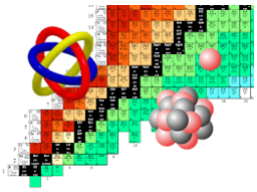
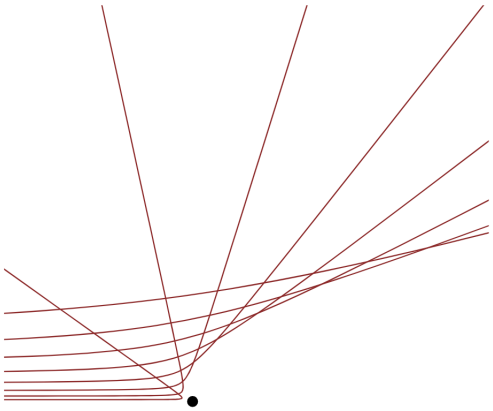
Instituto de Física da Universidade de São Paulo
Group of direct reactions with exotic nuclei



A short guide for *secRIBRAS*

Program developed to simulate the production of secondary beams in the RIBRAS facility



Kaique Albuquerque[1]
Osvaldo Camargo Botelho dos Santos[2]

São Paulo, Brasil

[1]kaique.albuquerque@usp.br
[2]osvaldo.santos@usp.br

Contents

1 A brief sumary of the program

1.1 The primary purpose of the program

Stable lithium atoms exist in nature in two isotopic forms ^{6}Li and ^{7}Li and also in three unstable form as ^{8}Li, ^{9}Li and ^{11}Li. For the energies used in RIBRAS facility, the main processes are the elastic scattering and fusion reaction, although stripping and breakup are also noteworthy. The ^{8}Li beam is produced in two different states of energy, in the ground state $(gs, 2^{+})$ and in the first exited state $(E_{exc} = 0.980\ MeV, 1^{+})$

A study conducted by doctor Osvaldo in 2021 has measured the elastic scattering and reactions of ^{8}Li in light mass targets. In his work, a study was made of the intensity of these two beams in secondary target. In the figure below, It is possible the visualize the relationship between the counts and current for these two beams.

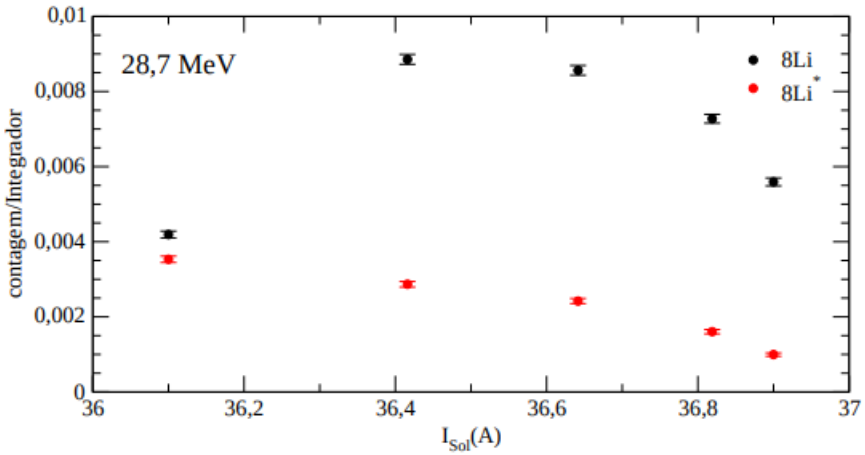


Figure 1: ^{8}Li production for the two different stated of energy.

That way, a Monte Carlo simulation was made with the aim to study the possibility of the blockage of the ^{8}Li first excited state before the secondary chamber. The results of the simulation showed that the complete blockage is possible, using a blocker at a certain position from the target. The next figure shows the first two-dimensional trajectories produced by the program in the blocking region.

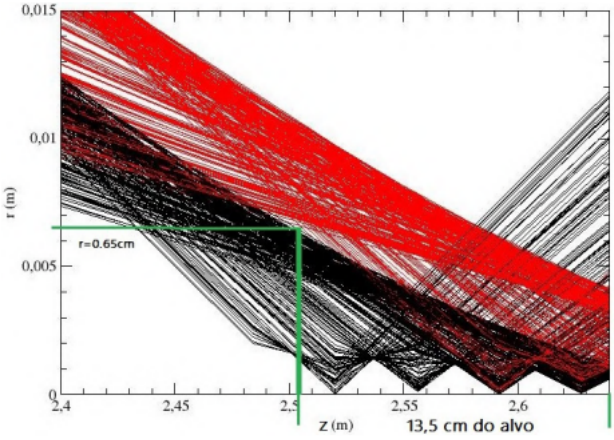


Figure 2: Pre simulation made by Osvaldo.

1.2 Program evolution and bi-parametrics

The program development period can be divided in two parts. The first one was dedicated exclusively to the evolution of the secondary beam program simulation, named "SecBeam". In this way, many simulation an analysis tool was integrated into the primary program developed by Osvaldo in order to increase its veracity.

The program considers all important physical principles. For the primary beam, It is considered the spatial and energy dispersion, in both cases as a Gaussian distribution. In general, to the spatial distribution is used σ as $0.5\ mm$ and for the energy as $40\ keV$. Furthermore, the reaction points in the primary target obey the spatial dispersion of the primary beam and in the direction of the beam, it is considered a constant distribution.

Inside the primary target, all important physical are considered, as the electronic an nuclear straggling, the kinematics of the reaction and the cross section - of the simulated reaction - in the simulation of the scattering angle.

The trajectories of the particles are calculated using the linear relationship between the current of the solenoid and the magnetic rigidity, calculated through

$$B\rho = \frac{\sqrt{2mE}}{q} \Longrightarrow I = \alpha \cdot B\rho + \beta.$$

In this matter, four linear regressions was made for the automatic current calculation, three involving the reactions

Radioactive beam	Production reaction
^{8}Li	^{9}Be(^{7}Li,^{8}Li)
^{6}He	^{9}Be(^{7}Li,^{6}He)
^{8}B	^{3}He(^{6}Li,^{8}B)

and another that represents the average of the three first ones.

The data analysis was divided in three big parts: energies distributions, angular distribution and trajectories. For the first two, codes have been programmed to generated in automatic way the histograms after the primary target and before the tertiary target (after the blocker). For the trajectories, the program plots the $\rho \times z$ chart and the $x \times y \times z$. In addition, the scatter on the secondary target also is plotted.

