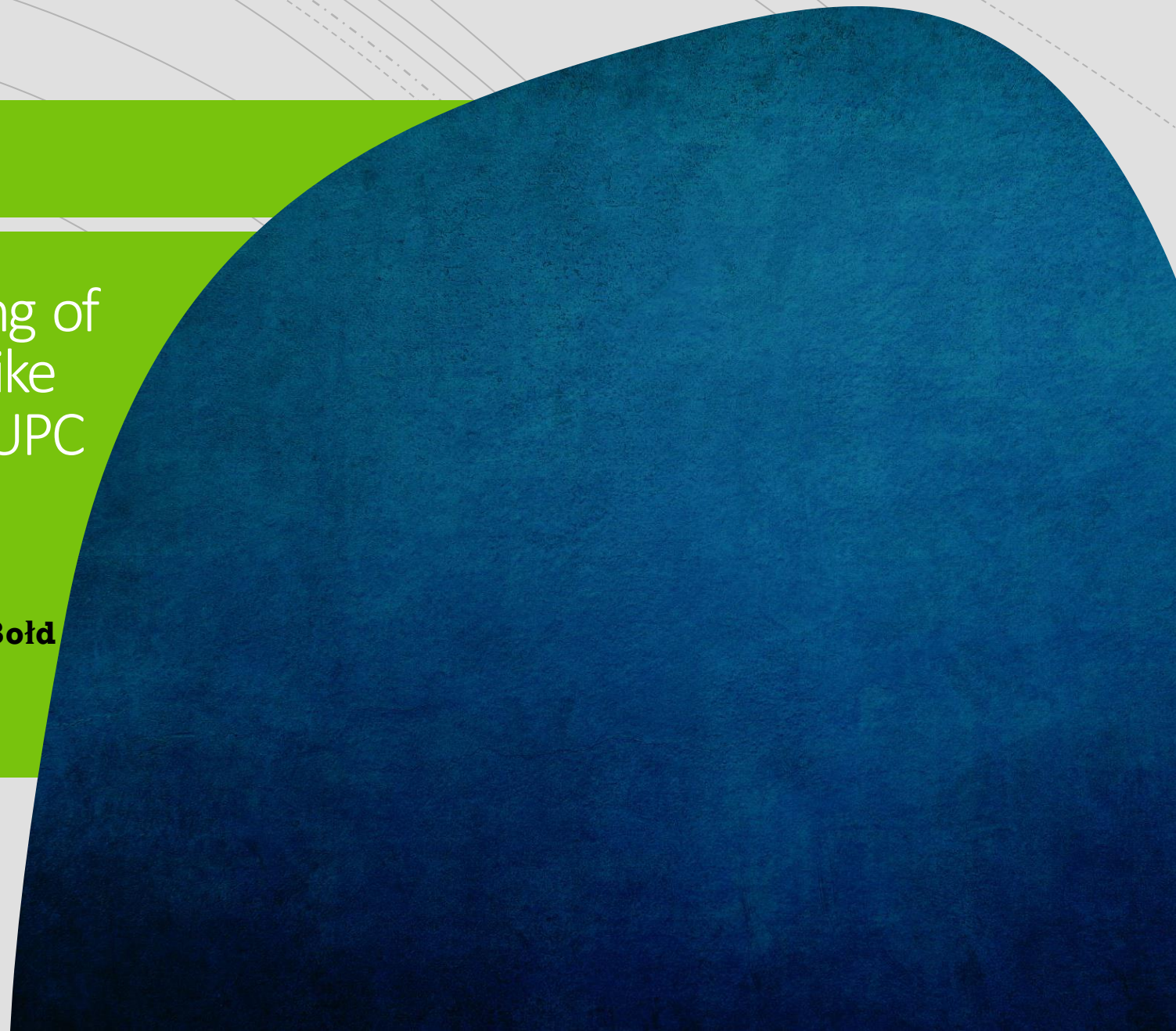
A method for searching of magnetic monopole-like tracks in ATLAS PbPb UPC collisions

Student : **Paweł Szczepaniak**
Supervisor : **dr hab. inż. Tomasz Bołd**



The main purpose

The aim of the project was to develop code in C++ to detect trajectories left by the magnetic monopole in the ATLAS detector through histogram representation of the detection points. In order to analyse the data we used the program named ROOT. Based on the resources taken and the information about the detection points, we were supposed to find the potential paths along which the particle moved.

A particle with magnetic monopole properties inside an external magnetic field would move just like a particle possessing electric charge inside a uniform electric field, i.e. with a parabolic motion.

The code

A simple ntuple analysis program (MakeClass) was developed to iterate over events and data inside the event.

