



Ionization process simulation in gas

Goal : Simulation/parameterization of ionization cluster generation in Geant4

To investigate the potential of the Cluster Counting technique (for He based drift chamber) on physics events a reasonable simulation/parameterization of the ionization clusters generation in Geant4 is needed.

Garfield++ :

- (Heed) simulates the ionization process in the gasses (not only) in a detailed way.
- (Magboltz) computes the gas properties (drift and diffusion coefficients as function of the fields value)
- solves the electrostatic planar configuration and simulates the free charges movements and collections on the electrodes.

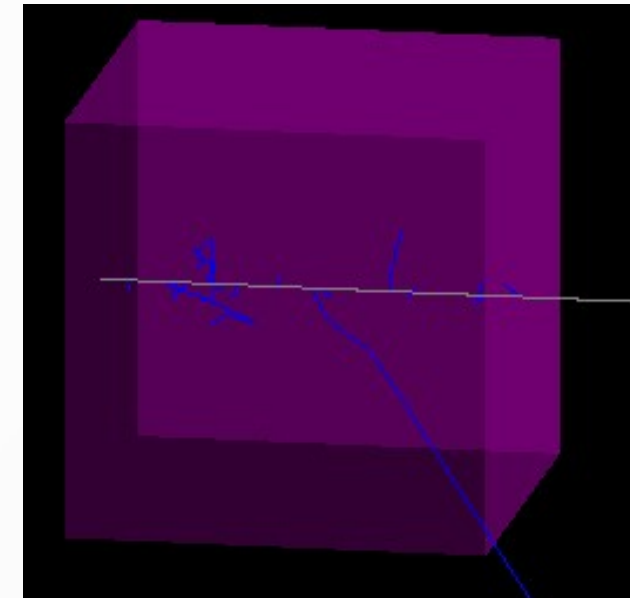
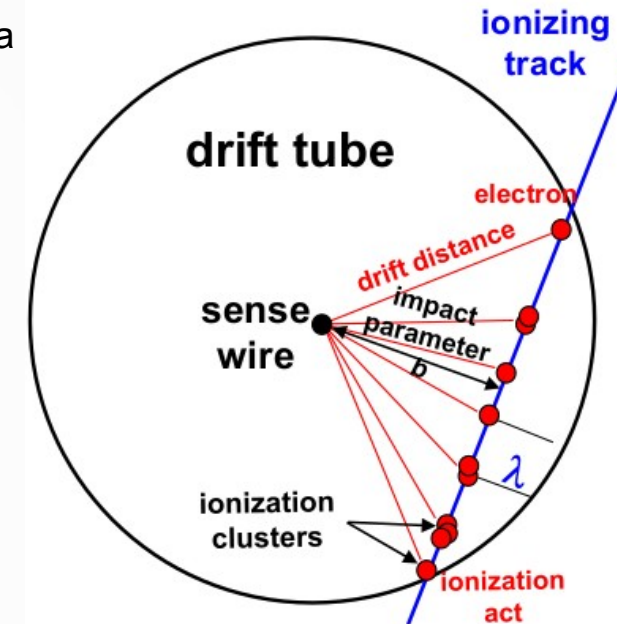
So Garfield can study and characterize the properties and performance of single cell or drift chamber with simple geometry, but cannot simulate a full detector neither study collider events.

Geant4 :

- Simulates the elementary particle interaction with material of a full detector.
- Studies colliders events

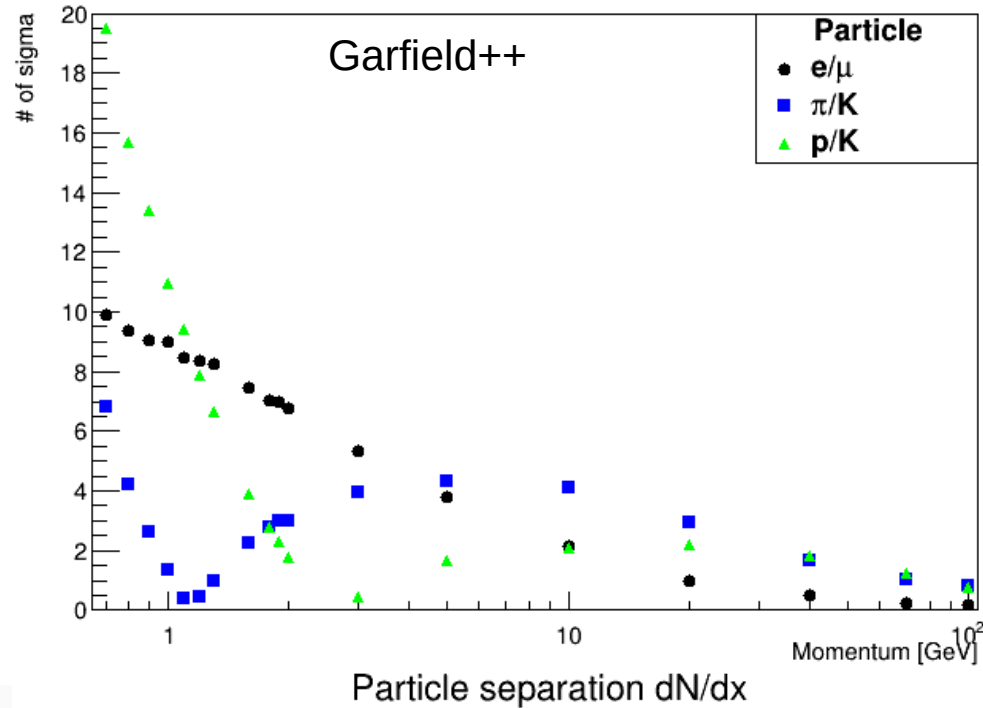
But...the fundamental properties and performance of the sensible elements (drift cells) have to be parameterized or ad-hoc physics models have to be defined.

Actually we are simulating 2m long tracks which pass through a 1 cm long side box of 90% He and 10% iC_4H_{10} , with Garfield++ and Geant4 .

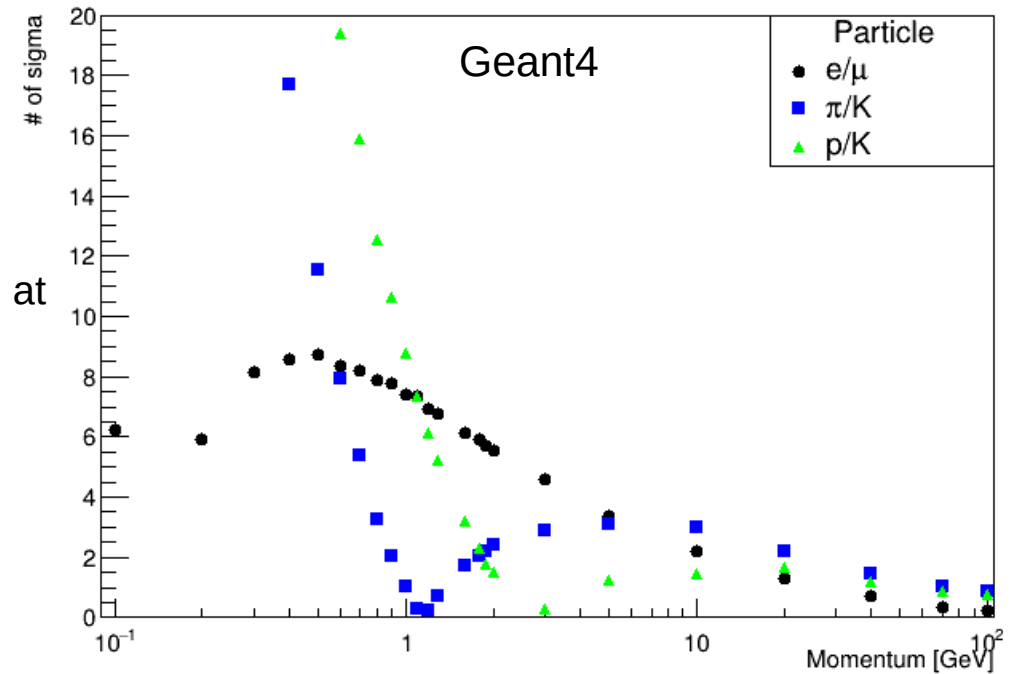


Particle separation with traditional dE/dx method and cluster counting

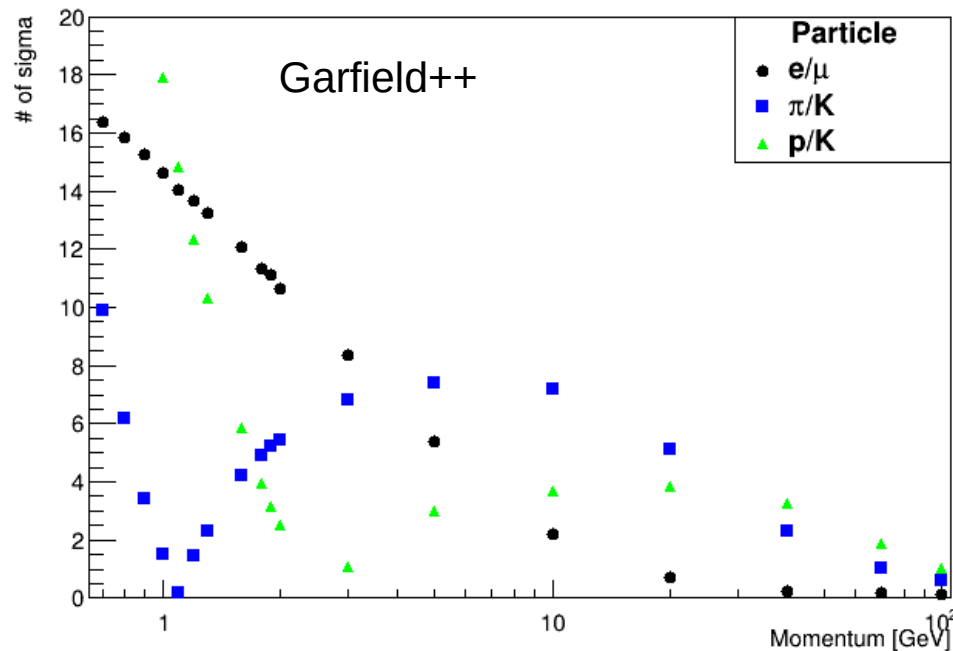
Particle separation from truncated mean dE/dx



Particle separation from truncated mean dE/dx



Truncated mean at 70%

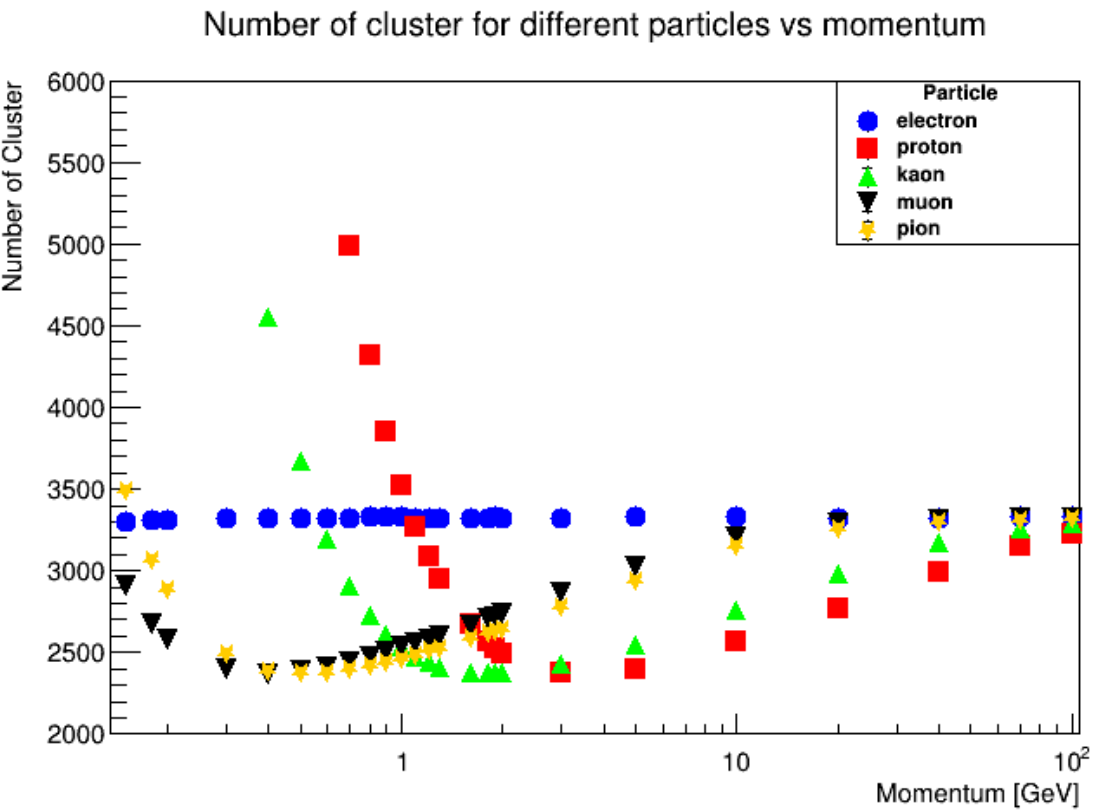


$$n_{\sigma} = \frac{\Delta_A - \Delta_B}{\langle \sigma_{A,B} \rangle} \cdot \langle \sigma_{A,B} \rangle \text{ is the average of the two resolutions.}$$

Cluster counting leads to an **improvement** on particle separation power. As example, around 5 GeV the power separation of a pion from kaon obtained with traditional method is about 4, the one obtained with cluster counting is around 8.