The second part of the period was used as a way to complete the study of the secondary beam. For this propose, a code was developed to simulate a scattered beam in the secondary target of RIBRAS, in direction to the telescopic detector system, providing us the production of two dimensional specter of energy for the detected particles. This part of the program consider all the physical principals previously commented, as electronic and nuclear straggling and reaction kinematics. The code also provide the energy information and the scattering angle about the recoil particle.

2 How to use it?

The following topics teach how to use the secRIBRAS in its lastest version, that is actually 15.

2.1 Preparing the work environment and compiling the program

To run the program it's not necessary to install any specific physics software. All programs used in the current run are located in the *Apps* folder inside the *bin* folder:

cinema, *kineq*, *stopx*, *stopx2* and *twsp*. Despite that, it is recommended install the app *Grace* to visualize in a practical way the output data in *secbeam* folder.

An immediate example is the $\rho \times z$ trajectories generate automatically by the program. This chart is produced with the `ploGraf` function that uses Grace to draw all the trajectories and the geometric aspects of the first solenoid, as the collimators and the faraday cup.

To install grace, you can access the following website and get the necessary informations: https://plasma-gate.weizmann.ac.il/Grace/doc/UsersGuide.html. In brief, to install the program on linux, open the terminal using `Ctrl + T` in any workspace, an type

```
1    $      sudo apt install grace
```

Wait until the installation is finished and than close the terminal. It is also recommended restart the computer after the installation.

When downloading the secRIBRAS folder, the program is already compiled, but to make sure the every thing is working and **set the running permissions of the softwares** in *Apps* folder, the following commands need to be given inside the *SecRIBRAS* folder:

```
1    .../SecRIBRAS$      make clean
```

and

```
1    .../SecRIBRAS$      make
```

The first command clear the *obj* folder, that contains all object files created in previously compilations. The second one, call the make file in the same folder, which in turn have the followin commands:

```
1    g++ -c $(LIBS)/library.cpp -I $(INCLUDE) -o $(OBJ)/library.o
2    g++ -c $(LIBS)/Plot2D.cpp -I $(INCLUDE) -o $(OBJ)/Plot2D.o
3    g++ -c $(LIBS)/MakeHistogram.cpp -I $(INCLUDE) -o $(OBJ)/
        MakeHistogram.o
4    g++ -c $(LIBS)/MakeScatter.cpp -I $(INCLUDE) -o $(OBJ)/
        MakeScatter.o
5    g++ -c $(LIBS)/secbeam.cpp -I $(INCLUDE) -o $(OBJ)/secbeam.o
6    g++ -c $(LIBS)/bipa.cpp -I $(INCLUDE) -o $(OBJ)/bipa.o
7    g++ $(APPS)/secRIBRAS.cpp $(OBJ)/*.o -I $(INCLUDE) -o $(BIN)/
        secRIBRAS
8    gfortran $(APPS)/cinema.for -o $(BIN)/Apps/cinema
9    gfortran $(APPS)/twsp.f90 -o $(BIN)/Apps/twsp
10   sudo chmod +x $(BIN)/Apps/stopx2
11   sudo chmod +x $(BIN)/Apps/stopx
12   sudo chmod +x $(BIN)/Apps/kineq
```

Each one of these has a specific function in the compilation. The first seven compile the files that contains the intern functions of the program. The line 8 and 9 is responsible to compile the *cinema* and *twsp* code (both are in *apps* folder). In the last three, the `chmod` command ensure (give the permission) that the softwares can be called as a executable by the secRIBRAS.

2.2 Input parameters

With all set, It's time to insert the input parameters of the program. All of these are in the *in.dat* file, inside the *bin* folder, with the exception of the informations about the cross section, that is inside the *InputFiles* in *SecBeam* folder (red arrows).

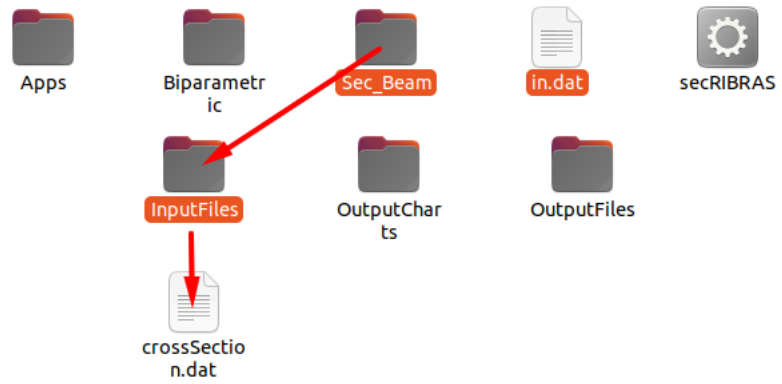


Figure 3: Arrangement of input files in program folders.

In the *in.dat* file, there are some instructions to how fill the file with the necessary informations. The first package of information is about the simulation of secondary beam and its instructions are following.

```
Secundary beam (yes / no):
NVector   PrimaryBeam   SecundaryBeam   PrimaryTarget  massSecBeam(uma) qSecBeam  Qreac(MeV) EnerPriBeam(MeV)  ThickPriTarg(um)
Minimum angle(deg) Maximum angle(deg)  sRadiusPriBeam(m) sEnergyPriBeam(MeV) FirstCollimatorR(m) FaradSayCupR(m)  LollipopR(m)  SecondCollimatorR(m)
posBlocker(m) BlockerR(m)
CrossSection(yes / no)
fixCurrent(yes Current(A) / no)
Enerstragg(yes / no)

Contaminants:
Sec.Beam  Mass.Sec.Beam   q.Sec.Beam
END
```

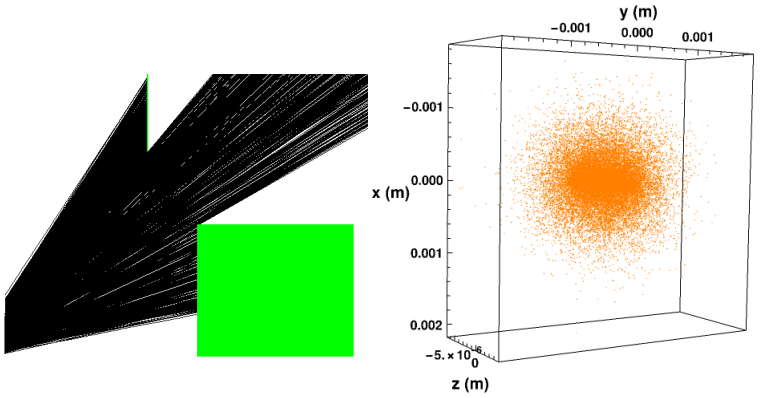
Figure 4: Instructions about the input of the secondary beam simulation.