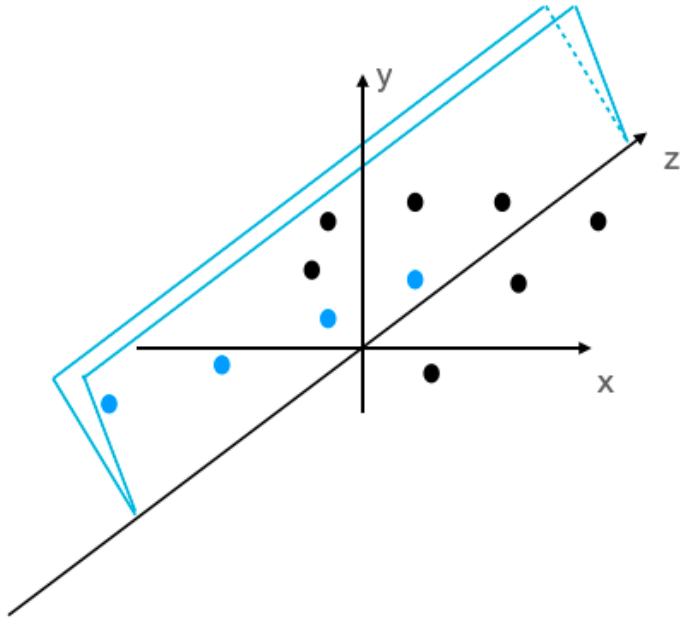
Getting started

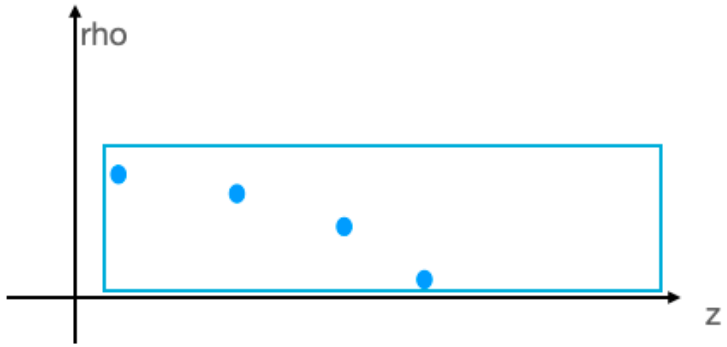
First step: print varaibles describing coordinates of hits in the ATLAS InnerDetector (x,y,z, phi)

```
44        for ( int i = 0; i < spX->size(); ++i )
45        {
46            double x = spX->at(i);
47            double y = spY->at(i);
48            double z = spZ->at(i);
49            double phi = TMath::ATan2(spY->at(i), spX->at(i));
50            std::cout << " " << x << " " << y << std::endl;
51            printf("Kąt pomiędzy OX = %lf\n", phi);
52               myHisto->Fill(phi);
53         }
```

Representation of spatial distribution

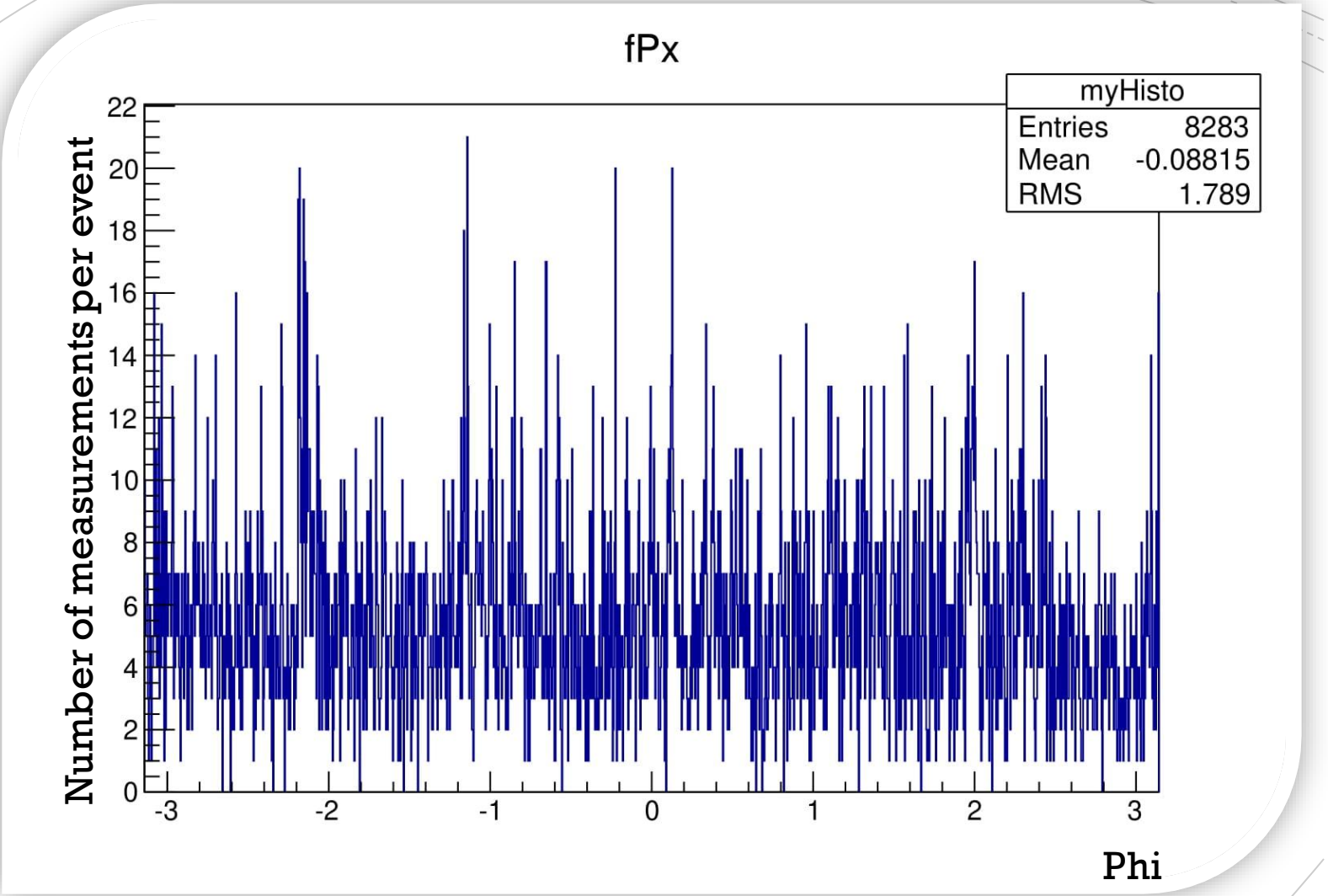
To represent the distribution of collisions in the detector chamber space, we used a histogram representation of the number of collisions as a function of the phi angle.





Angle division into ranges

In order to represent the number of detection points for a given angle phi, this angle is divided into an adequate ranges, called bins.