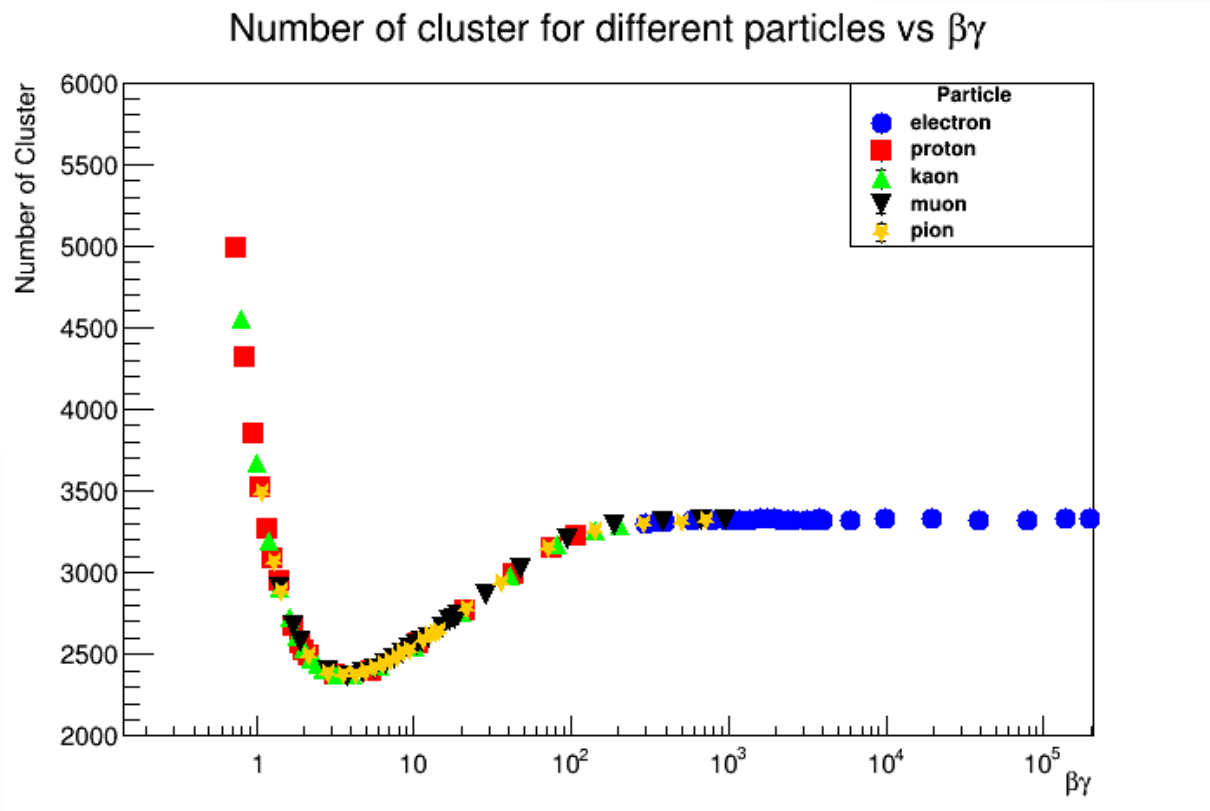
We are simulating 2m long tracks which pass through a 1 cm long side box of 90% He and 10% iC₄H₁₀, with Garfield++ and Geant4

Studying the results from Garfield++ simulations, we can interpret correctly the results obtained from Geant4 simulations with the goal of **reconstruct the number of clusters and the cluster size** generated from different particles with different momenta passing through the tracker detector.

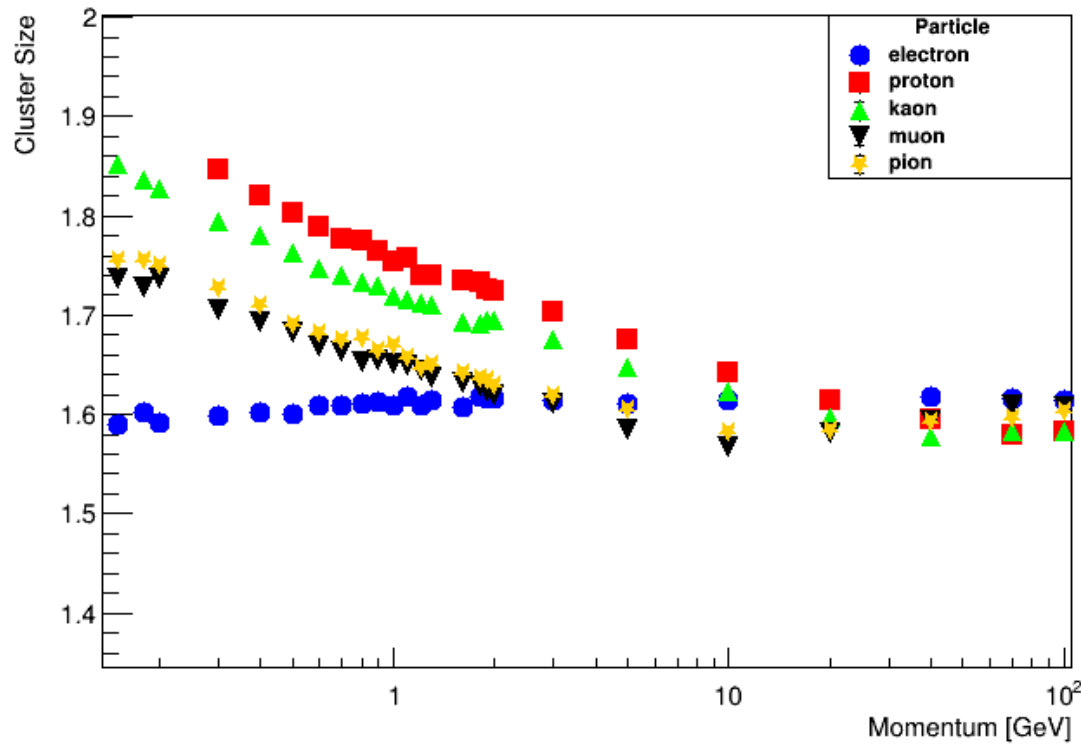


Number of cluster from
Garfield++

Here the distribution of number of cluster produced by different particle at different momenta, obtained with Garfield++



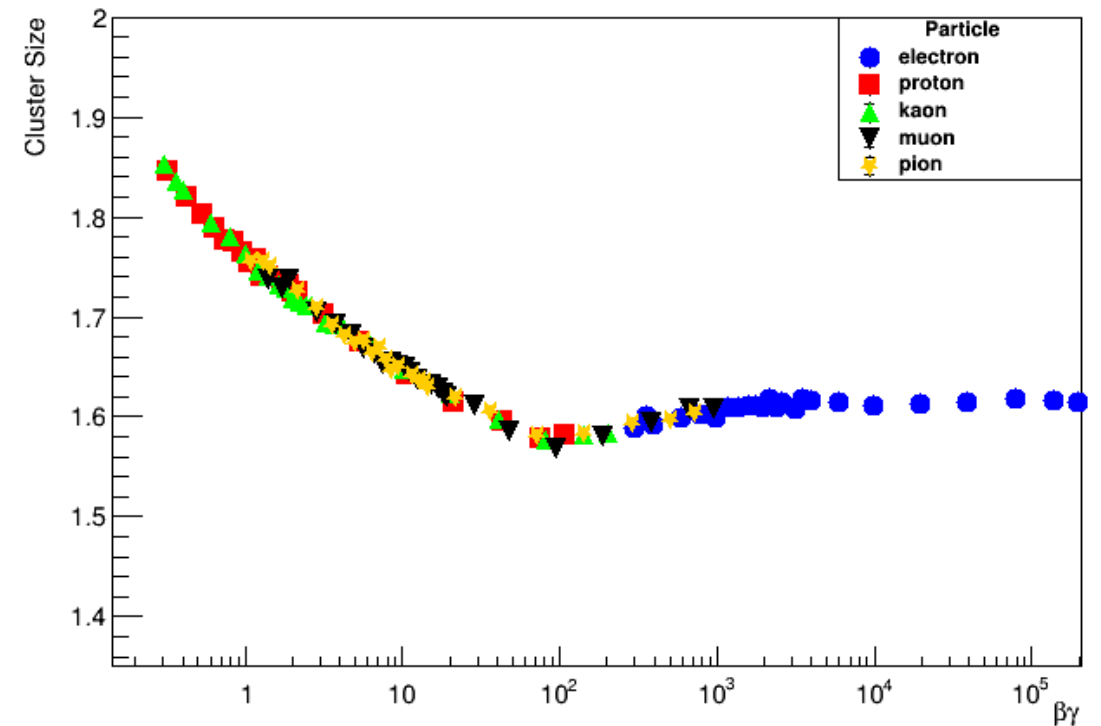
Cluster Size vs momentum



Cluster size
from Garfield++

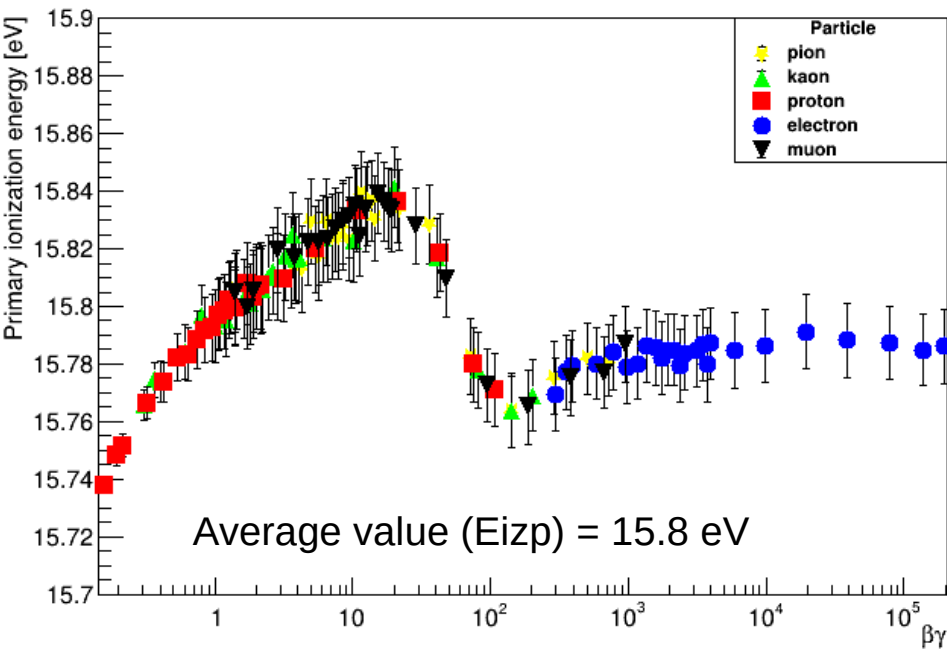
Here the distribution of cluster size produced by different particle with different momenta, obtained with Garfield++

Cluster Size vs $\beta\gamma$

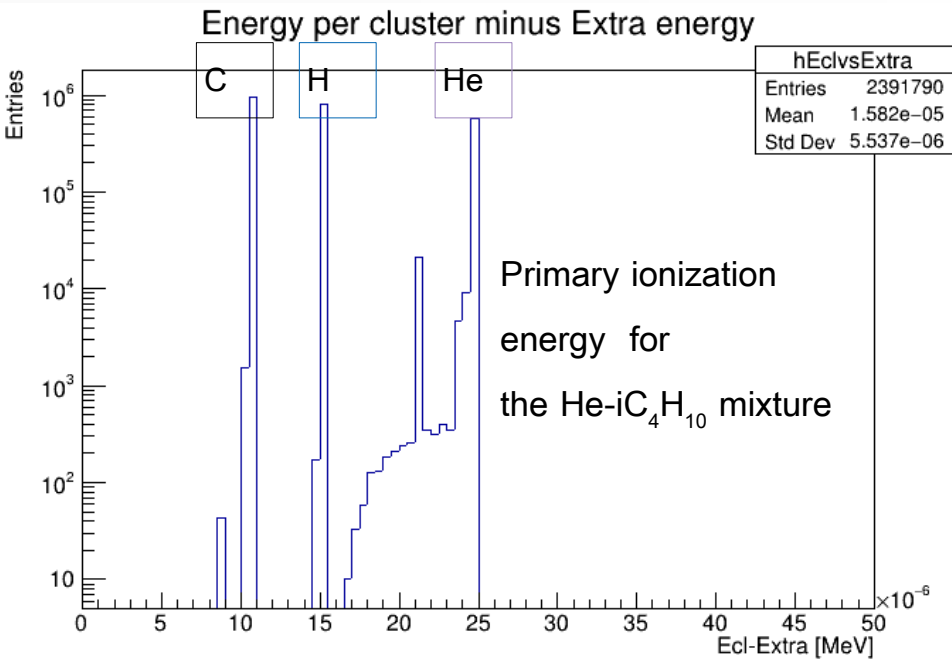
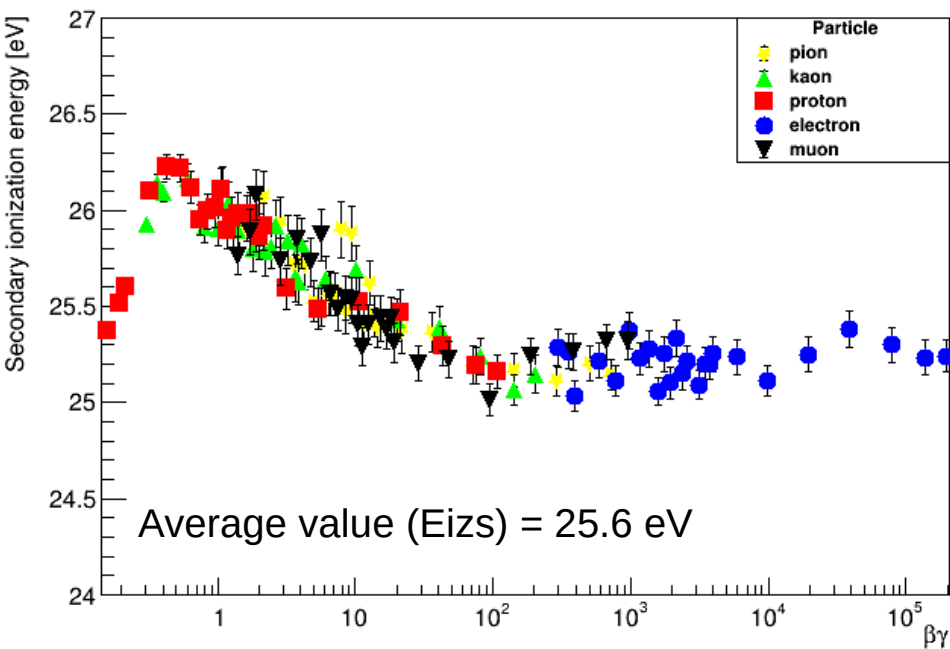


Primary and secondary ionization energy

Primary ionization energy



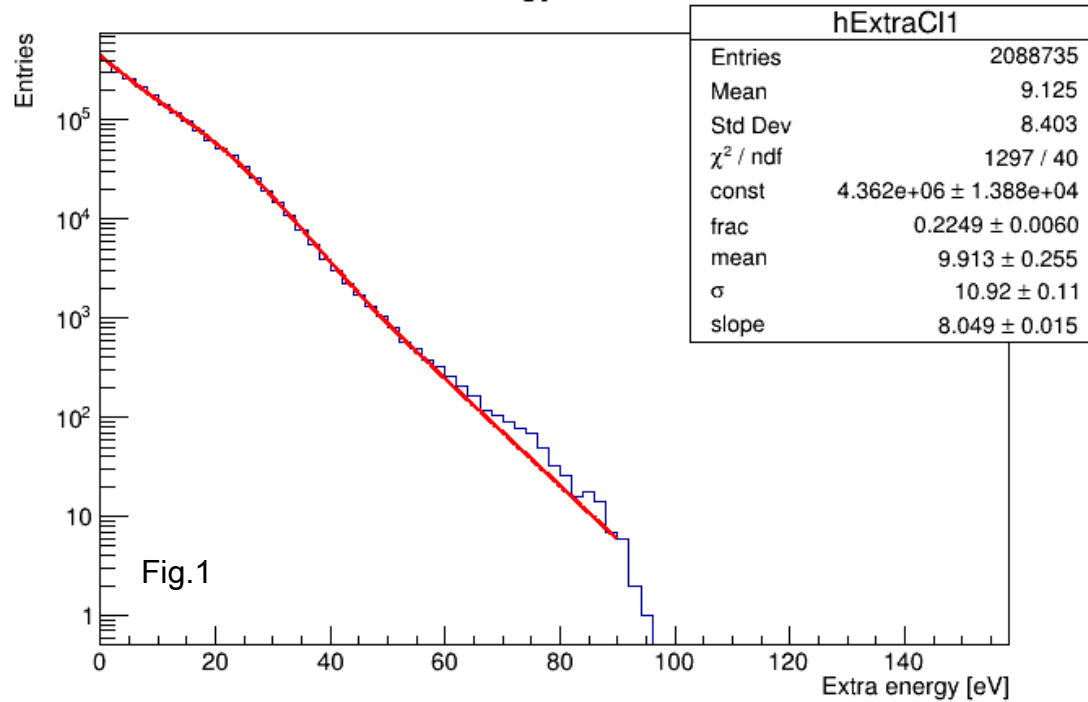
Secondary ionization energy



Primary ionization energy indicates the total energy loss by the track minus the total extra energy over number of cluster .