The first option is just the choice if the secondary beam will be simulated or not. Each one of the other options are explained following together with its influence in the program.

i. Secondary beam

- **NVector** is the number of events that will be simulated in the first target.
- **PrimaryBeam** is the tag of the nucleus that makes up the primary beam. For example, if it is the ^{7}Li, the user must insert 7Li.
- **SecondaryBeam** is the tag of the nucleus that makes up the secondary beam. For example, if it is the ^{6}He, the user must insert 6He.
- **PrimaryTarget** is the tag of the nucleus that makes up the primary target. For example, if it is the ^{9}Be, the user must insert 9Be.
- **massSecBeam** is the mass of the secondary beam, given in atomic units (uma). For example, if it is the ^{6}He, the user have to write 6.
- **qSecBeam** is the charge of the secondary beam, given in number of elementary charge (e^{-}). For example, if the secondary beam is ^{8}Li, that has charge $+3$, the user must insert 3.
- **Qreac** is the excitation energy of the reaction, given in MeV. For example, the ^{8}Li of the reaction $^{9}\mathrm{Be}(^{7}\mathrm{Li},^{8}\mathrm{Li})$ can be produced in two different states of energy: $^{8}\mathrm{Li}(gs, 2^{+})$, and $^{8}\mathrm{Li}(E_{exc} = 0.980\ \mathrm{MeV}, 1^{+})$, so to simulate both states, the user must insert 0.980.
- **EnerPriBeam** is the energy of the primary beam, given in MeV.

- **ThickPriTarg** is the thick of the primary target, given in cm. The negative sinal must be put in front of the value, this is an internal issue of energy calculation software.
- **Minimum angle** and **Maximum angle** set the angular range that the simulation will cover, given in degrees. The following figure is the $\rho \times z$ chart of the simulation, in it is possible to visualize the range from 1° to 8°.

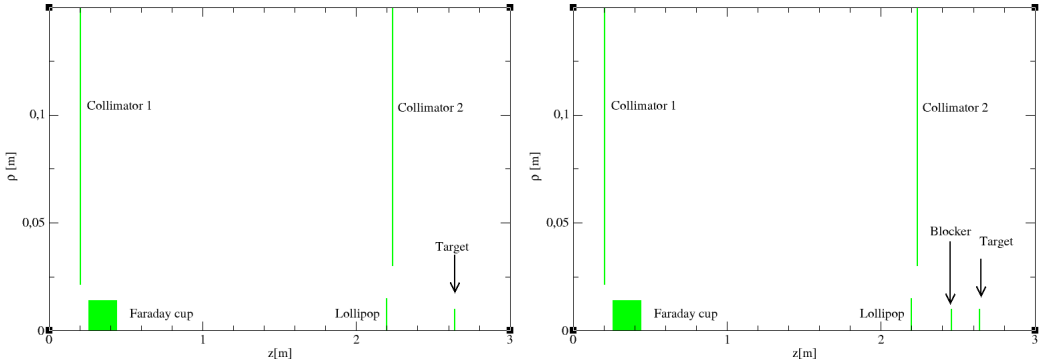


(a) 1° to 8° (b) Reaction points

Figure 5: Angular range and reaction points.

- **sRadiusPriBeam** is the σ value of Gaussian distribution with mean in zero of the primary beam, given in meters. This value is responsible to consider a spacial dispersion of the primary beam. Usually, it is considered 0.0005 m. The figure above shows the reaction points in the primary target in a simulation. This points distribution are direct consequence of the spacial distribution of the primary beam.
- **sEnergyPriBeam** is the σ value of Gaussian distribution with mean in the energy of the primary beam, given in MeV. This value is responsible to consider a energy dispersion of the primary beam. Usually, it is considered 0.04 MeV (40 keV).
- **FirstCollimatorR** is the radius of the first collimator, given in meters.
- **FaradayCupR** is the radius of the faraday cup, given in meters.
- **LollipopR** is the radius of the lollipop, given in meters.
- **SecondCollimatorR** is the radius of second collimator, given in meters.
- **posBlocker** is the position of a blocker, given in meters. This value must be greater than zero.
- **BlockerR** is the radius of the blocker, given in meters. To omit this blocker from the simulation just put the radius as zero.

  In the following figure are the geometric elements commented above arranged in a $\rho \times z$ graph.



(a) Geometric aspects without the blocker (b) Geometric aspects with the blocker

- **CrossSection** is the option to consider or not cross section calculated. If set *yes*, the cross section must be placed in the *InputFiles* folder inside *Sec_beam* folder. The program has the cross section of two different reactions: $^{9}\mathrm{Be}(^{7}\mathrm{Li},^{6}\mathrm{He})$ and $^{9}\mathrm{Be}(^{7}\mathrm{Li},^{8}\mathrm{Li})$. If set *no*, the program makes an isotropic simulation (uniform cross section).
- **fixCurrent** is the option to set a current (*yes*) or let the program calculate the ideal one for focusing (*no*). If the user set *yes*, it is also necessary gives the value of current in ampere, for example: *yes 36.9*.
- **Enerstragg** is the option to consider or not (*yes* or *no*) the eletronic straggling.

ii. **Contaminants** are particles produced in the primary target that for some reason converge their trajectories to the target. To include in the simulation your presence, It is necessary provide some information:

- **Sec.Beam** is the tag of the contaminant.
- **Mass.Sec.Beam** is the contaminant mass.
- **q.Sec.Beam** is the charge of the contaminant.

The user can provide as many contaminants as he want, always writring their information line after line. After the last line, it is important put the key word "END".

An example of input for secondary beam simulation can be observed below.

```
yes
1500    7Li     8Li     9Be     8           3          0.980   28    -0.0012
1       8       0.0005  0.04    0.02125 0.013625       0.0151  0.03
2.455   0.0
yes
no
yes

6He     6       2
4He     4       2
3H      3       1
END
```

Figure 6: An input example for secondary beam simulation. Note that reaction used was $^{9}\mathrm{Be}(^{7}\mathrm{Li},^{8}\mathrm{Li})$ with two different states of energy.