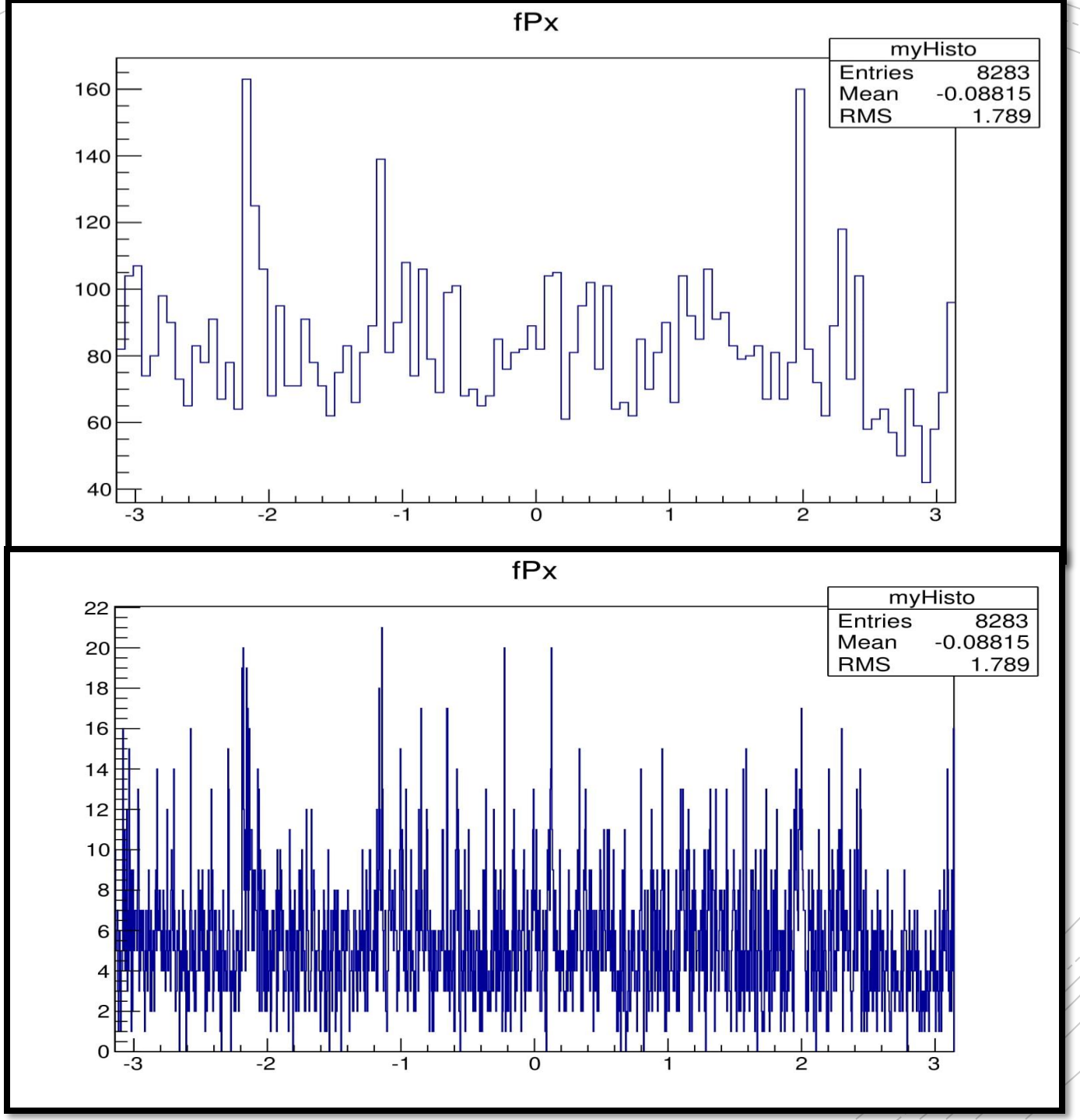
Setting the histogram resolution

I analysed the graphs, for different binning resolutions. Analysing these cases, I come to the conclusion that the optimum for accurately qualifying the number of counts to the appropriate angle would be between 1000-1500. Anything above 1500 stopped reducing the number of counts in peaks so drastically, whereas with too much binning, the graph actually becomes a blob.

```
26    TH1F *myHisto  = new TH1F("myHisto","fPx", 1500, -TMath::Pi(), TMath::Pi());
```