Secondary ionization energy indicates the ratio of extra energy over the cluster size simulated by Garfield++.

Extra energy with ClSz=1



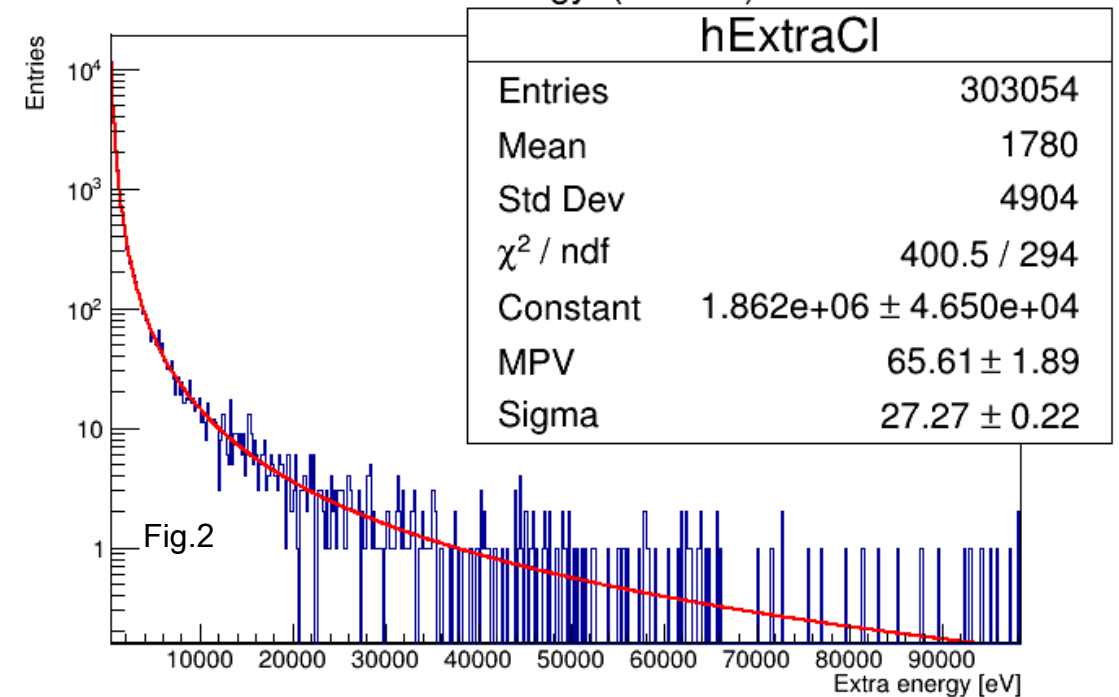
This is the kinetic energy distribution for cluster with cluster size equal to 1.

The fit is the sum of an exponential function plus a Gaus function.

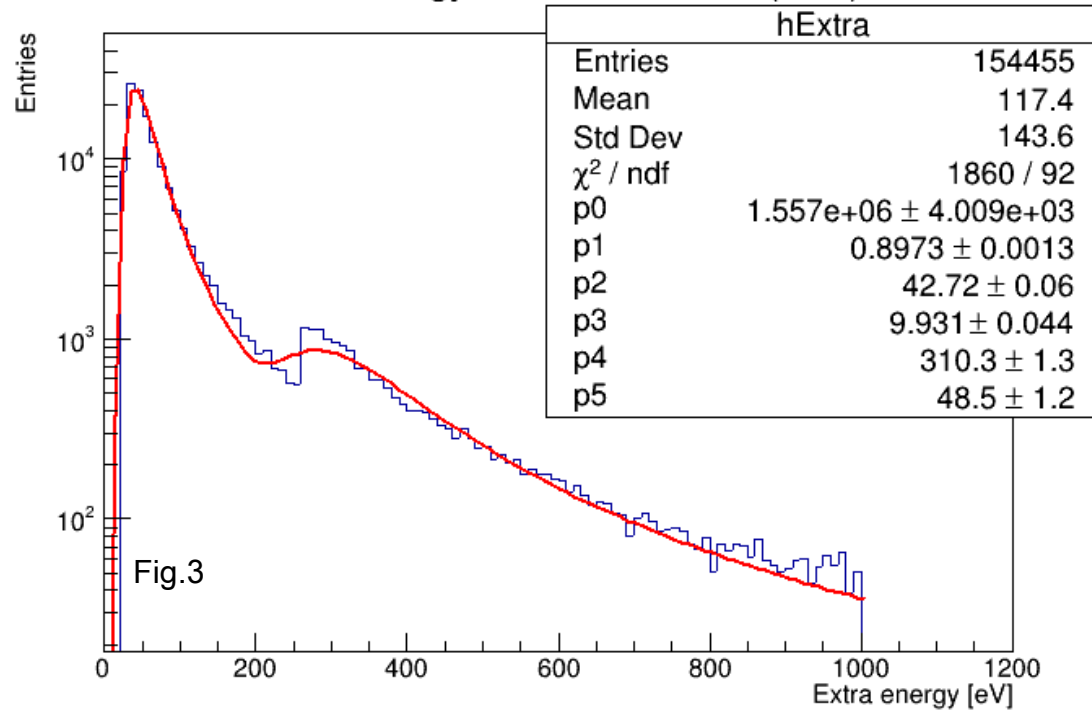
This is the kinetic energy distribution for cluster with cluster size higher than 1.

The fit is performed with a Landau function.

Extra energy (ClSz>1)



Extra energy Cl>1 first Cluster (Gen)



This is the kinetic energy distribution for cluster with cluster size higher than 1 up to the energy cut of delta rays (maxCut)

The fit is performed with a double landau function.
The total kinetic energy reconstructed for cluster with cluster size higher than 1 is a convolution of n times (number of cluster) of this distribution.

NOTE

maxCut is set to 1 keV. If maximum kinetic energy of particle is less than 1 keV, maxCut is set to the value of the maximum kinetic energy . It is the energy equivalent to the range cut set in Geant4.

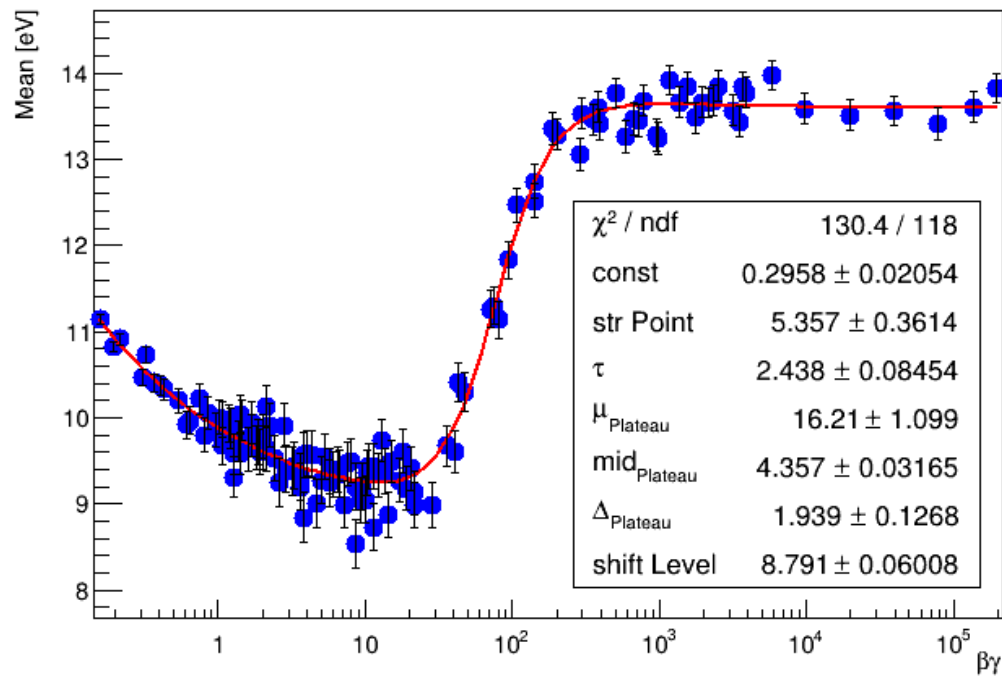
The **goal** is to create two different models for kinetic energy of clusters with cluster size equal to 1 and higher than 1, to interpret correctly the total energy loss by different particles in a single cell.

We implemented the fit in slides 7 and 8 for different particles at different momenta and studied the fit results.

Moreover we evaluated the probability to have cluster size equal to one respect to have cluster size higher than one.

Fit parameters for kinetic energy distribution for cluster with cluster size equal to 1

Mean from gaus+exp fit of Extra Energy with CISz=1



Other parameters studied are:

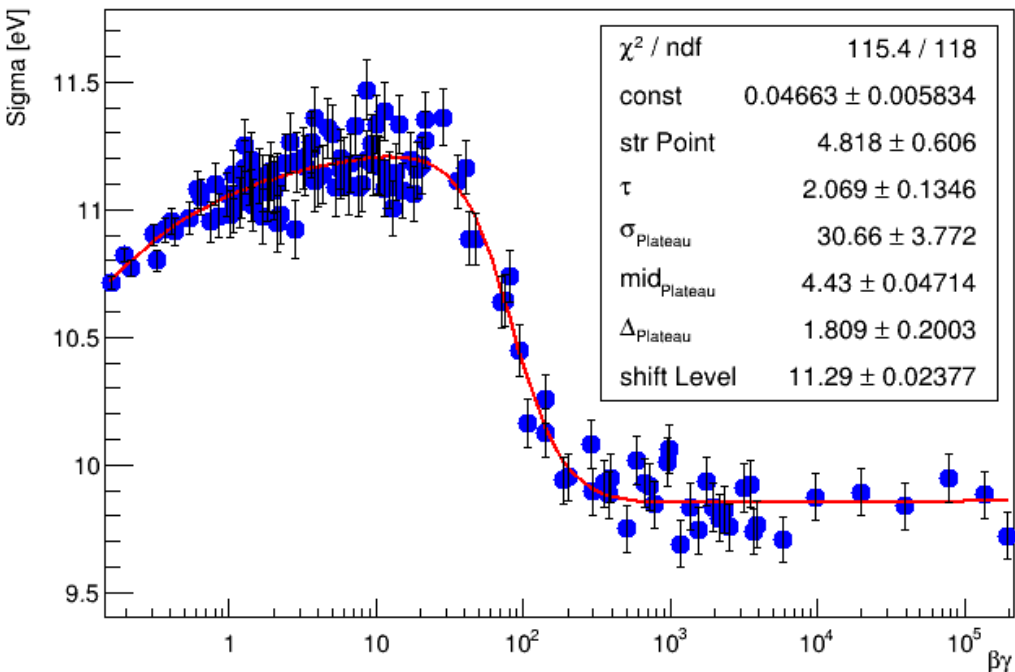
- Slope and fraction for exp+gaus fit
- Most probable value and sigma for landau fit
- Rt value, evaluated as the ratio between the clusters containing a single electron and the clusters containing more than an electron.
- The two MPV and sigma for double landau fit (in progress).

Here the distribution of mean value (top left) and sigma value (bottom right) for different particles with different momenta.