The second set of information is about the bi-parametric parameters. This part also has a quick explanation, as following.

```
Biparametric (yes/no)
Det.Tag  Det.Ang(deg)  Det.Shape(circular,rectangular)  {Det.radius(cm)}*OR*{Det.WidthX(cm) Det.WidthY(cm)}  Det.thick(cm)  Det.dist(m)
Sec.Beam  Targ.Tag      Eject.Part.Tag   Recoil.Part.Tag  QWindow(MeV)    Qgs.Proj(MeV)   Sec.Targ.Thick(cm)
END

Sec.Beam  Targ.Tag      Eject.Part.Tag   Recoil.Part.Tag  QWindow(MeV)    Qgs.Proj(MeV)   Sec.Targ.Thick(cm)
END
```

Figure 7: Instructions about the input of the bi-parametric simulation.

Again, the first option is the choice if the bi-parametric program will be simulated or not. The first line are the information about the detector, and the others about the beams. The following is the explanation of each parameter.

i. Detector

- **Det.Tag** is the tag of the particle that constitute the detector.
- **Det.Ang** is the angle that the detector will be positioned, given in degrees.
- **Det.Shape** is the shape of the detector. There are two different shapes available: rectangular and circular.
- **Det.radius** is the radius of the circular detector, only provide this information if the shape is circular.
- **Det.WidthX** and **Det.WidhtY** is the dimensions of the rectangular detector, only provide these information if the shape is rectangular.
- **Det.thick** is the thick of the detector, given in cm, with the previously commented signal detail.
- **Det.dist** is the distance between the center of the detector and the center of the secondary target.

ii. Bi-parametric of the secondary beam

- **Sec.Beam** is the tag of the secondary beam that arrives at the secondary target.
- **Targ.Tag** is the tag of the secondary target.
- **Eject.Part.Tag** is the eject particle that leaves the secondary target.
- **Recoil.Part.Tag** is the recoil particle that leaves the secondary target.
- **QWindow** is the window of energy of the excitation in the reaction.
- **Qgs.Proj** is the excitation of the reaction in the ground state.
- **Sec.Targ.Thick** is the thick of the secondary target.

The user can provide as many reaction as he want, always writring their information line after line. After the last line, it is important put the key word "END".

iii. Bi-parametric of the contaminats.

- **Sec.Beam** is the tag of the secondary beam that arrives at the secondary target.
- **Targ.Tag** is the tag of the secondary target.
- **Eject.Part.Tag** is the eject particle that leaves the secondary target.

- **Recoil.Part.Tag** is the recoil particle that leaves the secondary target.
- **QWindow** is the window of energy of the excitation in the reaction.
- **Qgs.Proj** is the excitation of the reaction in the ground state.
- **Sec.Targ.Thick** is the thick of the secondary target.

Again, you can simulate as many contaminants as you like, taking into account the fact that they must be previously simulated on the secondary beam.

In the following figure is an example of the *in.dat* file completely filled.

```
yes
1500    7Li     8Li     9Be     8           3          0.980   28    -0.0012
1       8       0.0005  0.04    0.02125 0.013625       0.0151  0.03
2.455   0.0
yes
no
yes

6He     6       2
4He     4       2
3H      3       1
END

yes
28Si    15      circular        0.55    -0.0052  0.07
8Li     58Ni    8Li     58Ni    0       0       -0.000242
8Li     58Ni    7Li     59Ni    14      6.966   -0.000242
END

6He     58Ni    6He     58Ni    0       0       -0.000242
4He     58Ni    4He     58Ni    0       0       -0.000242
3H      58Ni    3H      58Ni    0       0       -0.000242
END
```

Figure 8: All information filled in the *in.dat*.

2.3 Running the program in the terminal

After prepare the environment and filling in all the necessary information, the program can be run in the terminal. For this, open one of them inside the *bin* folder, and type

```
1     .../bin$      secRIBRAS < in.dat
```

it is always recommended to use the 'tab' key to complete typing, this avoid typos.

The user can follow the execution in the same terminal, with the load bars and status messages.

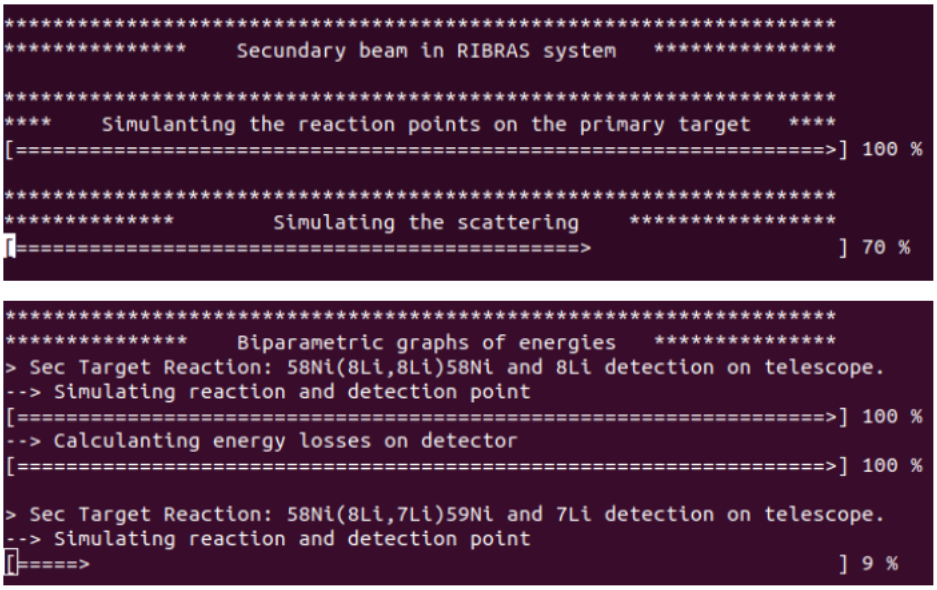


Figure 9: Program running in the terminal in two different times.

2.4 Output data

At the end of the secondary beam simulation, the program shows three values: mean energy on target, the solenoid current and the average straggling. The first one is the average energy of the particles that arrive at the target, i.e., the particles that collide with the colimators, the Faraday cup or with the blockers are not consider for the calculation of this value. The second one is the current used to focus the beams in the secondary target, it can be calculated with the magnetic rigidity, as explained previously, or it is a input value provide in the *in.dat* file. The last one is the average straggling in the primary target.