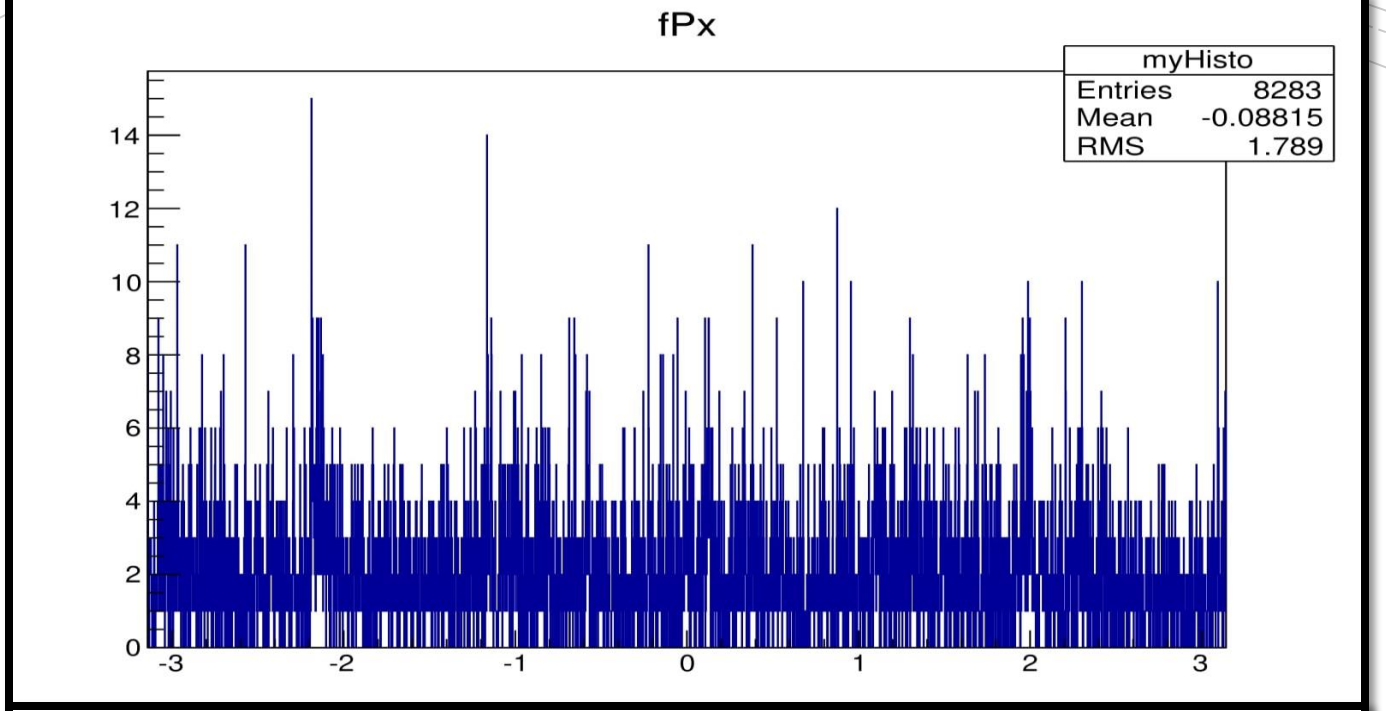
Underscaling of the bins amount

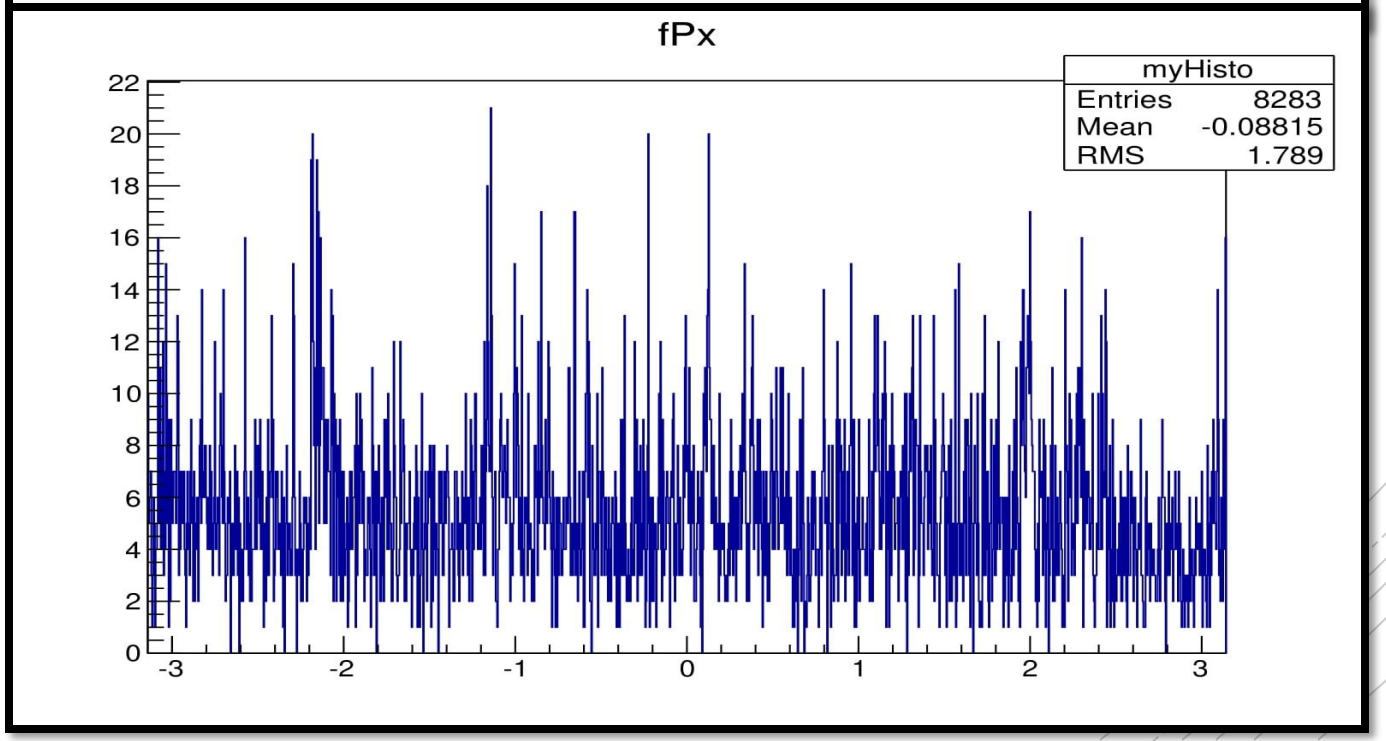
Dividing the X-axis into too few intervals results in a coarse graph, as the width of each bin tolerates the categorisation of too many particles into a particular angle bin.



Overscaling of the bins amount

The use of a histogram with too many bins results in a graph with contradictory data, as one detection can be classified into two adjacent angle bins and thus lost.





