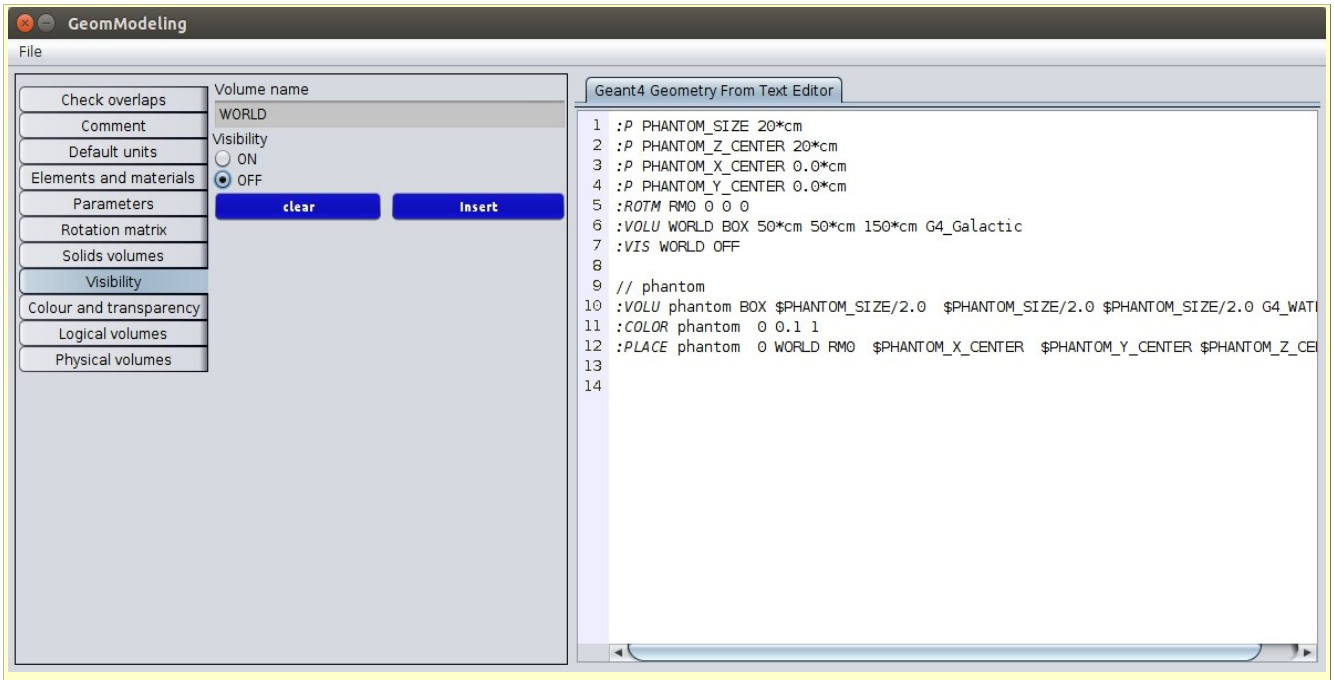
- **Histogramming of dosimetric and h5Phase-Space Data**

G4Linac_DataAnalysis module allow users to do histogrammig jobs for a number of physics quantities. Here we can see in the following a screenshot of this module:



- **Fastly modeling geometry and materials by using GeomModeling tool**

Users can in an easy way perform modelization of materials and geometry of a such Linac or a given phantom, this by clicking on "tools->GeomModeling" menu, after that a new widow will be apperead that contains all required visual components that help users to built a model of Linac head or phantom.



5. 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. Alternatively, users can use gdb tool for debuging as follows:

```
gdb
file ./ nameofModule
run run.mac
```

Sometime when user forgot to set the name of h5Phase-Space file in "run.mac" macro file, he probably will hit by an error message like this:

```
#000: H5F.c line 586 in H5Fopen(): unable to open file
    major: File accessibilty
    minor: Unable to open file
  #001: H5Fint.c line 1236 in H5F_open(): unable to open file: time = Thu Sep 14 15:45:17 2017
, name = 'PhaseSpace_x6.h5', tent_flags = 0
    major: File accessibilty
```

```
    minor: Unable to open file
  #002: H5FD.c line 809 in H5FD_open(): open failed
    major: Virtual File Layer
    minor: Unable to initialize object
  #003: H5FDsec2.c line 346 in H5FD_sec2_open(): unable to open file: name =
'PhaseSpace_x6.h5', errno = 2, error message = 'No such file or directory', flags = 0, o_flags = 0
    major: File accessibilty
    minor: Unable to open file
terminate called after throwing an instance of 'H5::FileIException'
Aborted (core dumped)
```

To resolve this issue, users need to set the correct name of h5Phase-Space file in the macro file.

- I need a help

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

6. For developers

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

7. License

G4Linac_MT is an open-source code distributed under the GNU General Public License. This code includes software developed by Members of the Geant4 Collaboration (http://cern.ch/Geant4).

```
//
// ********************************************************************
// * License and Disclaimer                                           *
// *                                                                  *
// * The  Geant4 software  is  copyright of the Copyright Holders  of *
// * the Geant4 Collaboration.  It is provided  under  the terms  and *
// * conditions of the Geant4 Software License,  included in the file *
```

```
// * LICENSE and available at  http://cern.ch/geant4/license .  These *
// * include a list of copyright holders.                              *
// *                                                                   *
// * Neither the authors of this software system, nor their employing *
// * institutes,nor the agencies providing financial support for this *
// * work  make  any representation or  warranty, express or implied, *
// * regarding  this  software system or assume any liability for its *
// * use.  Please see the license in the file  LICENSE  and URL above *
// * for the full disclaimer and the limitation of liability.         *
// *                                                                   *
// * This  code  implementation is the result of  the  scientific and *
// * technical work of the GEANT4 collaboration.                       *
// * By using,  copying,  modifying or  distributing the software (or *
// * any work based  on the software)  you  agree  to acknowledge its *
// * use  in  resulting  scientific  publications,  and indicate your *
// * acceptance of all terms of the Geant4 Software license.           *
// *******************************************************************//
```