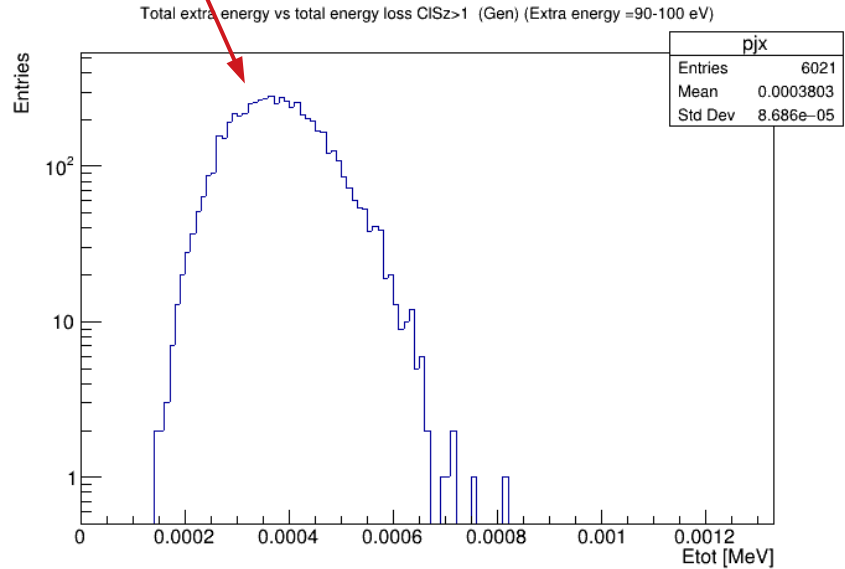
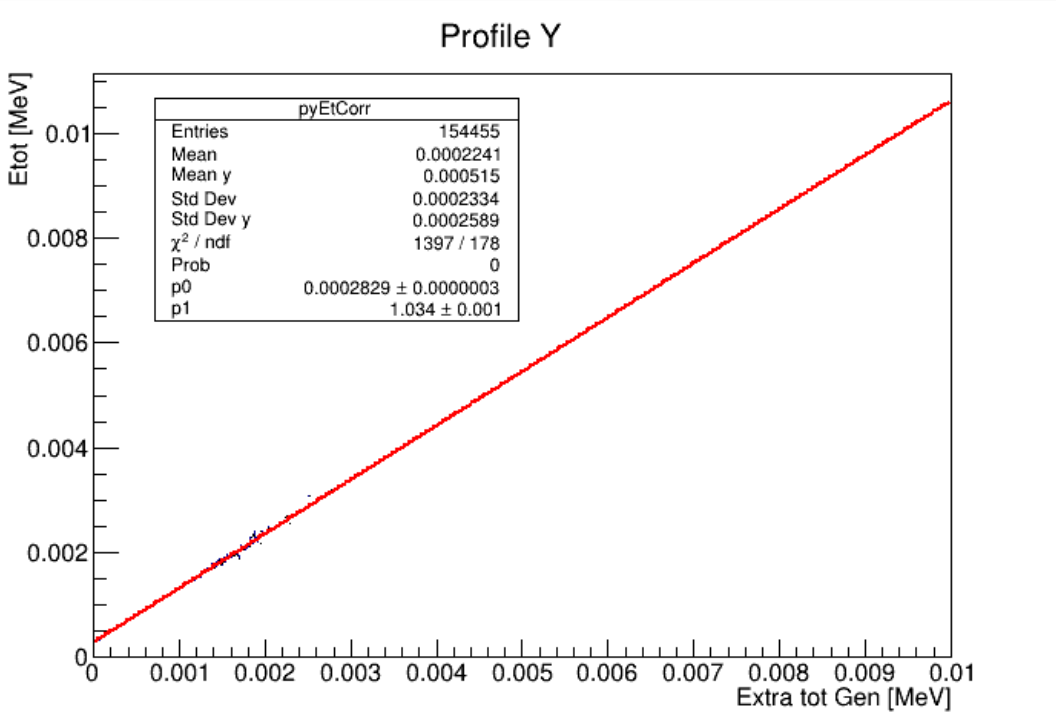
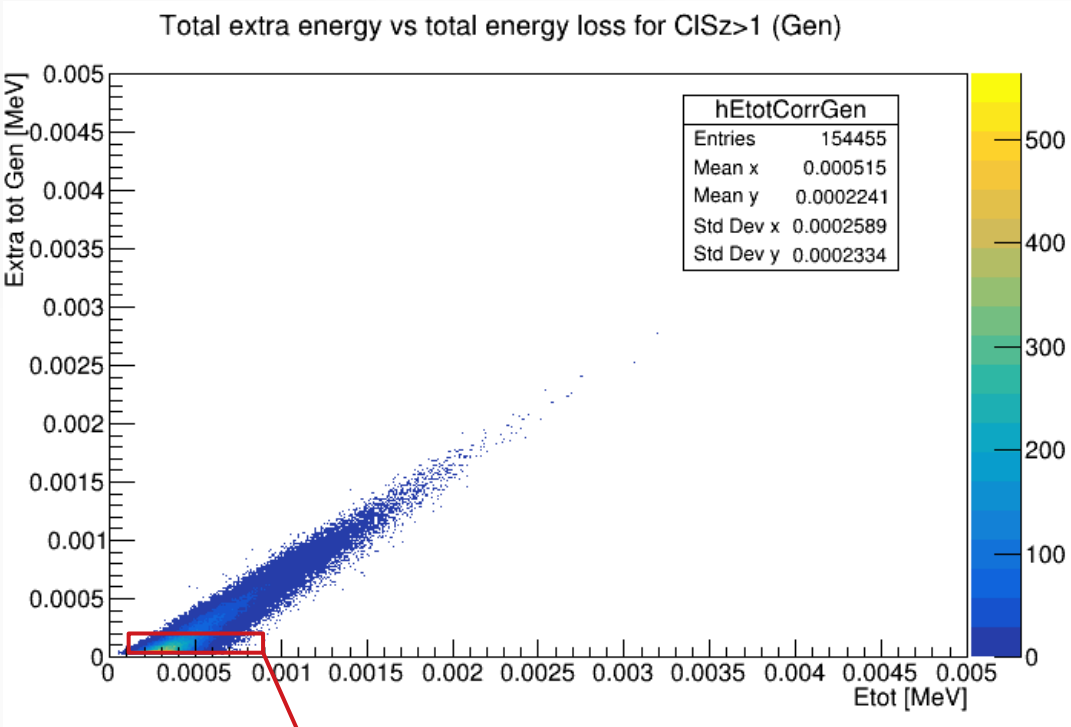
The distributions are fitted with an exponential function plus an efficiency function.

$$Eff = \frac{E f f_{\text{plateau}}}{1 + 81 \frac{V_{1/2} - V}{\Delta_{10\%}^{90\%}}}$$

Sigma from gaus+exp fit of Extra Energy with CISz=1



First step for all the algorithms tried is the evaluation of the total kinetic energy spent to create clusters with cluster size higher than 1, identifying the value of “maxExEcl”.



$$\text{maxExEcl} = (\text{Etot} - \text{maxEx0} + g\text{Random} \rightarrow \text{Gaus}(0, \text{ExSgm})) / \text{maxExSlp}$$

MaxEx0 is the first parameter of the linear fit

MaxExSlp is the second parameter of linear fit

ExSgm is the average of the sigma of each point in the correlation trend.

The figure shows an example of distribution of total energy loss for extra energy between 90 and 100 eV for cluster with cluster size higher than one.

Reconstruction algorithms

We implemented seven different algorithms trying to reproduce the number of cluster and the cluster size.

The first step common to all algorithm is the evaluation of the total kinetic energy for cluster with cluster size higher than one (maxExEcl) event by event.

1) The first algorithm uses a reference value of the ratio between clusters containing a single electron and clusters containing more than one electron (R_t). Using the R_t value, the algorithm chooses to create cluster with cluster size one or higher. Then, it assigns the kinetic energy to each cluster by using the proper distributions. If the cluster has more than one electron, a check on the total kinetic energy is performed and its cluster size is evaluated. The procedure is repeated until the sum of primary ionization energy and kinetic energy per cluster saturate the energy loss of the event.

2) The second algorithm, if maxExEcl is higher than zero, generates the kinetic energy for clusters with cluster size higher than one by using its distribution and evaluates cluster size. This procedure is repeated until the sum of primary ionization energy and kinetic energy per cluster saturate the maxExEcl of the event.

Then, using the remaining energy ($E_{\text{loss}} - \text{maxExEcl}$), the algorithm creates clusters with cluster size equal to one by assigning their kinetic energy according to the proper distribution.

The reconstruction of clusters with cluster size equal to one remains the same for all next algorithms.

3) The third algorithm (similar to the previous), during the generation of cluster with cluster size higher than one, assigns the kinetic energy to them, choosing the best over five extractions that makes the total kinetic energy for cluster with cluster size higher than one approximating better the maxExEcl .

To correct a systematic underestimation of the mean number of clusters, an additional correction to the residual energy for generating cluster with cluster size equal to one can be used.

4)The fourth algorithm (similar to the previous), during the generation of cluster with cluster size higher than one, assigns (by extracting from the proper distribution) the kinetic energy to them, until the total kinetic energy better approximates the maxExEcl.

5)The fifth algorithm is similar to the fourth with almost differences in the technical implementation.

6)The sixth algorithm follows a different methodology. Indeed it uses the total kinetic energy of the event to evaluate a priori the number of cluster, applying the most likelihood criterium.

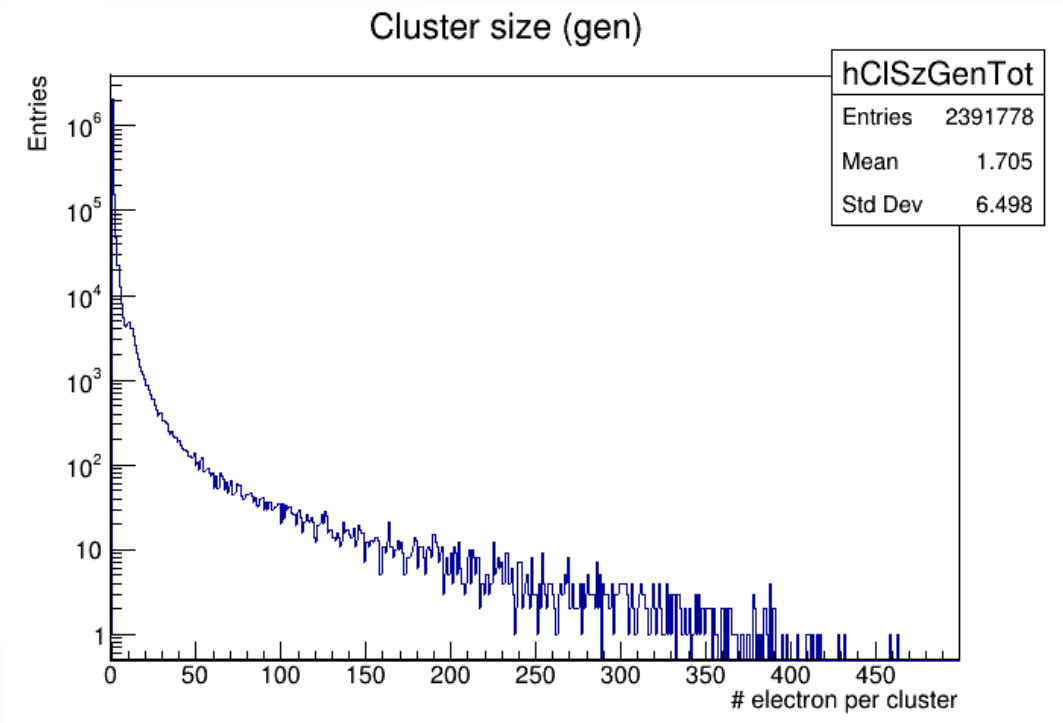
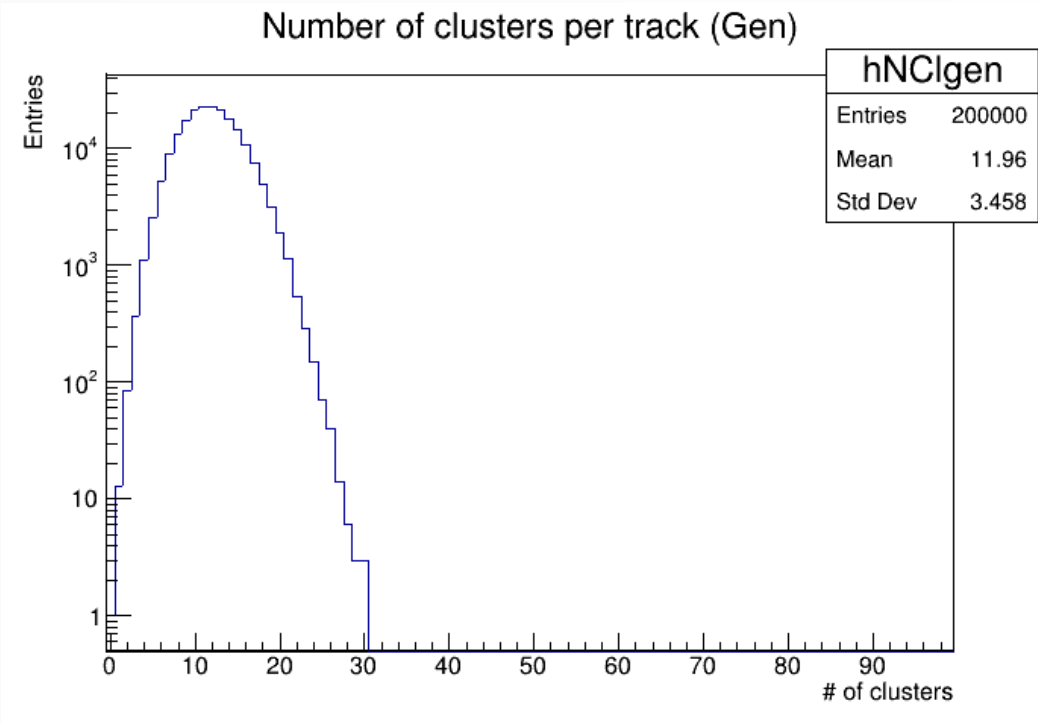
7)The last algorithm is similar to the second algorithm but generates the kinetic energy for cluster with cluster size higher than one by using the fit of kinetic energy distribution.

List of variables

- maxExEcl : total kinetic energy spent to create clusters with cluster size higher than 1
- ExECI : kinetic energy generated per cluster
- Ncl1 : number of clusters with cluster size equal to one
- Nclp : number of clusters with cluster size higher than one
- maxCut : energy value equivalent to the range cut set in Geant4
- totExECI : total kinetic energy reconstructed to create clusters with cluster size higher than one
- Eloss : energy loss from a track passing through the cell
- ClSz : cluster size
- Eizp : primary ionization energy, 15.8 eV
- Eizs : secondary ionization energy, 25.6 eV

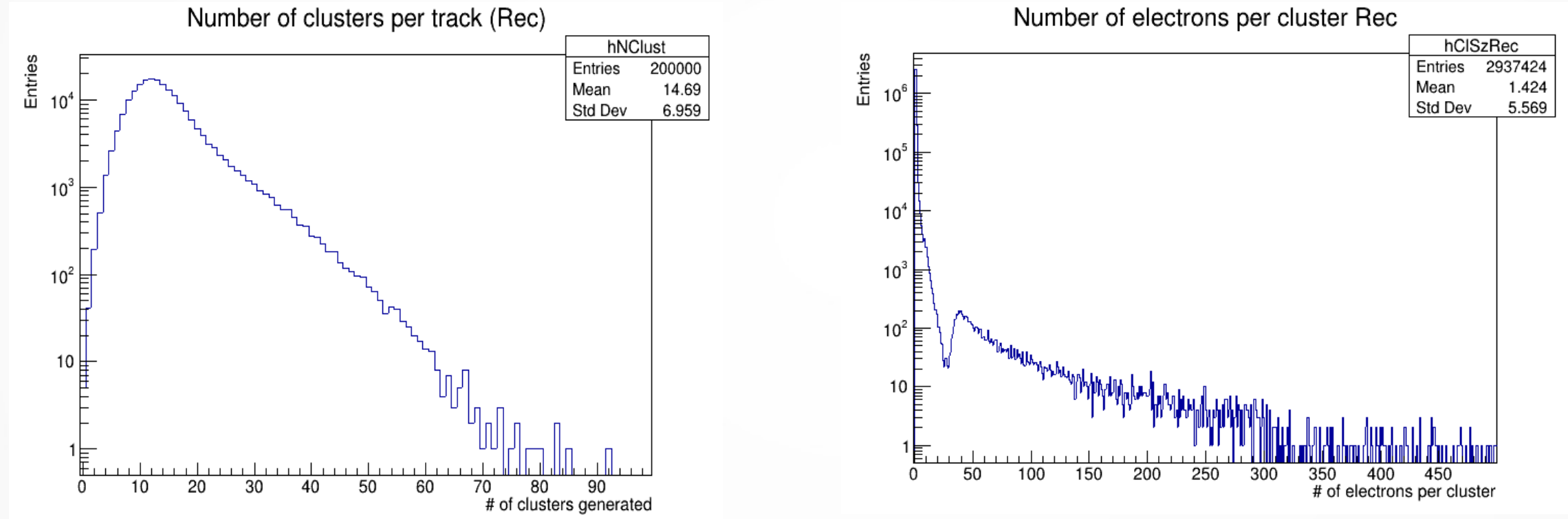
Number of clusters and cluster size for muon at 300 MeV

The next slides will show different attempts to reproduce the number of cluster and the cluster size for a muon in He- iC_4H_{10} .
The reference distribution obtained from Garfield++ simulations are here reported.



First algorithm

The first algorithm uses a reference value of the ratio between clusters containing a single electron and clusters containing more than one electron (R_t). Using the R_t value, the algorithm chooses to create cluster with cluster size one or higher. Then, it assigns the kinetic energy to each cluster by using the proper distributions. If the cluster has more than one electron, a check on the total kinetic energy is performed and its cluster size is evaluated. The procedure is repeated until the sum of primary ionization energy and kinetic energy per cluster saturate the energy loss of the event.