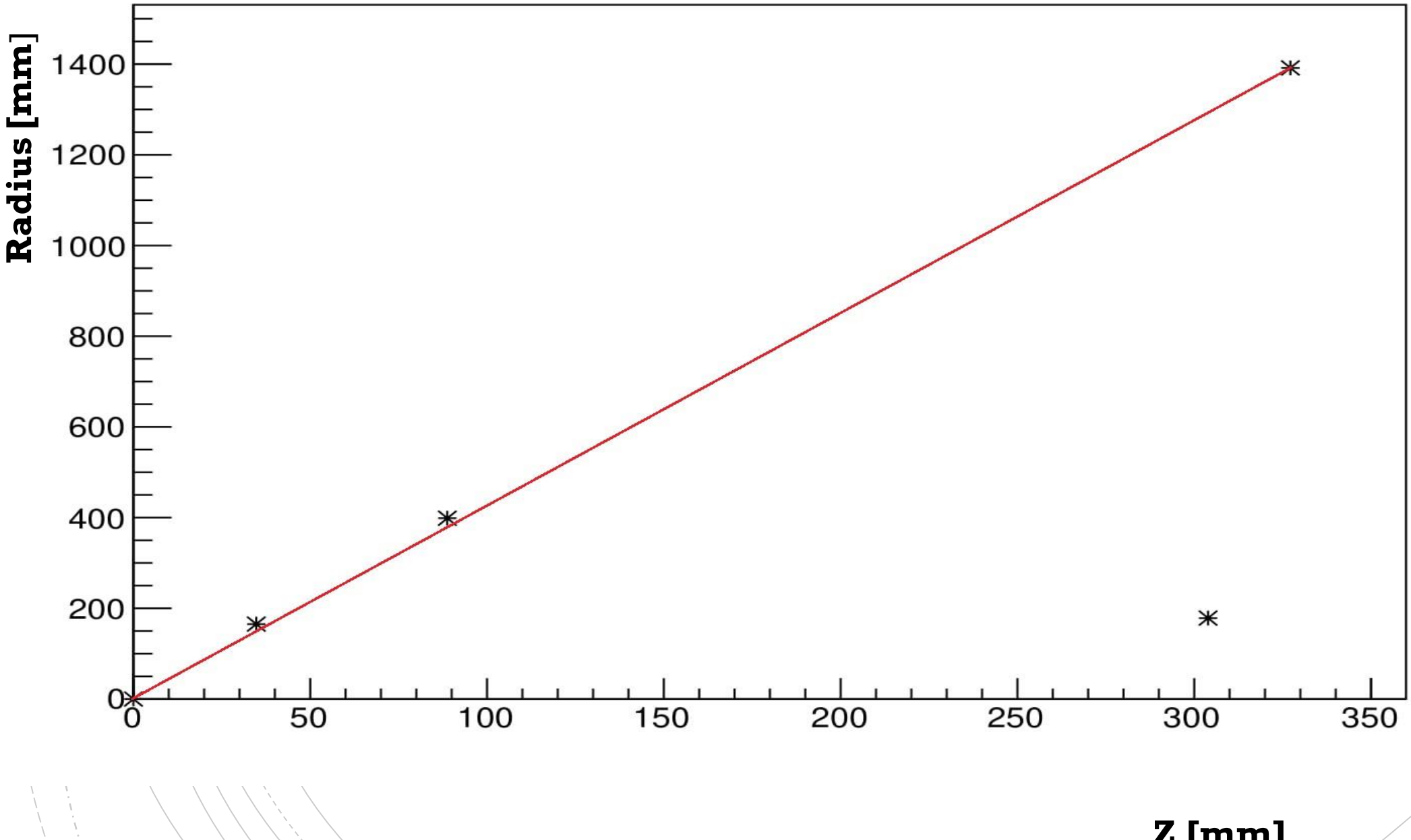
Extracting information on bins

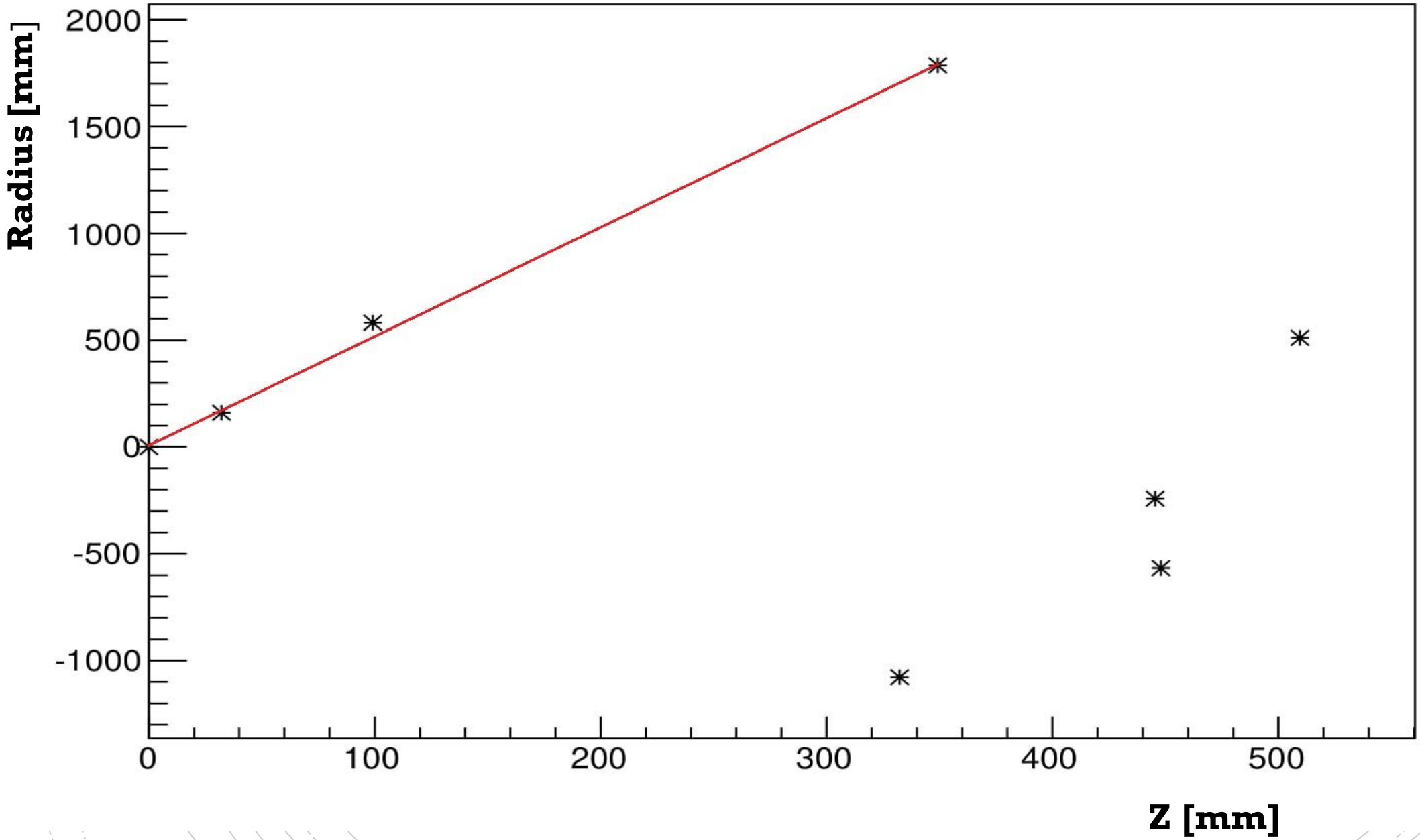
- After dividing the axis of angles into bins, selected phi bins that have 7<N<12 entries (== one particle +noise).