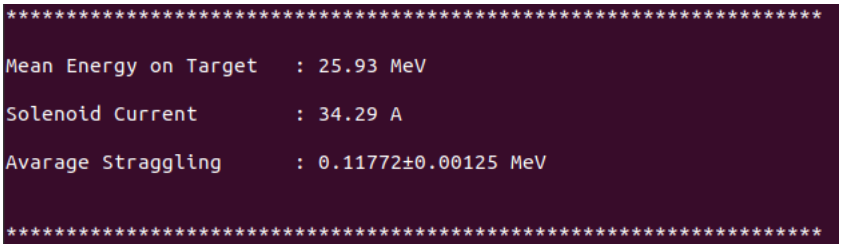


Figure 10: Example of the three output data in the terminal.

As mentioned, the outputs was divided in three big parts: energies distributions, angular distribution and trajectories. Furthermore, the program divide the data for secondary beam simulation for the bi-parametric simulation. For the first, the file and folder arrangement can be observed in the following figure, where the first folder is *bin*.

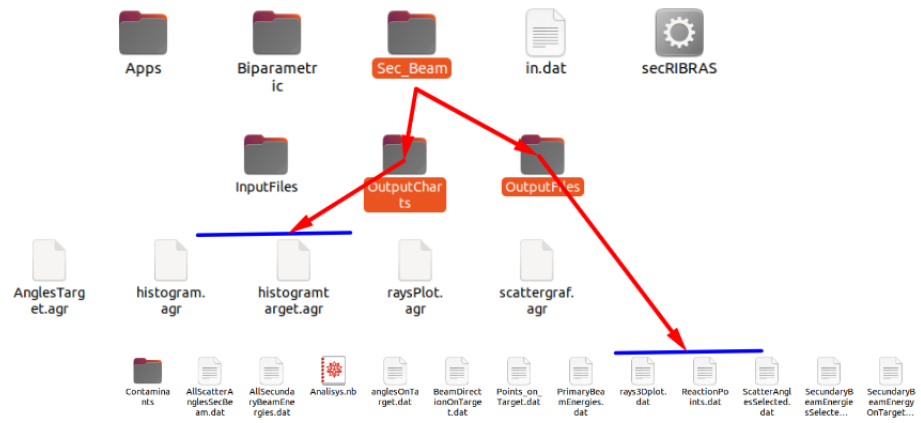


Figure 11: Output files for secondary beam simulation.

The *OutputCharts* is the folder where the program puts the ready-made graphics, using the Grace software. This charts are:

i. **AnglesTarg** is the angular distribution of the particles arriving the secondary target. The angle used in this histogram is the angle that the trajectories make with the perpendicular direction of the target.

ii. **histogram** is the energy distribution of the particles that enter in the solenoid.

iii. **histogramtarget** is the energy distribution of the particles arriving the secondary target.

iv. **raysPlot** is the two dimensional trajectories inside the solenoid.

v. **scattergraf** is the dispersion of the particles on the secondary target.

The folder *OutputFiles* contain all distribution data in text files (*.dat* more precisely). It is necessary to comment on four of them:

i. **Contaminants** is the folder where the energy files of the contaminants are placed.

ii. **Points_on_Target** is the three dimensional coordinates of the arrival point on secondary target.

iii. **rays3Dplot** is the three dimensional trajectories.

iv. **ReactionPoints** is the three dimensional coordinates of the reaction points in the primary target.

For the bi-parametric simulation, the files are also text files, like as in the previously secondary beam folder.

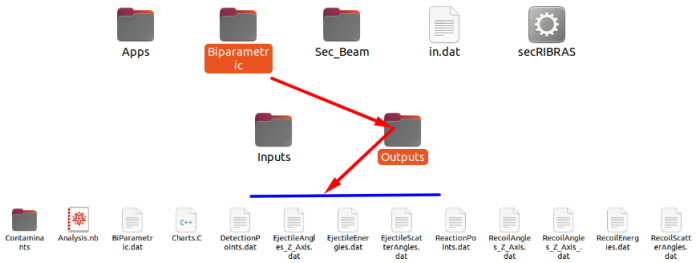


Figure 12: Output files for bi-parametric.

In this case, It is crucial note the difference between the two types of angular distributions: "_Z_axis" and "Scatter". The first is the angle between the trajectories and the perpendicular direction from the secondary target (z axis). In the other hand, the Scatter is the denomination used to refer to the angle between the ejectile and recoil directions. A schematic of this can be seen below.

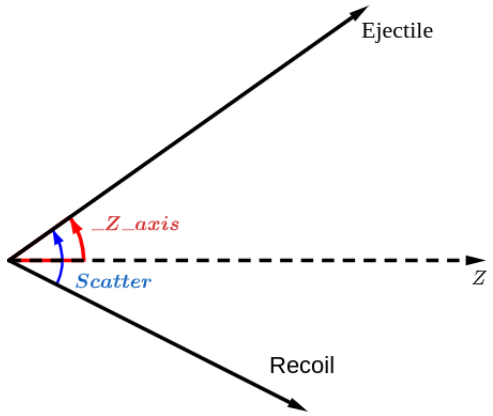


Figure 13: Difference between "_Z_axis" and "Scatter".

To help the user in the visualization of the data, a macro was written to make all necessary charts with the text files. This macro runs in the root terminal. To use it, open a terminal inside the *Outputs* folder and call root, like as

```
...bin/Biparametric/Outputs$      root -l
```

After open, charge the macro using

```
root[0]   .L Charts.C
```

this command prepares all functions of the macro to be used. If you want to see what functions can be called, type

```
root[1]   help()
```

and all available function will appear.

```
root [0] .L Charts.C
root [1] help()
*****************************************
This program has the following functions:
---> biparametric()
---> biparametric_contaminants()
---> AngularHistogram(const char *file_name)
---> EnergyHistogram(const char *file_name)
---> ReactionPoints()
---> DetectionPoints()
---> DetectionAndReactionPoints()
*****************************************
root [2]
```

Figure 14: root running.

3 Examples

In this section you will find three examples of simulations using *SecBeam*.