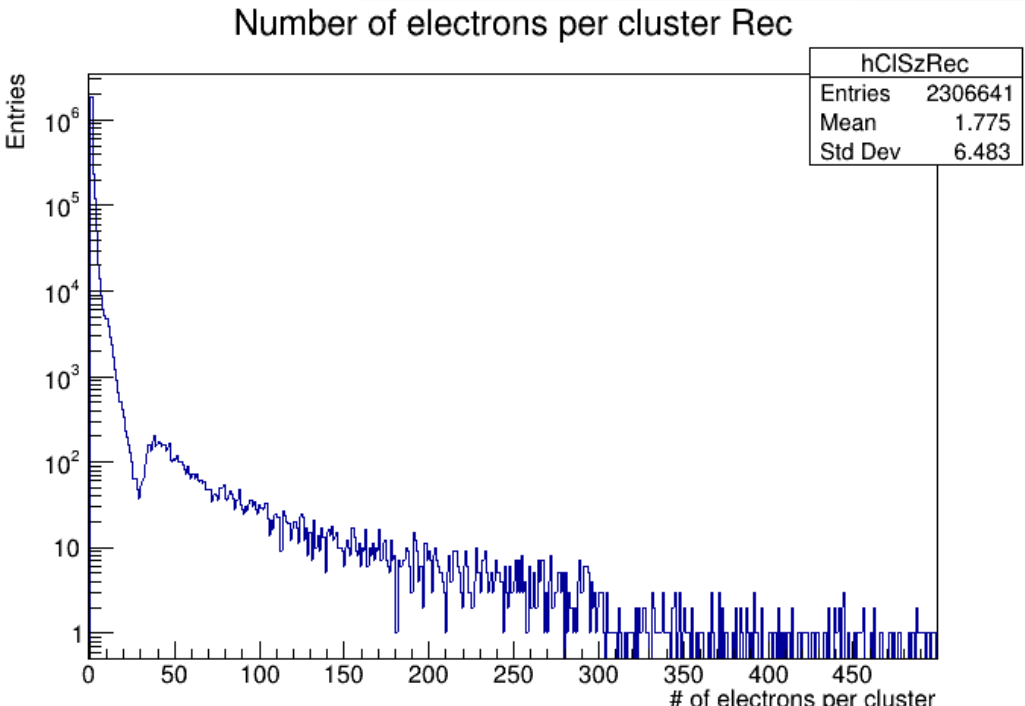
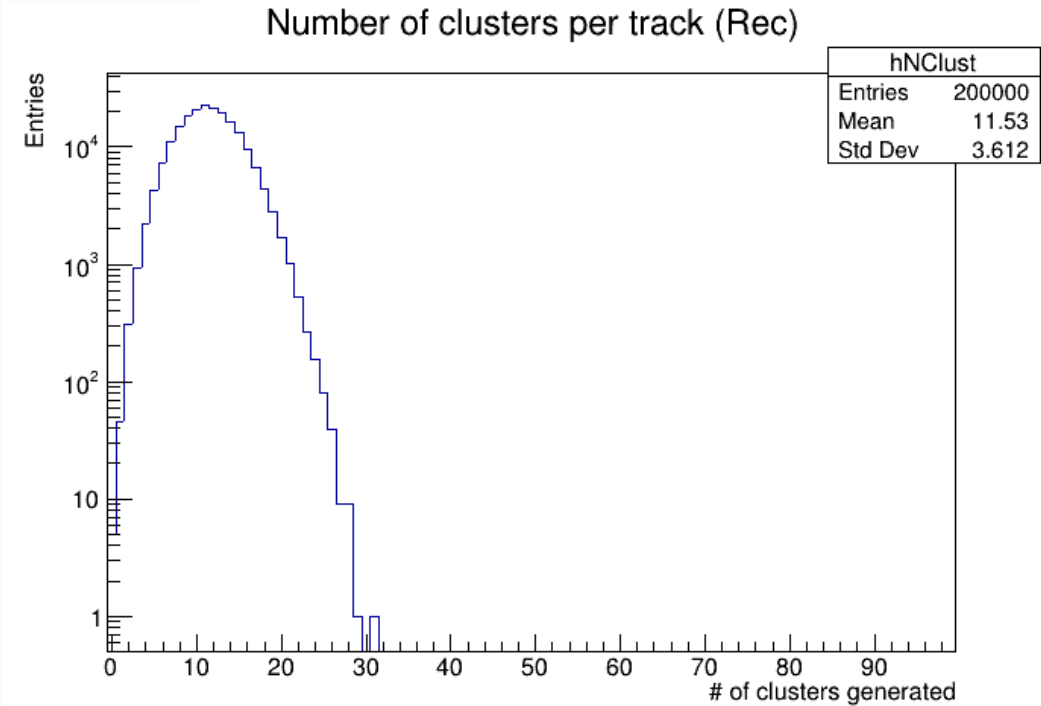
- The peak is almost in agreement with the expected value, but the distribution of number of clusters is not Poissonian, indeed presents a long fake tail.
- The cluster size presents a depression before the value of 50 electrons.

We change the method for algorithm implementation.
09/02/2021

Second algorithm

The second algorithm, if maxExEcl is higher than zero, generates the kinetic energy for clusters with cluster size higher than one by using its distribution and evaluates cluster size. This procedure is repeated until the sum of primary ionization energy and kinetic energy per cluster saturate the maxExEcl of the event.

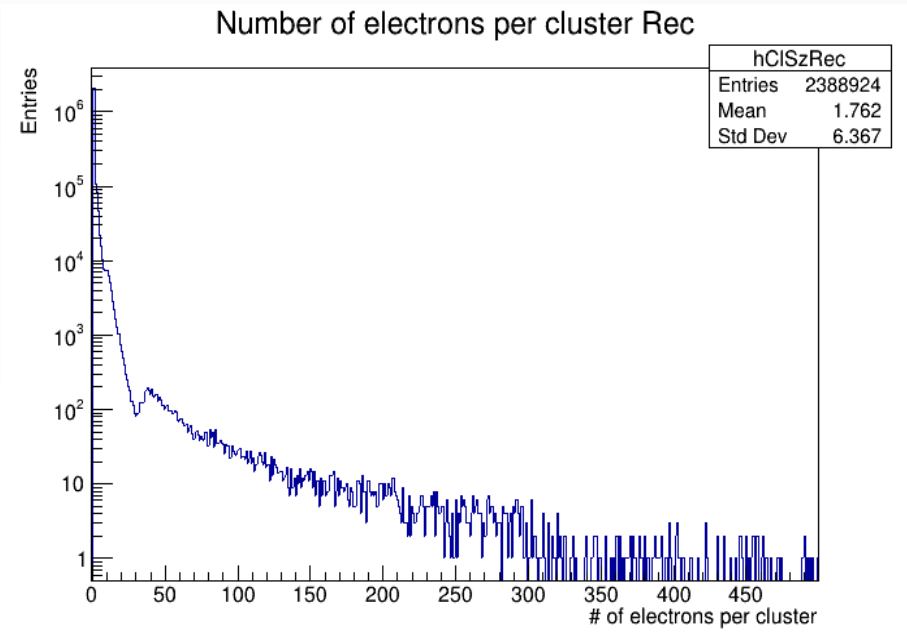
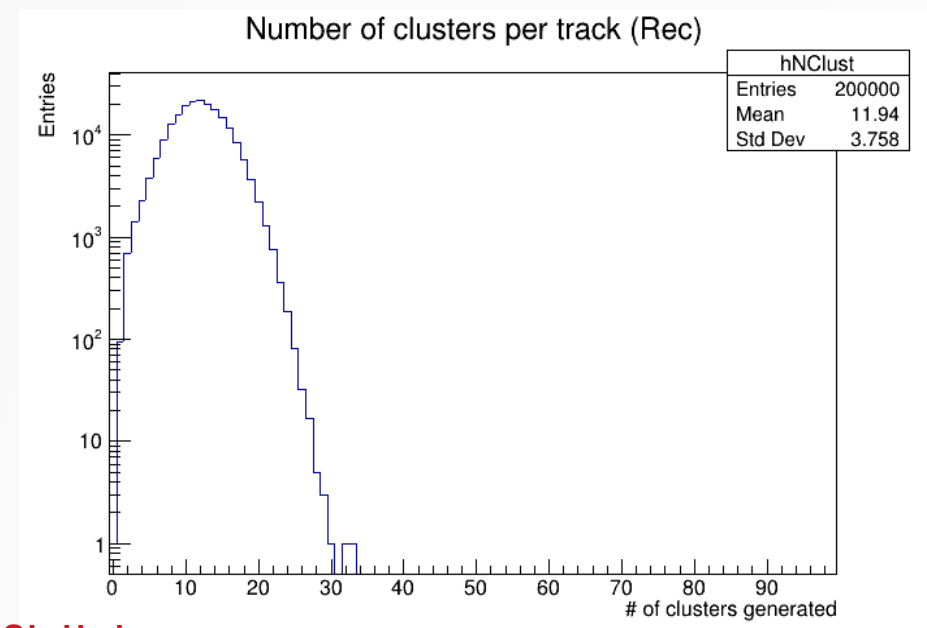
Then, using the remaining energy ($E_{loss} - \text{maxExEcl}$), the algorithm creates clusters with cluster size equal to one by assigning their kinetic energy according to the proper distribution.



- The distribution is Poissonian.
- The cluster size presents again a depression before the value of 50 electrons.

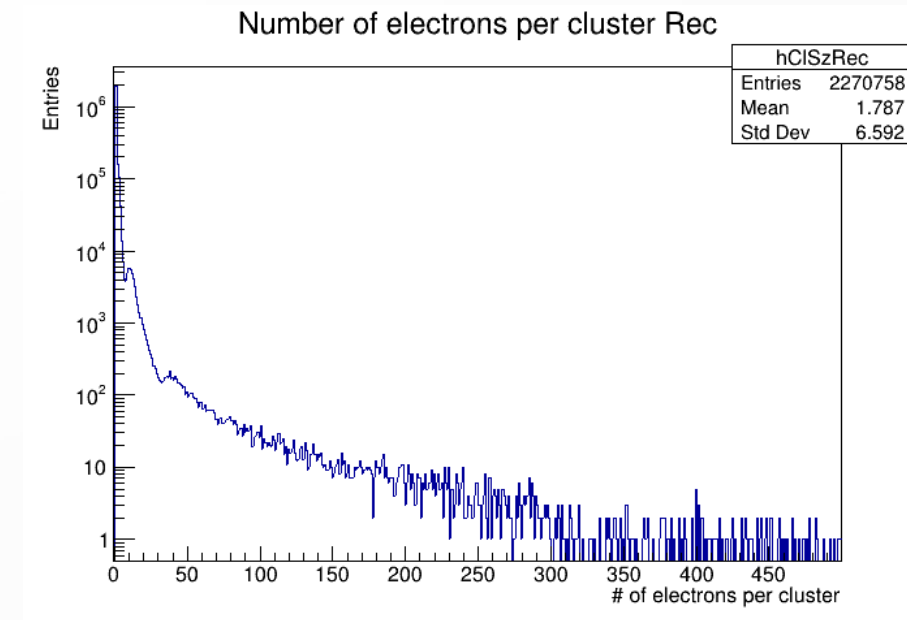
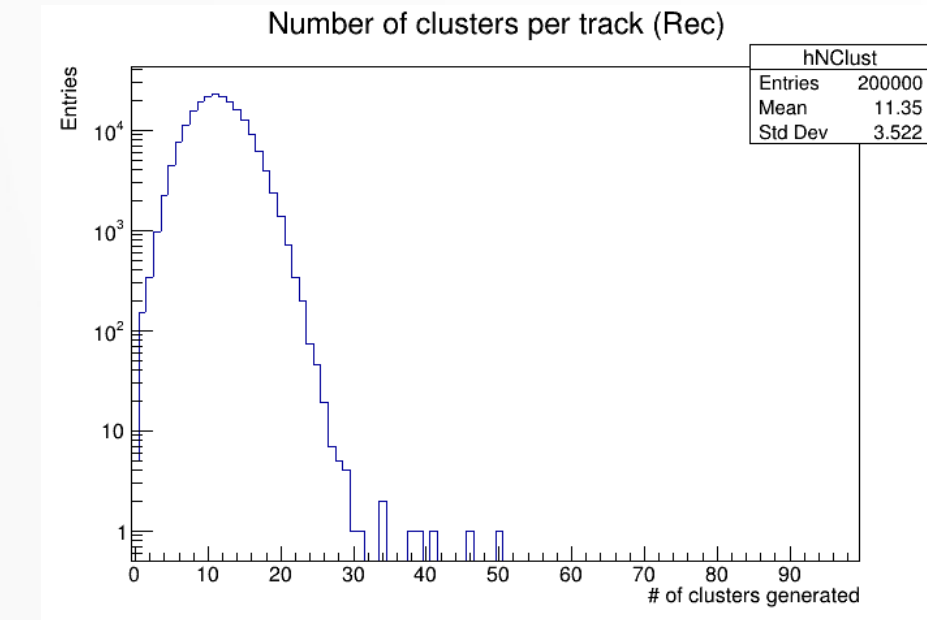
We tried other algorithms.

Third try: with correction



Sixth try

The sixth algorithm follows a different methodology. Indeed it uses the total kinetic energy of the event to evaluate a priori the number of cluster, applying the most likelihood criterium.