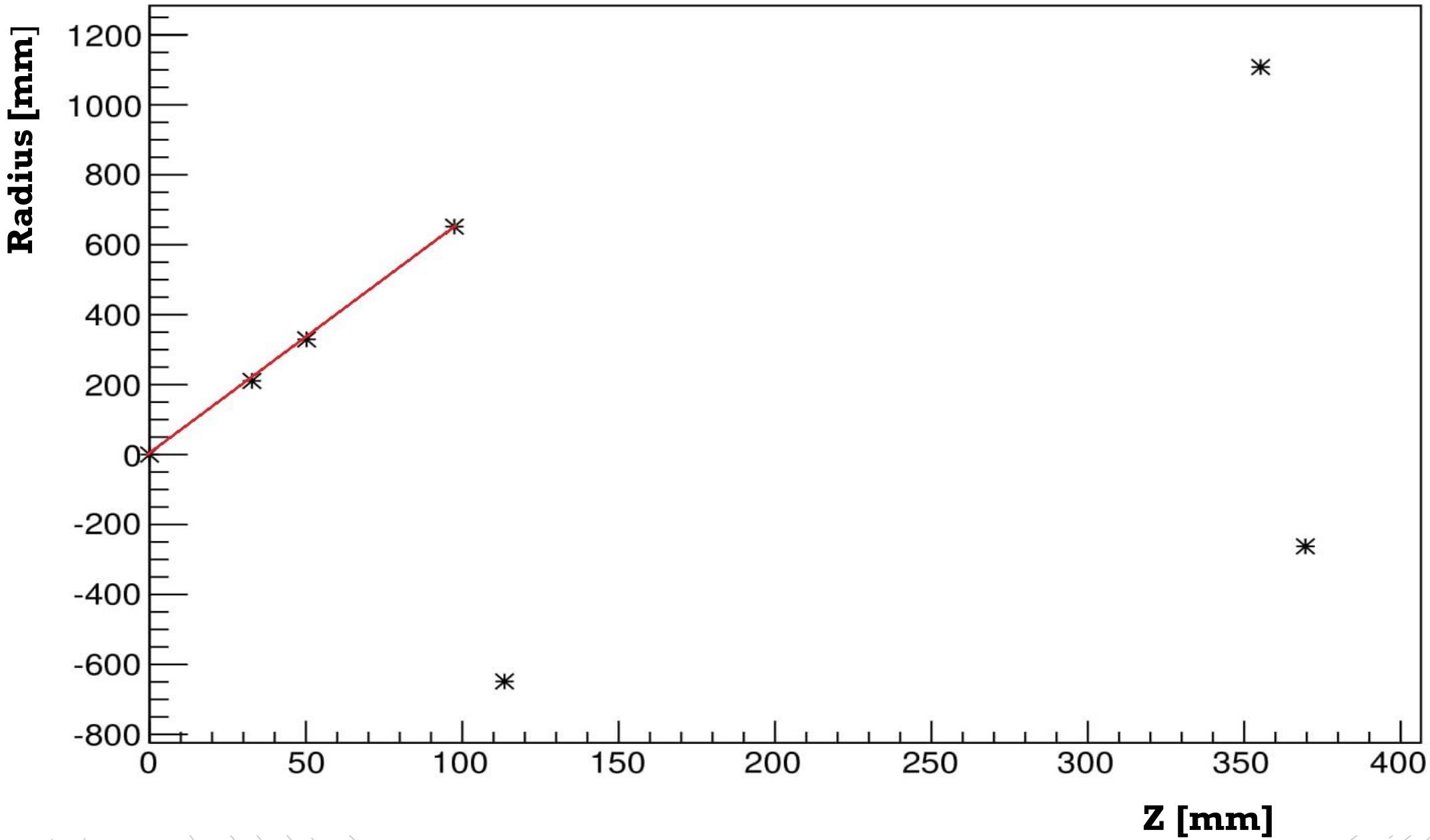
```
55      for ( int phibin = 1; phibin <= myHisto->GetXaxis()->GetNbins(); ++phibin )
56          {
57           if ( myHisto->GetBinContent(phibin) > 7 && myHisto->GetBinContent(phibin) < 12 ) // liczba punktów na wykresie
58           {
59             double min = myHisto->GetXaxis()->TAxis::GetBinLowEdge(phibin);
60             double max = myHisto->GetXaxis()->TAxis::GetBinUpEdge(phibin);
61            // printf ("--- Znaleziono bin %d ----\nmax=%lf\nmin=%lf\n", phibin, max, min);
62            // printf("Ilość punktów na bin = %d\n", myHisto->TAxis::GetBinContent(phibin));
```

High momentum particles

We were able to distinguish characteristic arrangements of measurement points in a straight line.
This are candidates for a high momentum particles (no bending in phi), as the curving effect of the magnetic field is the less observable the higher the momentum of the particle is.