3.1 The ^{8}Li production

The production o ^{8}Li occurs through the reaction:

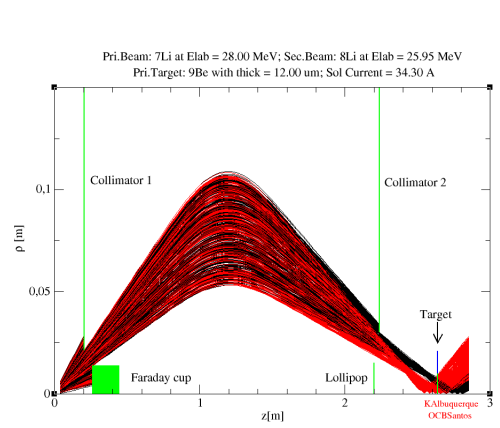
$$^{7}\mathrm{Li}(^{9}\mathrm{Be},^{8}\,\mathrm{Li})$$

and produce this particle in two different states of energy, ground state (gs , 2^{+}) and in the first excited state ($E_{exc} = 0,980$ MeV, 1^{+}). Then, to do a complete simulation, the input file looks like this.

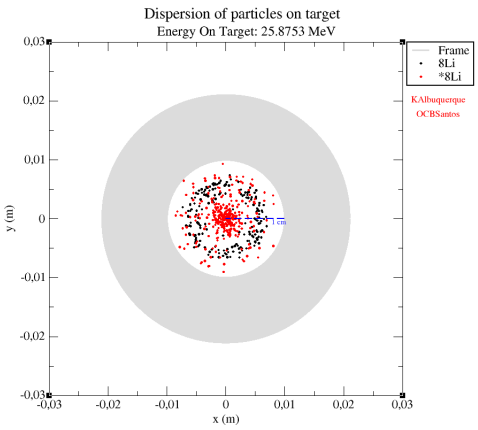
```
yes
1500    7Li     8Li     9Be     8           3          0.980   28    -0.0012
1       8       0.0005  0.04    0.02125 0.013625       0.0151  0.03
2.455   0.0
yes
no
yes

6He     6       2
4He     4       2
END

yes
28Si    15      circular        0.55    -0.0052  0.07
8Li     58Ni    8Li     58Ni    0       0       -0.000242
8Li     58Ni    7Li     59Ni    14      6.966   -0.000242
END

6He     58Ni    6He     58Ni    0       0       -0.000242
4He     58Ni    4He     58Ni    0       0       -0.000242
END
```

Figure 15: Input file for example 1.

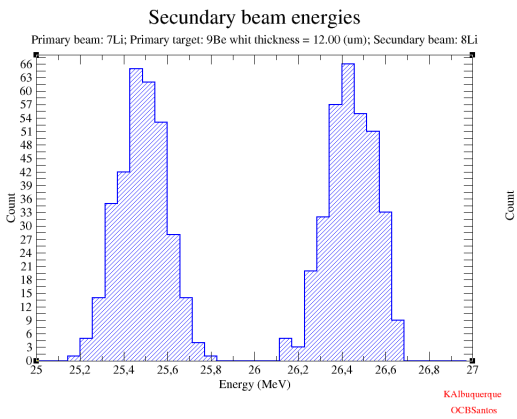
For the secondary beam, the output charts:



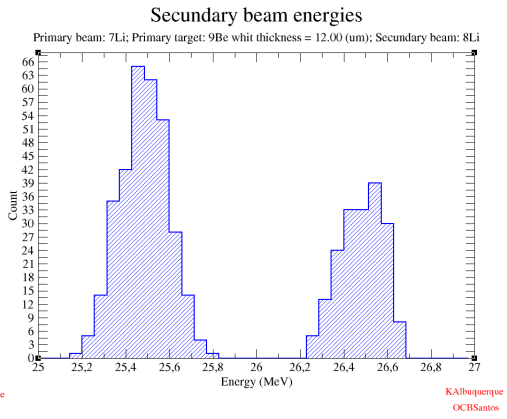
(a) Trajectories $\rho \times z$.



(b) Scatter on target.



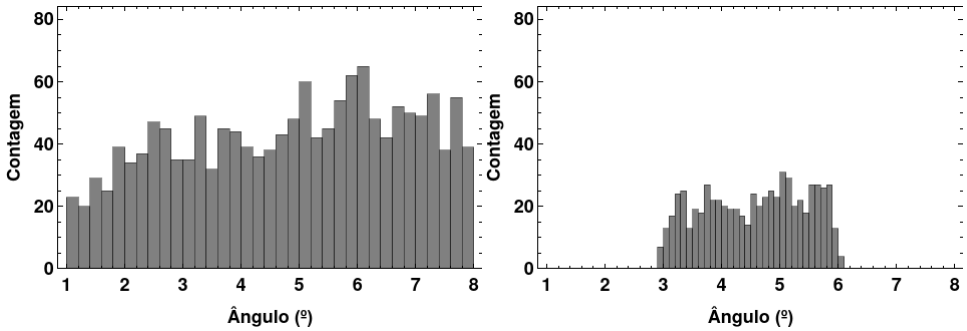
(c) Histogram of energy inside the solenoid.



(d) Histogram of energy on target

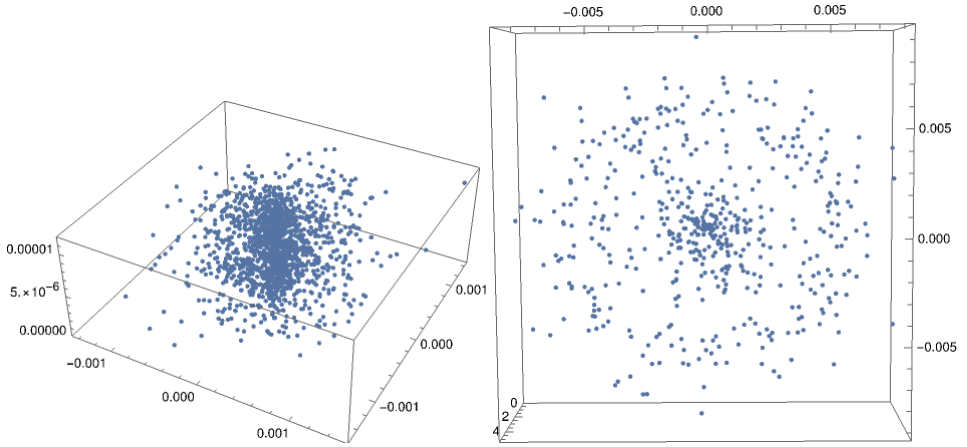
Figure 16: Charts automatically made by the program.

Using the wolfram mathematica to analyse the files in the *OutputFiles* folder, I represent here some following charts.



(a) Histogram of angles produced.

(b) Histogram of the angles selected.



(c) Reaction points in primary target.

(d) Points in the secondary target.