Summary table

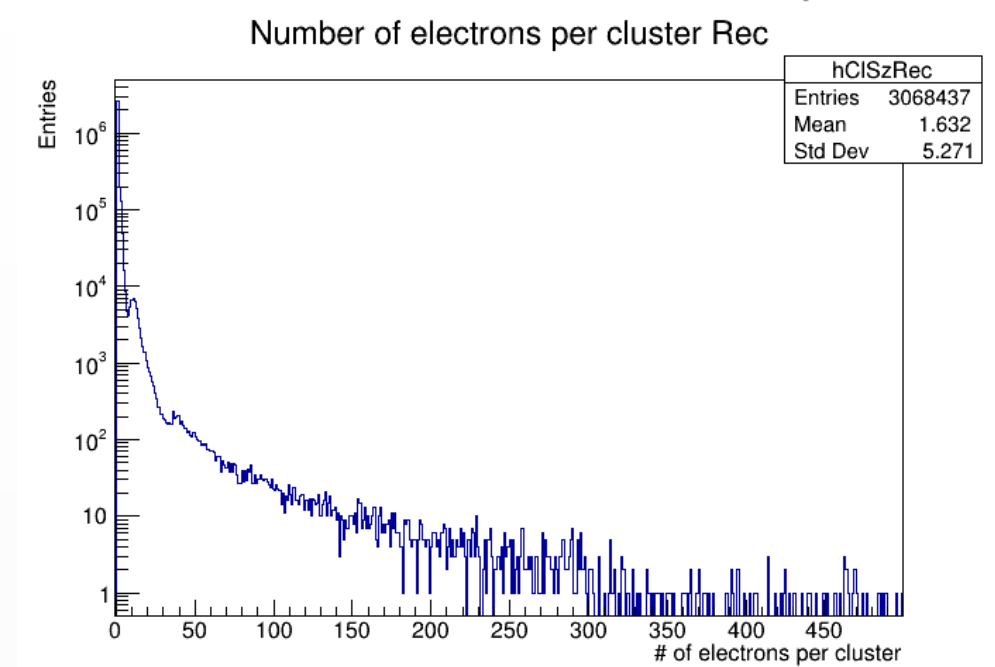
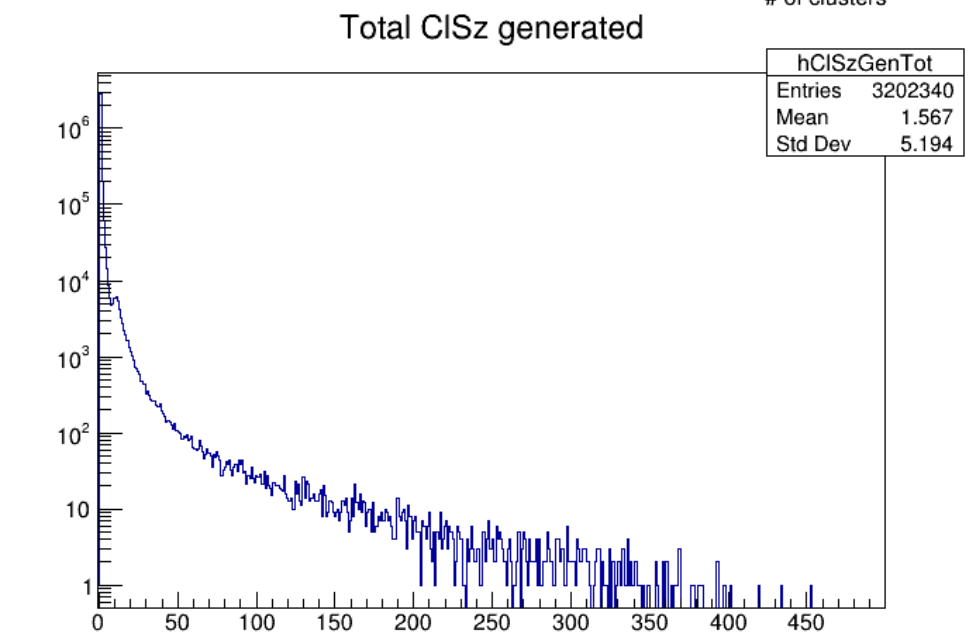
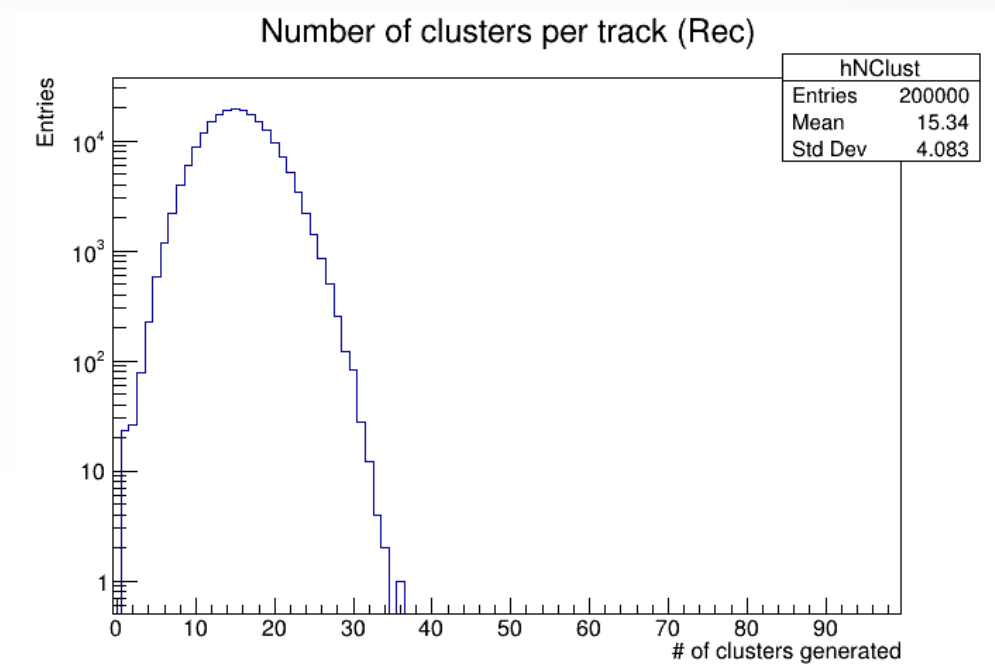
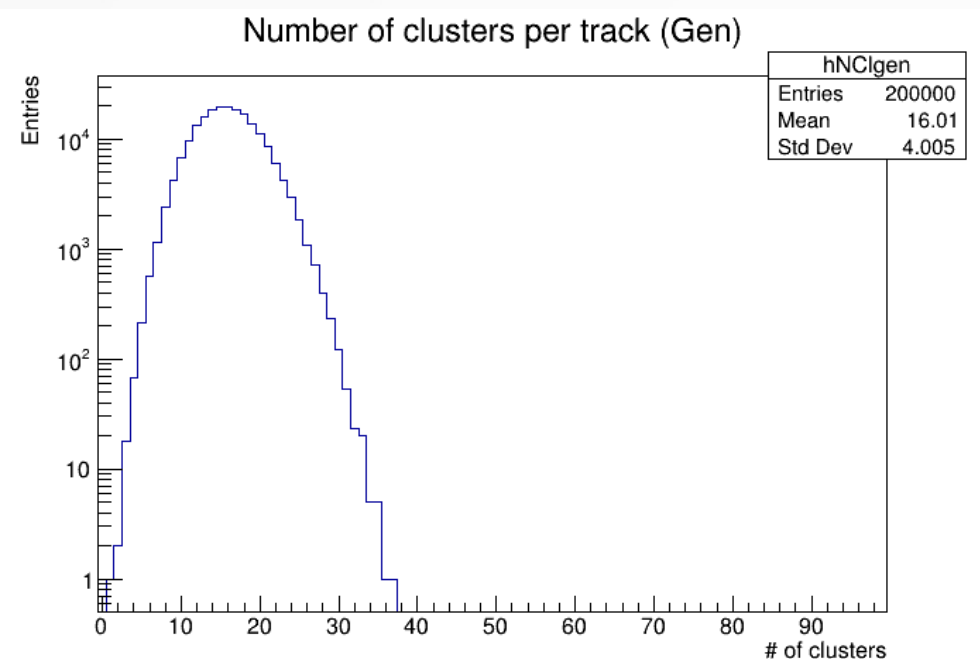
	Ncl	σ Ncl	Ncl1	σ Ncl1	Nclp	σ Nclp	maxNclp	#entries of Nclp gen/ #entries of Nclp rec
expected value	11.96	3.458	10.44	3.228	1.912	1.04	10.05	
Algorithm								
1	14.69	6.959	12.85	6.426	2.157	1.25	13.5	1.082
2	11.53	3.612	9.225	3.633	3.448	2.602	25.5	0.899
3 (without correction)	10.99	3.72	9.339	3.608	2.428	1.321	14.5	0.886
3 (with correction)	11.94	3.758	10.25	3.69	2.429	1.317	12.5	0.889
4	11.63	3.642	9.388	3.633	3.349	2.675	24.5	0.889
5	12.11	3.808	9.533	3.935	4.186	2.972	24.5	0.820
6	11.36	3.525	9.501	3.511	2.724	1.311	12.5	0.886
7	7.012	4.026	7.593	3.862	2.286	1.258	12.5	1.295

	CISz	σ CISz
expected value	1.705	6.498
Algorithm		
1	1.424	5.569
2	1.775	6.483
3 (without correction)	1.828	6.695
3 (with correction)	1.762	6.367
4	1.753	6.434
5	1.698	6.231
6	1.787	6.67
7	2.485	9.012

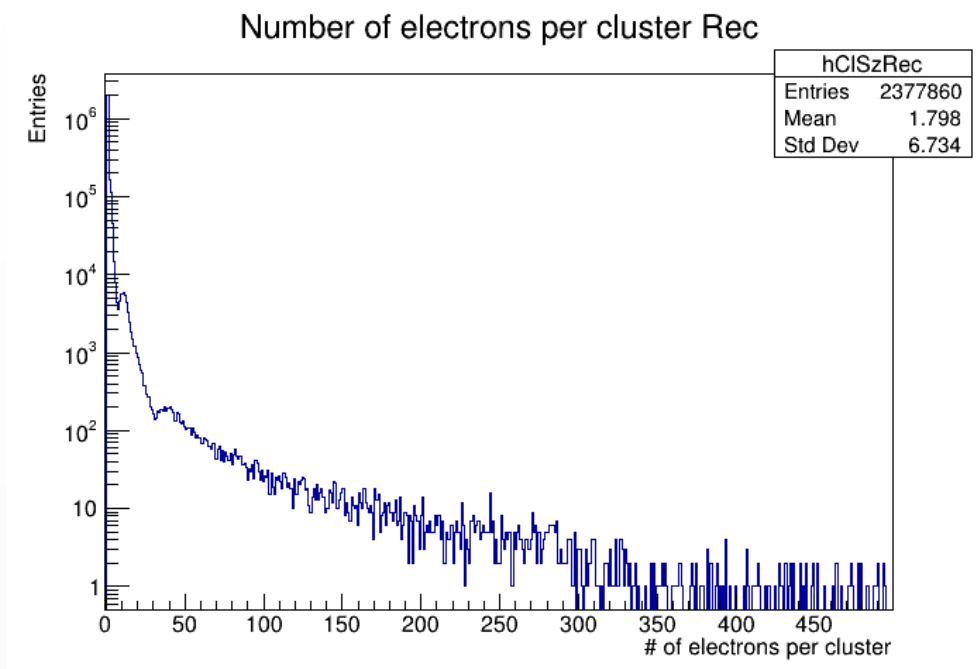
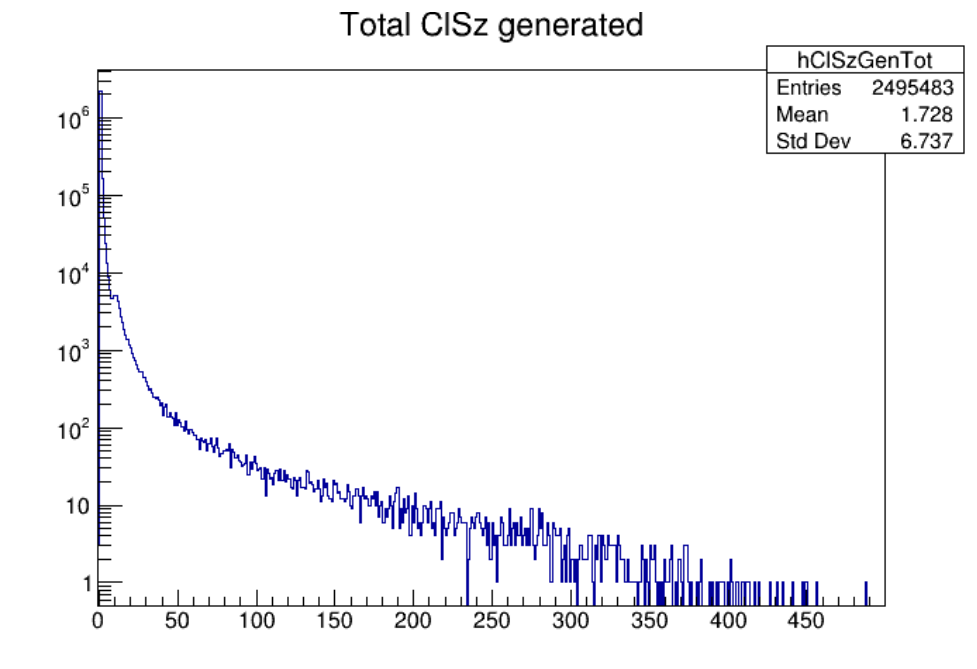
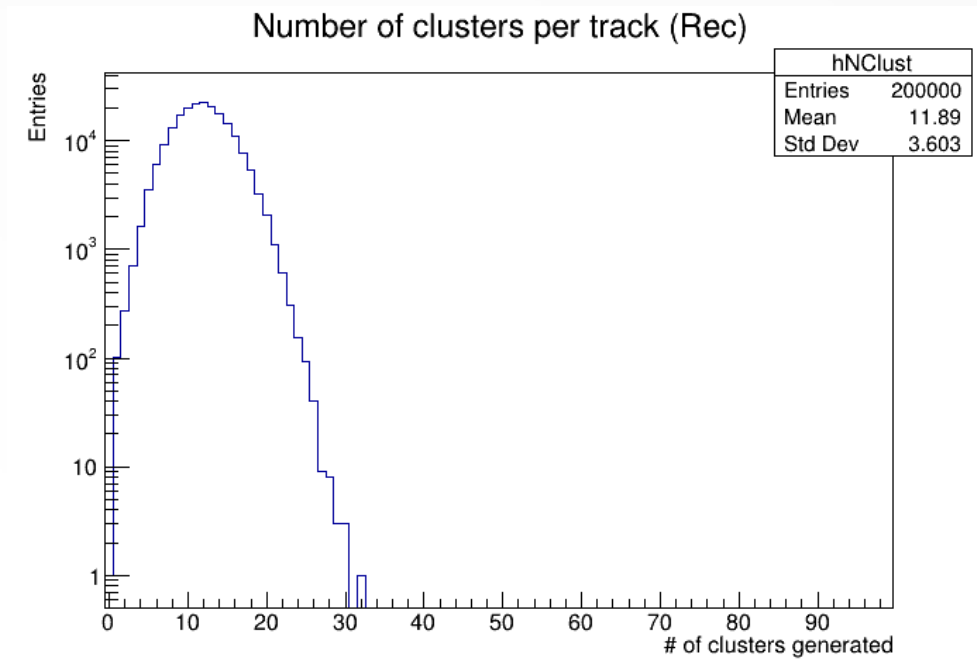
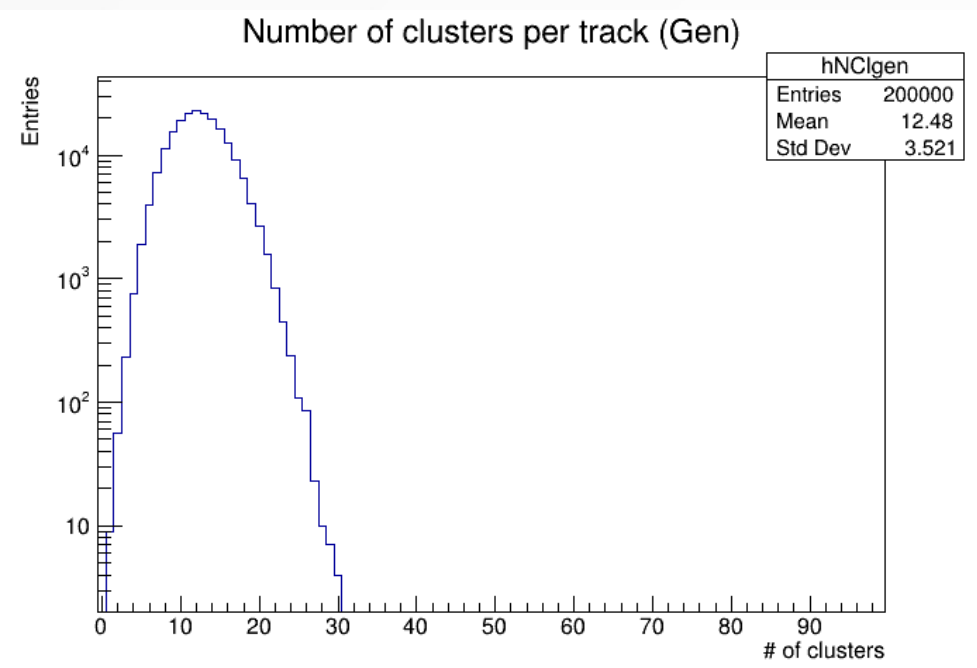
- The second and third algorithm produce a number of cluster distribution, which follows the Poissonian shape and gives a mean value compatible with the one expected.
- The sixth algorithm produces a number of cluster distribution, which follows the Poissonian shape and gives a mean value compatible with the one expected and also reconstructs a cluster size distribution whose shape is similar to the one expected.
- The other algorithms do not reproduce the Poissonian shape expected for number of clusters distribution.