Radius [mm]
1400
1200
1000
800
600
400
200
0
0
50
100
150
200
250
300
350
Z [mm]


Radius [mm]
2000
1500
1000
500
0
-500
-1000
0
100
200
300
400
500
Z [mm]

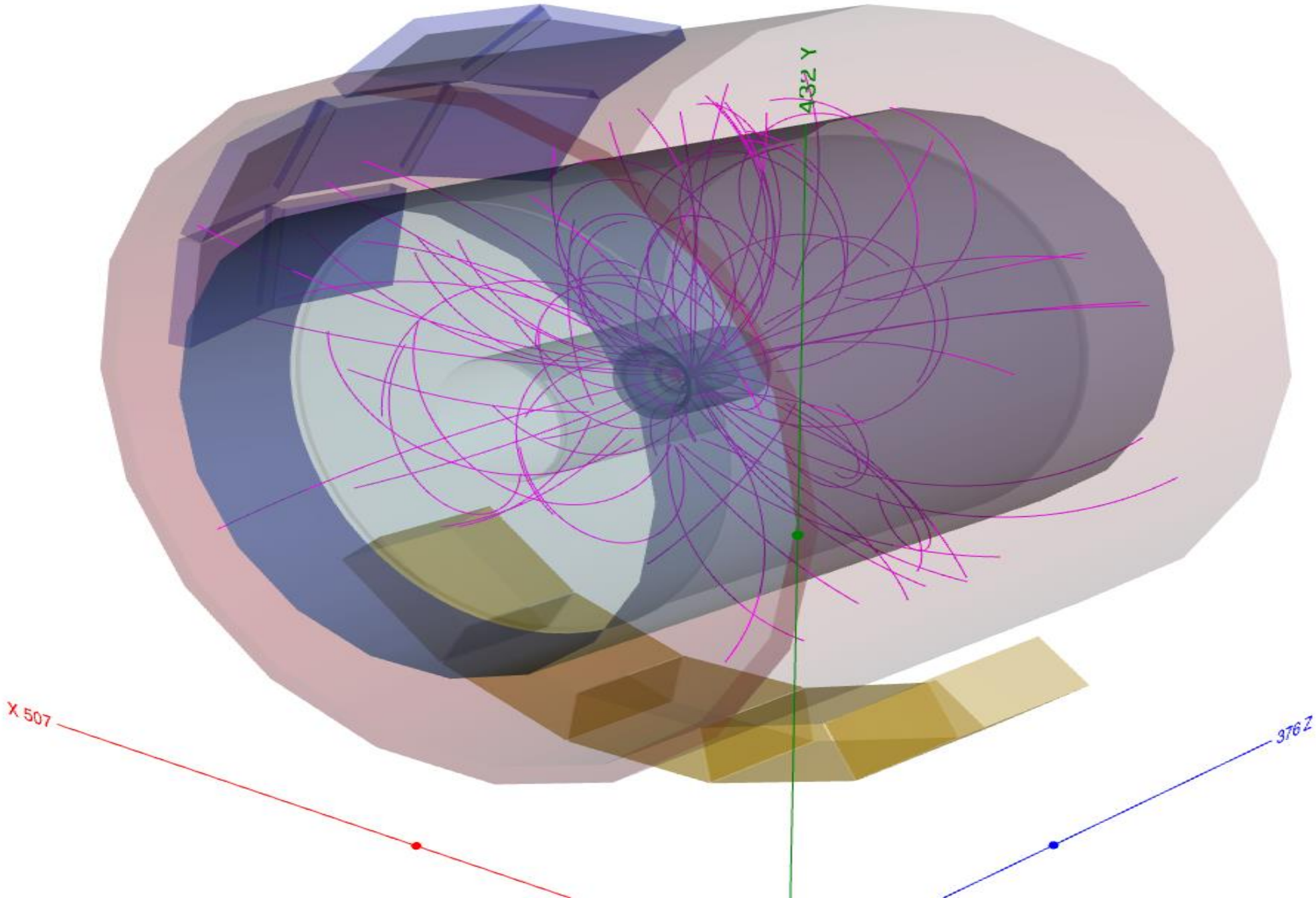

Radius [mm]
1200
1000
800
600
400
200
0
-200
-400
-600
-800
0
50
100
150
200
250
300
350
400
Z [mm]