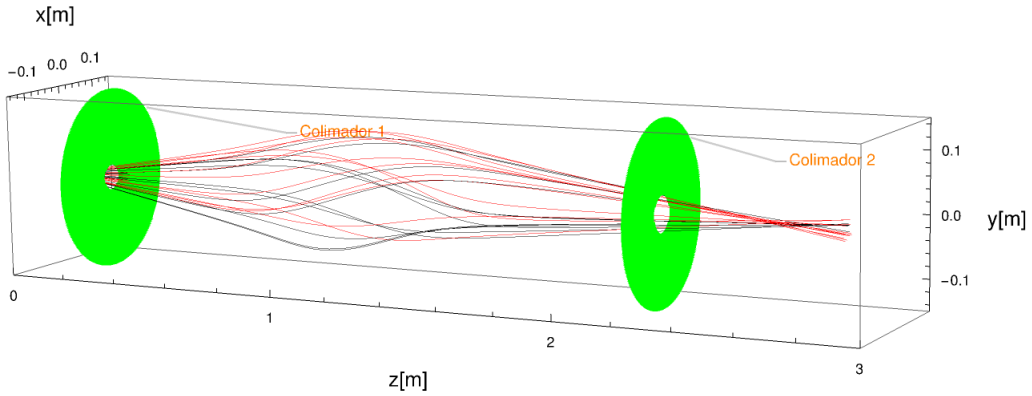
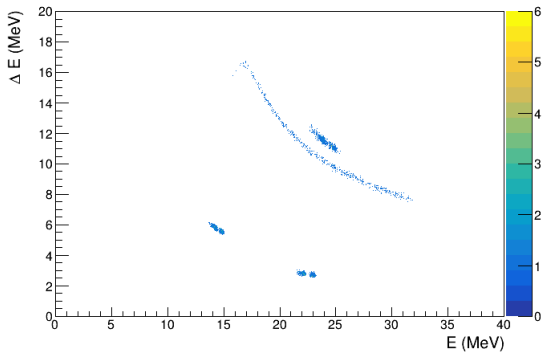
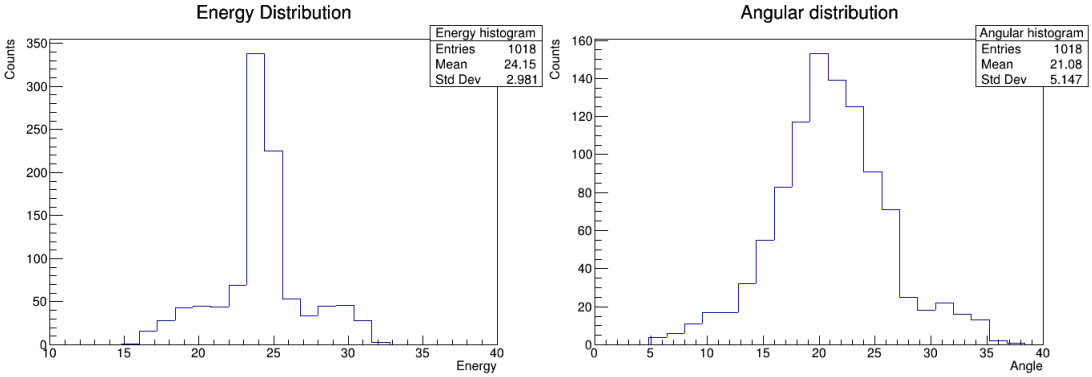


Figure 17: Charts made using wolfram.

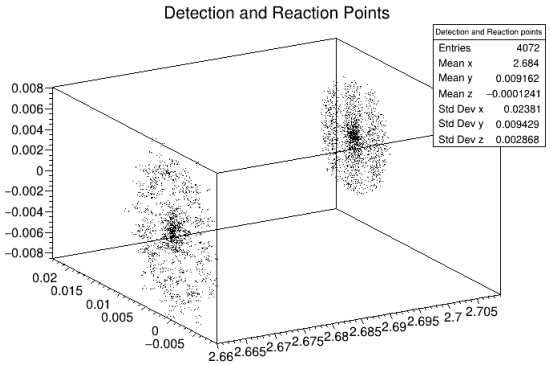
Now, the charts generated by the macro on root with the bi-parametric files.



(a) Bi-parametric.



(b) Histogram of energy of the ejectile particles. (c) angular distribution of the ejectile particles.



(d) Secondary target and detector (circular).

Figure 18: Charts made using root.

3.2 The ^{8}B production

The production o ^{8}B occurs through the reaction:

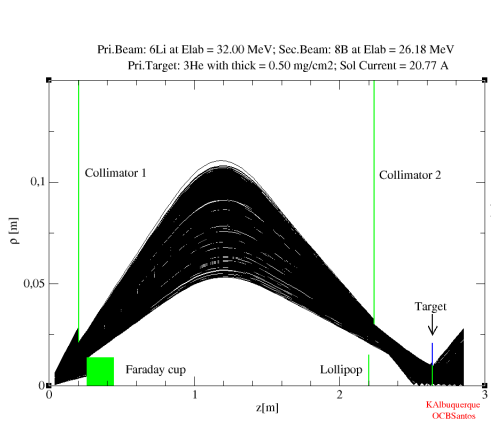
$$^{4}\mathrm{He}(^{6}\mathrm{Li},^{8}\mathrm{B})$$

The input file looks like this.

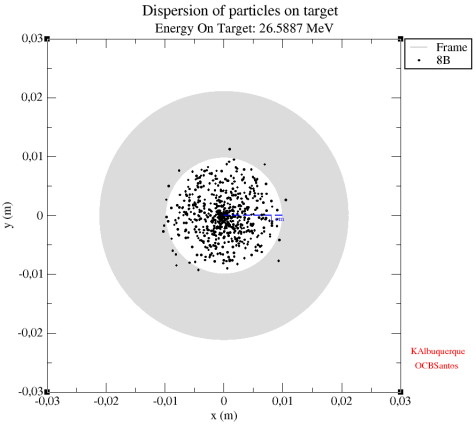
```
yes
1500    6Li     8B      3He     8           5          0.000   32    0.5
1       8       0.0005  0.04    0.02125 0.013625       0.0151  0.03
2.455   0.0
no
no
yes

6Li     6       3
7Be     7       4
4He     4       2
3He     3       2
3H      3       1
2H      2       1
END

yes
28Si    35      circular        0.55    -0.0015  0.07
8B      197Au   8B      197Au   0       0        -0.00037
END

6Li     197Au   6Li     197Au   0       0        -0.00037
7Be     197Au   7Be     197Au   0       0        -0.00037
4He     197Au   4He     197Au   0       0        -0.00037
3He     197Au   3He     197Au   0       0        -0.00037
3H      197Au   3H      197Au   0       0        -0.00037
2H      197Au   2H      197Au   0       0        -0.00037
END
```

Figure 19: Input file for example 1.