Test of algorithms with other particles and other momentum values

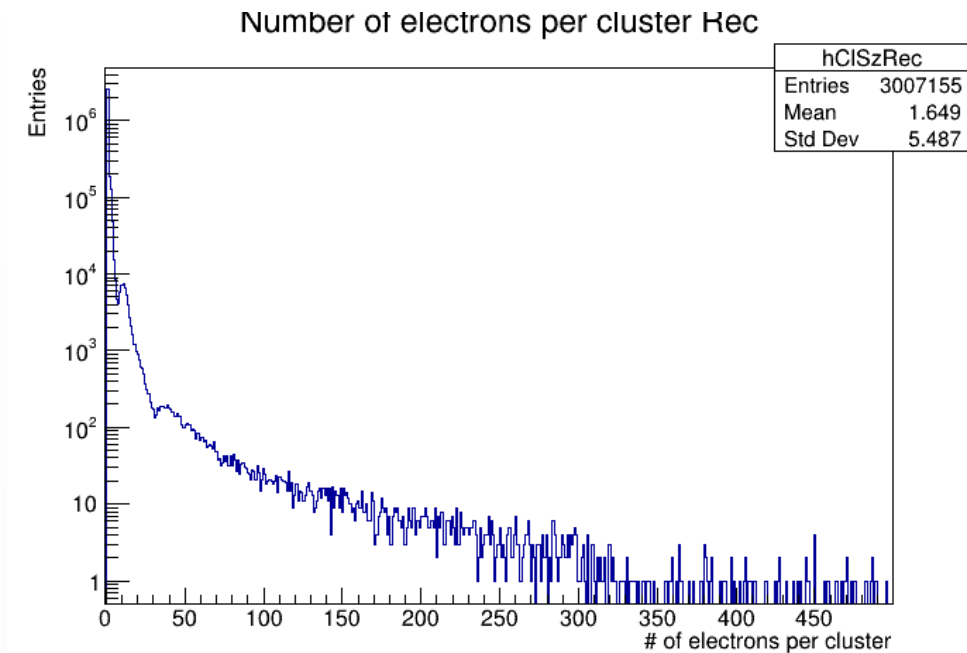
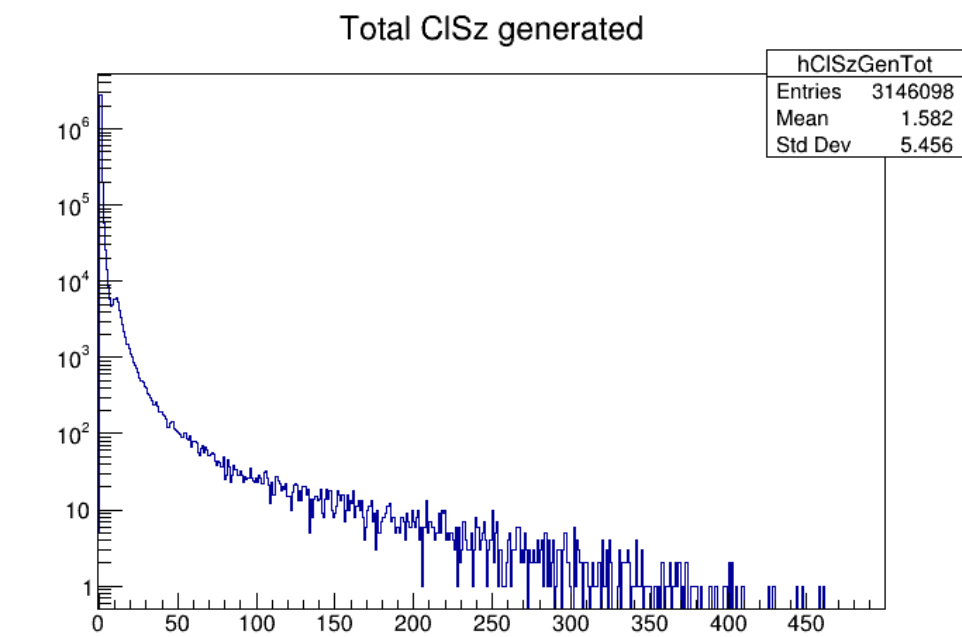
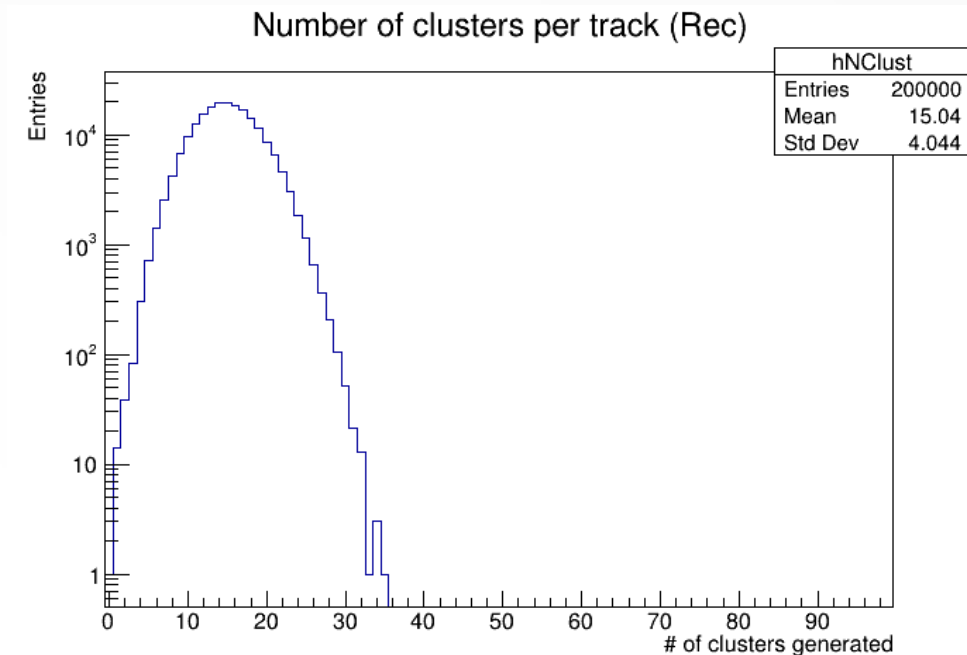
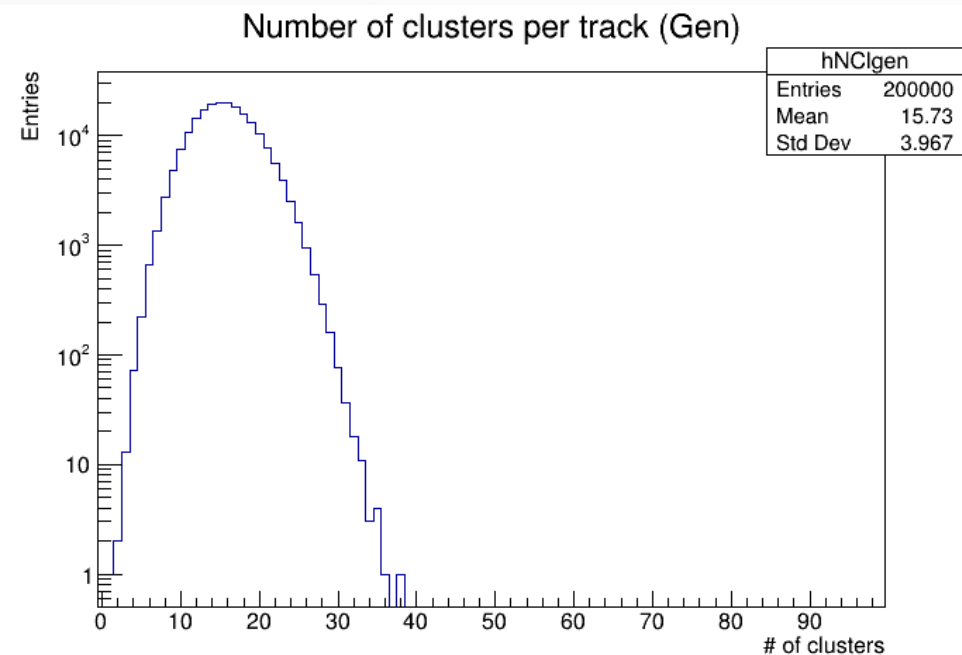
Muon at p= 10 GeV, sixth algorithm



Pion at p= 300 MeV, sixth algorithm

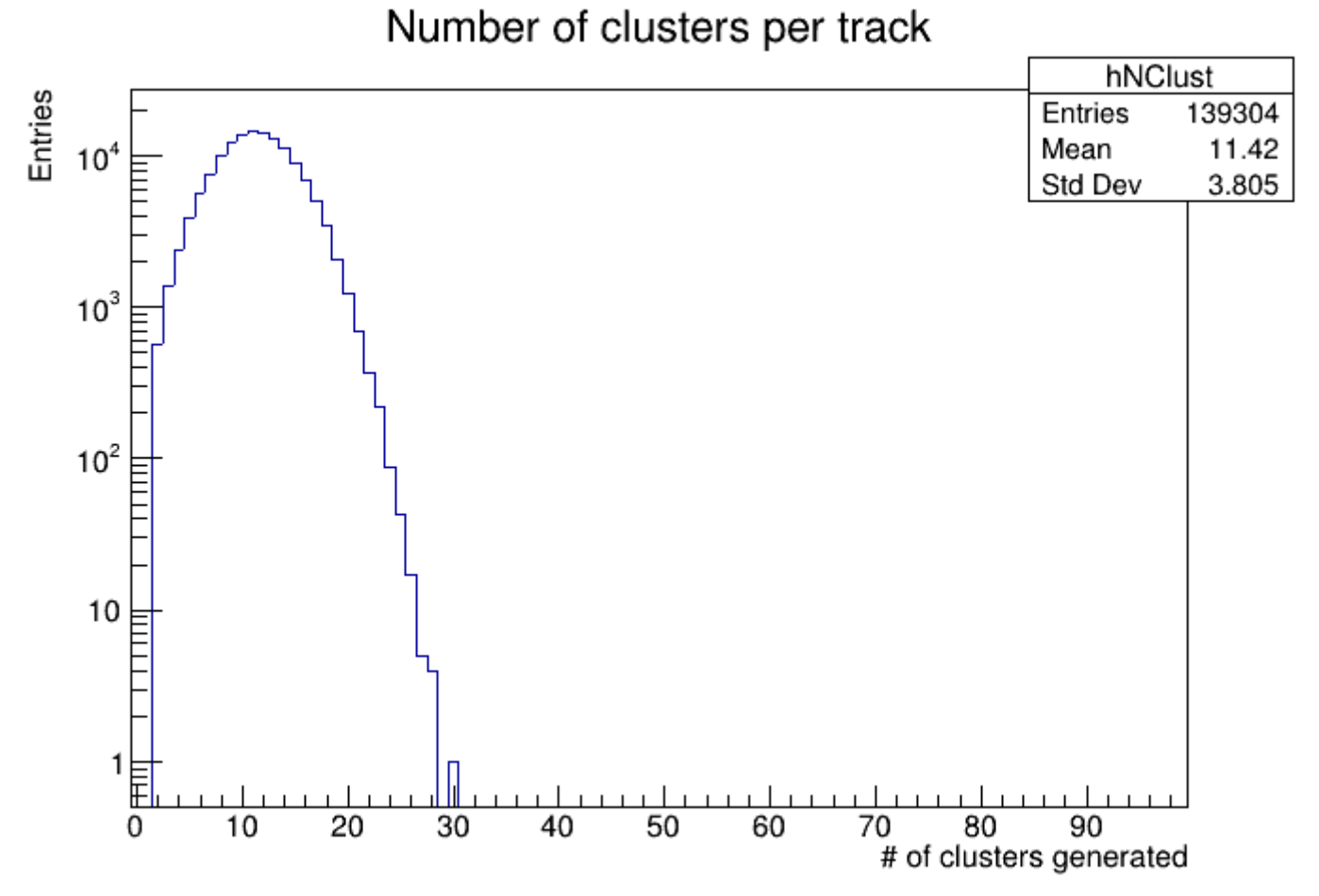


Pion at p= 10 GeV, sixth algorithm



Geant4 result

The algorithm is tested with Geant4 simulations and the results obtained are compatible with the ones obtained with Garfield++.
The plot shows the number of cluster distribution obtained with Geant4 by applying the third algorithm.



Muon at p=300 MeV

Conclusion and next step

We have three algorithms that give us good results (2,3,6).

Now, we are working on evaluation of the trend of the other important parameters necessary to automate the procedure.