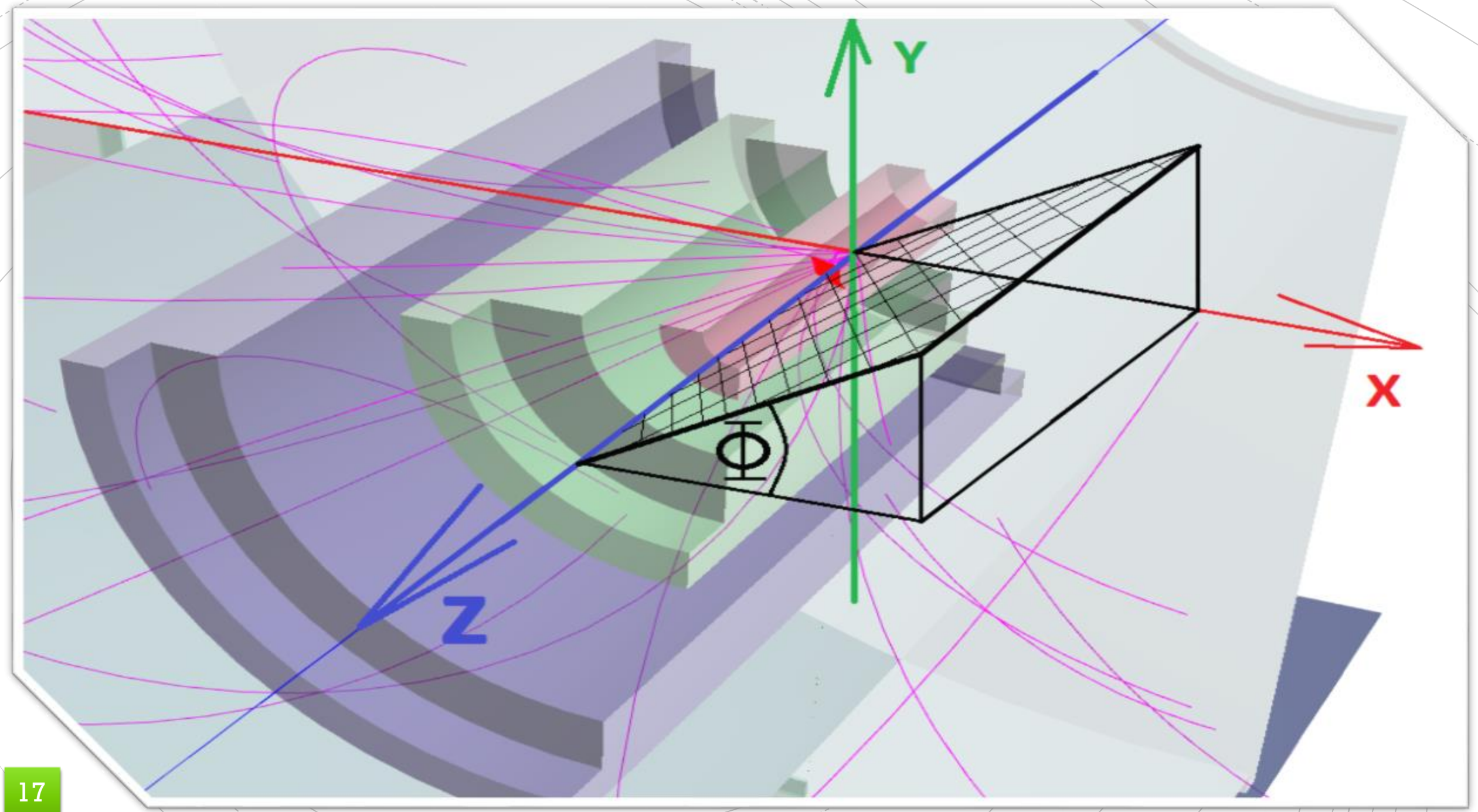
Conclusion

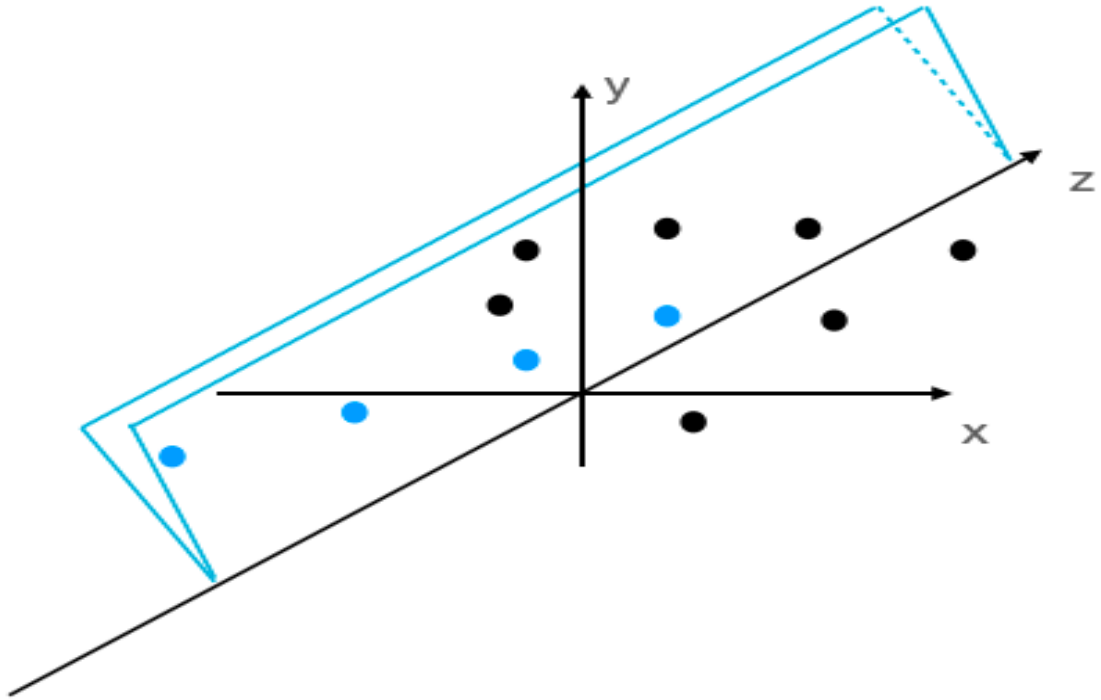
I learned about the data analysis using ROOT (first c++ experience)
and analysed small sample of ATLAS experiment data
The simple algorithm was not able to detect what we looked for

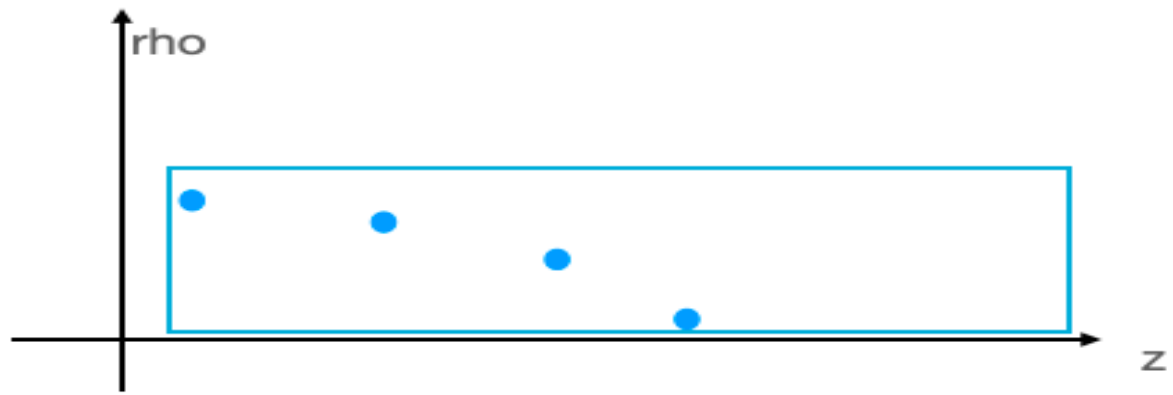
But it seems to work for a control objects (high momentum particles)

)


432 Y
X 507
376 Z


Y
X
Z
Φ


y
z
x


rho
z