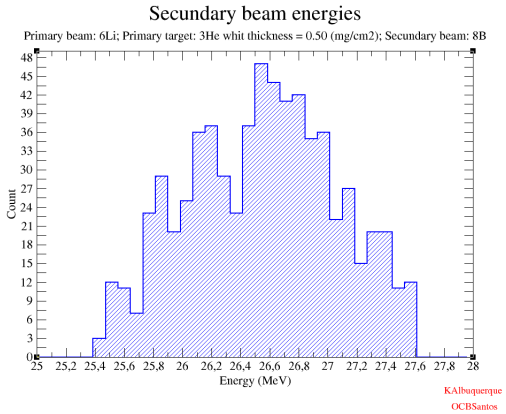
For the secondary beam, the output charts:



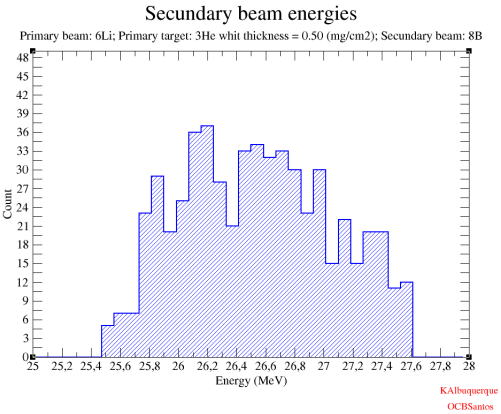
(a) Trajectories $\rho \times z$.



(b) Scatter on target.



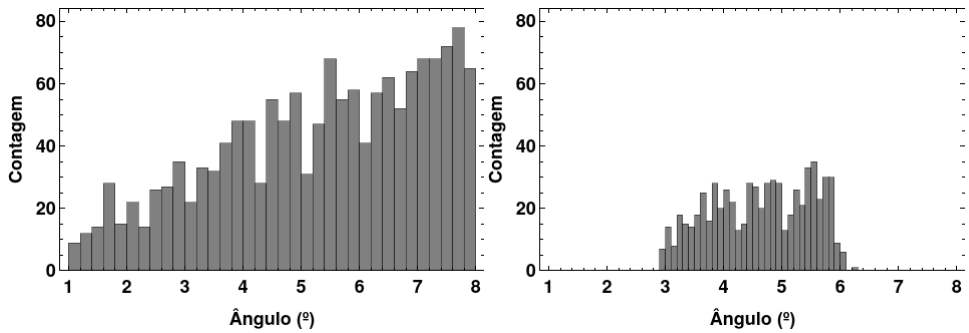
(c) Histogram of energy inside the solenoid.



(d) Histogram of energy on target

Figure 20: Charts automatically made by the program.

Using the wolfram mathematica to analyse the files in the *OutputFiles* folder, I represent here some following charts.



(a) Histogram of angles produced. (b) Histogram of the angles selected.

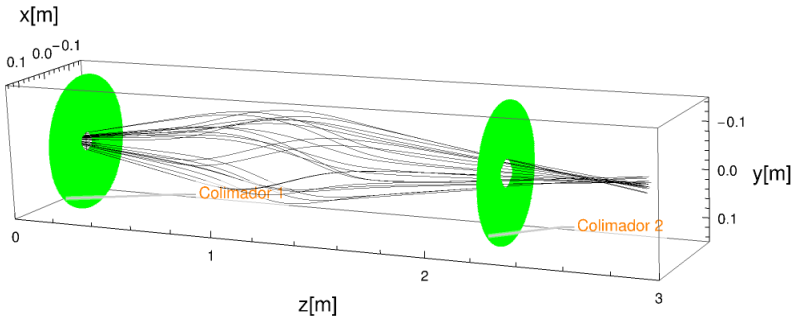
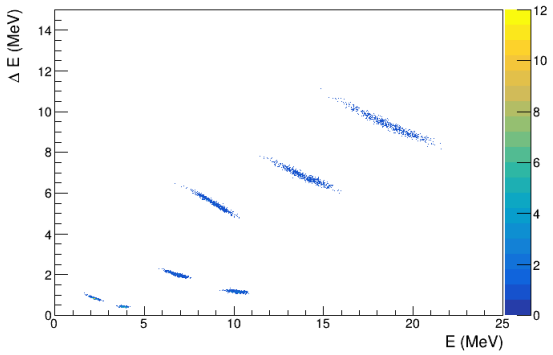
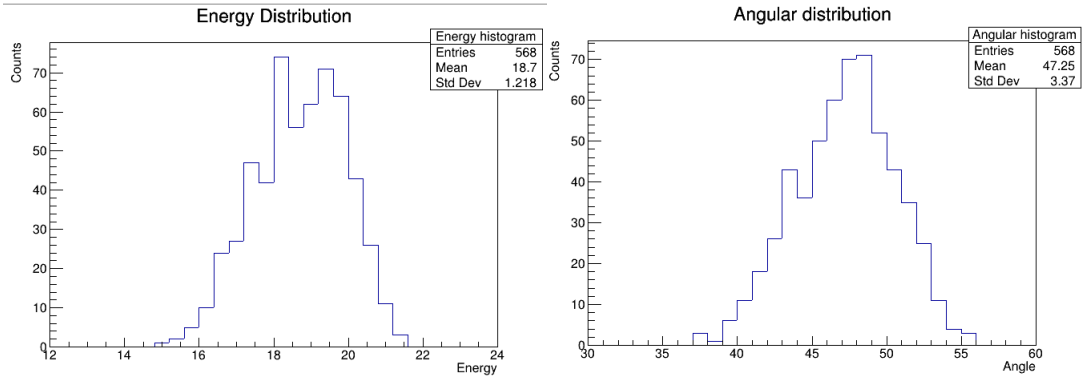


Figure 21: Charts made using wolfram.

Now, the charts generated by the macro on root with the bi-parametric files.



(a) Bi-parametric.



(b) Histogram of energy of the ejectile particles. (c) Angular distribution of the ejectile particles.

Figure 22: Charts made using root.