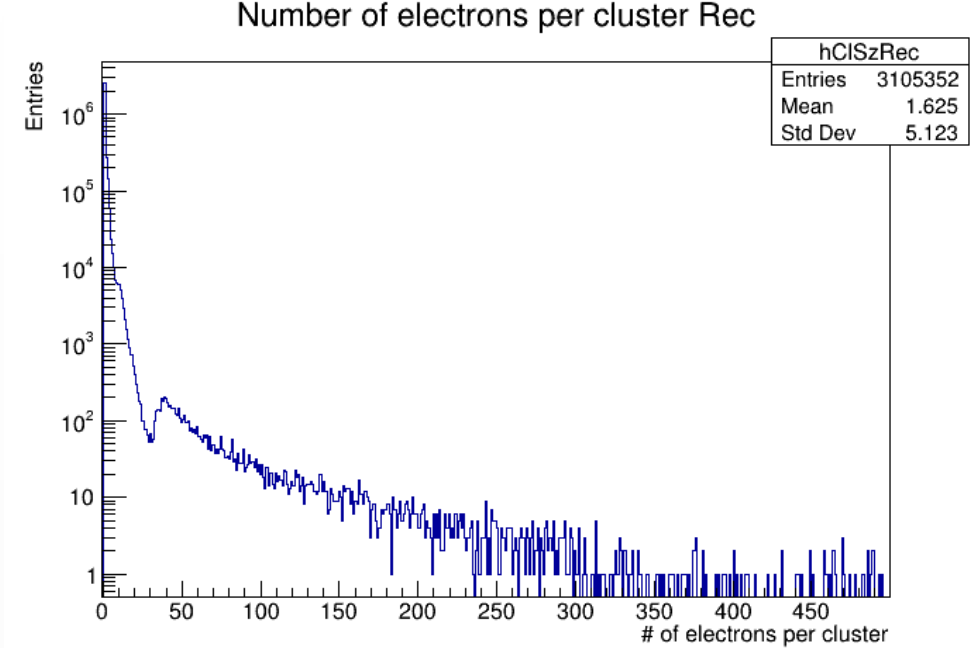
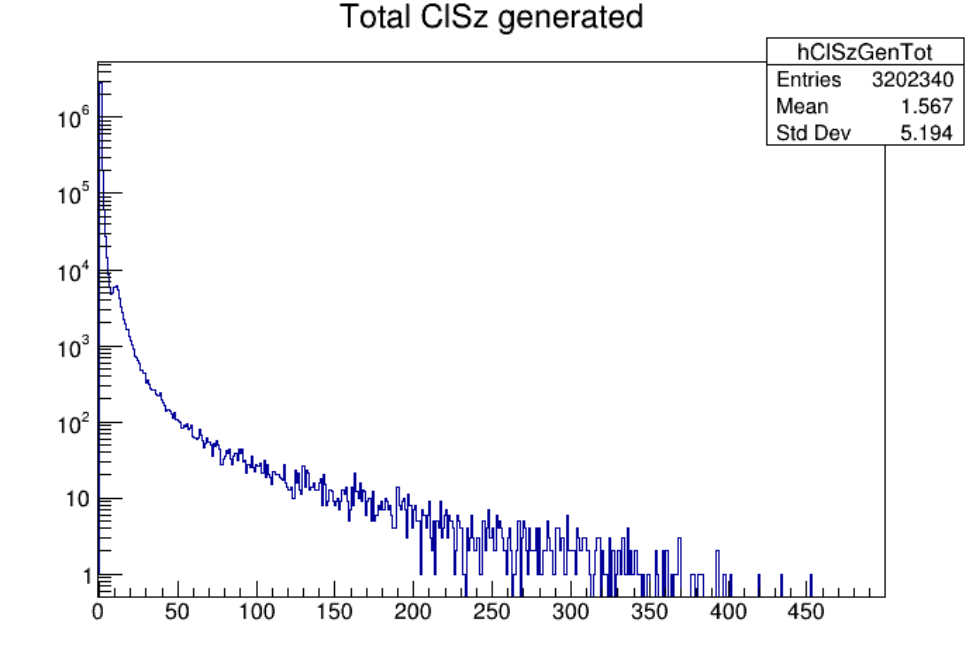
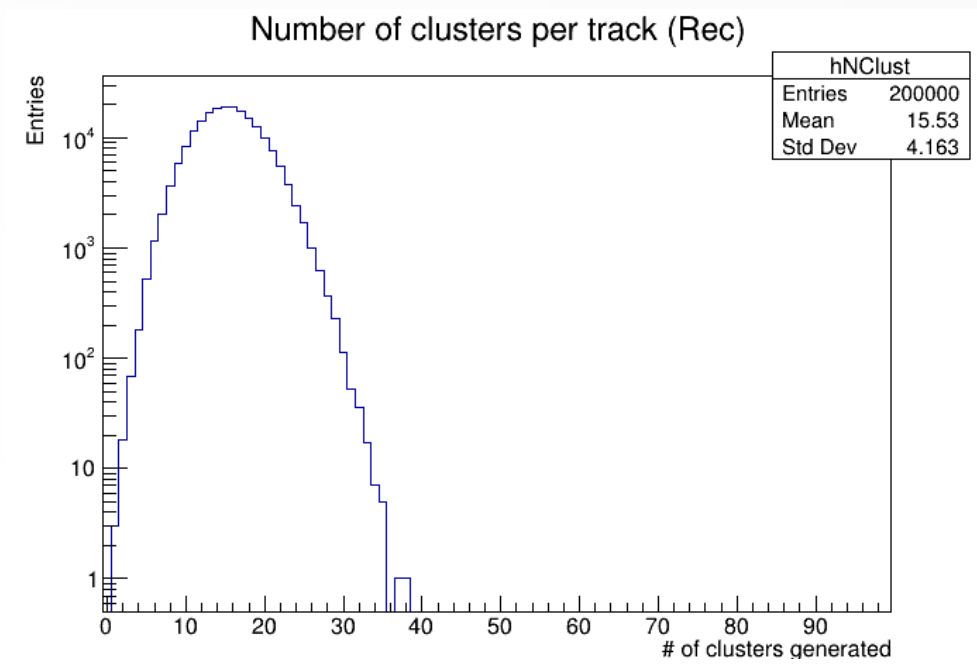
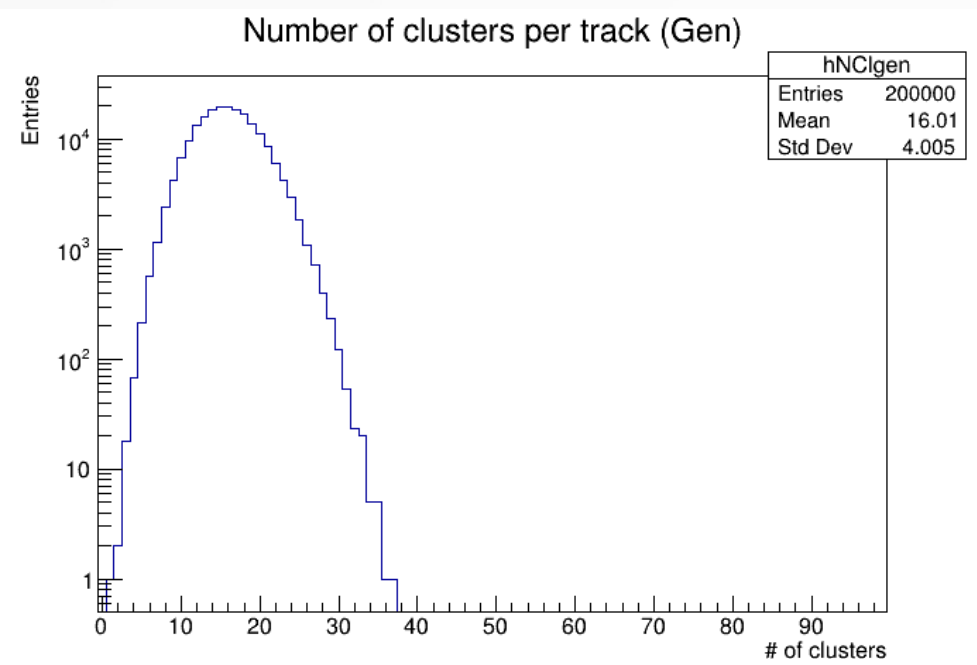
We will compare simulation results with experimental results.

We will apply the reconstruction algorithm in Geant4 full drift chamber simulation.

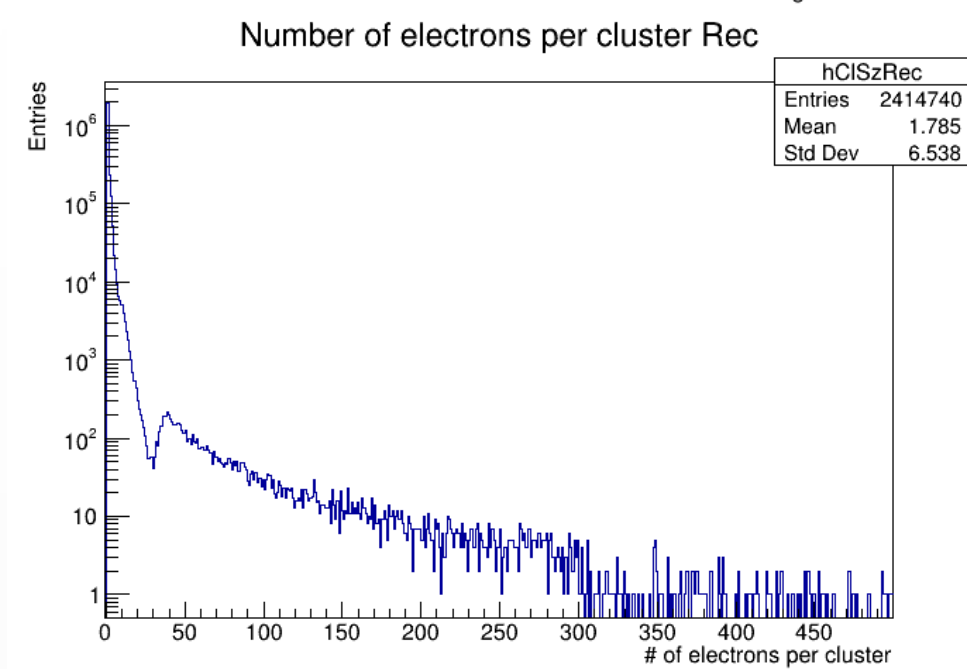
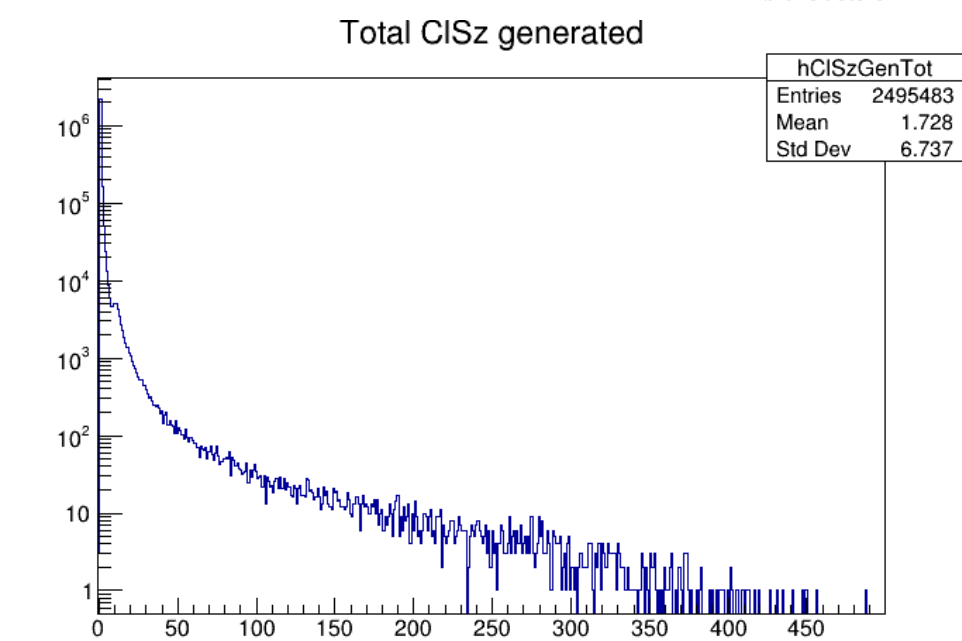
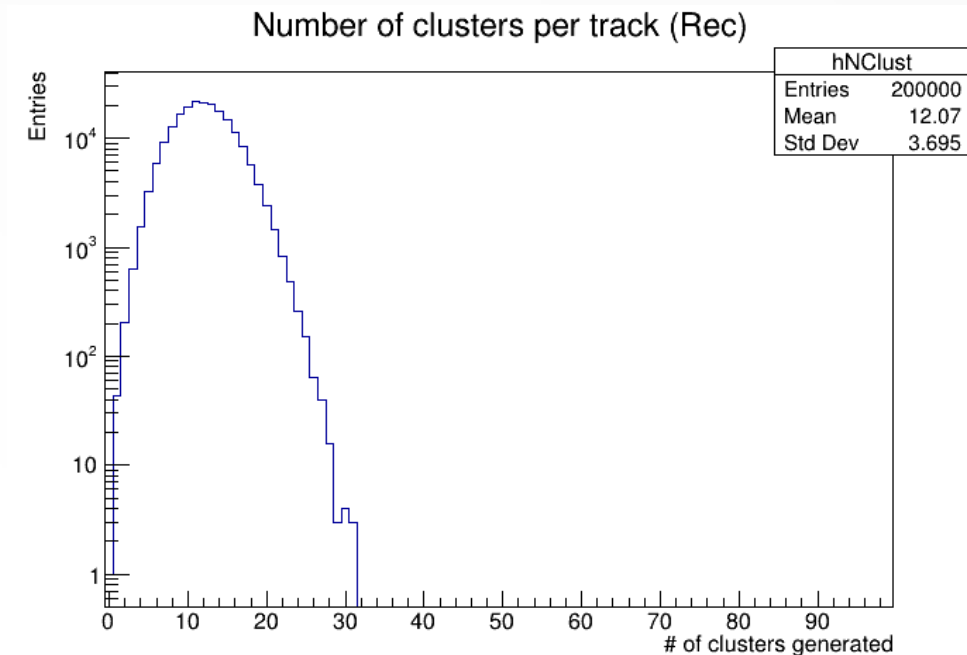
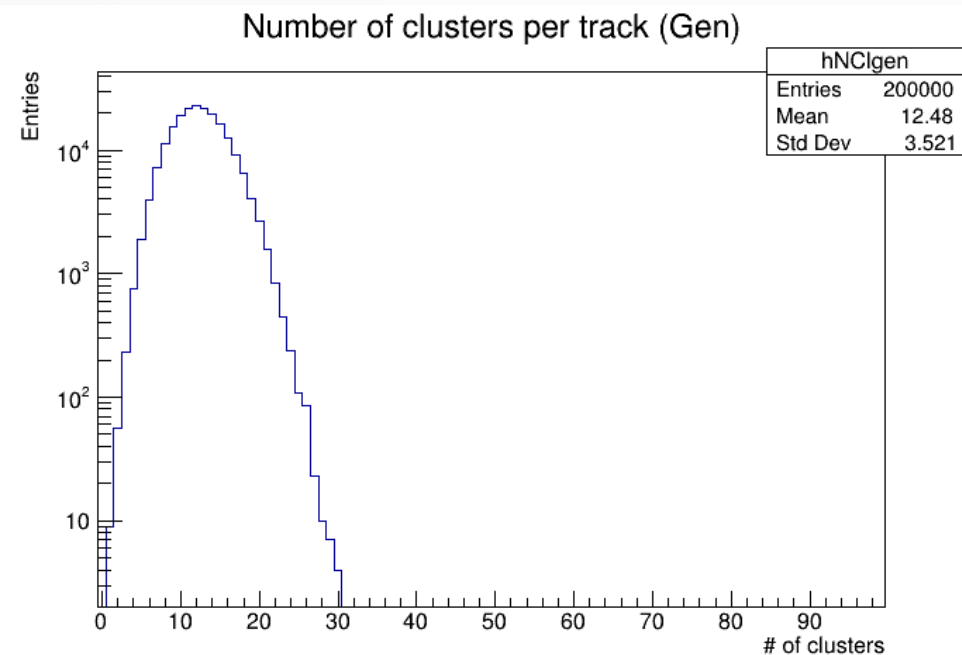
Back up

Test of algorithms with other particles and other momentum values

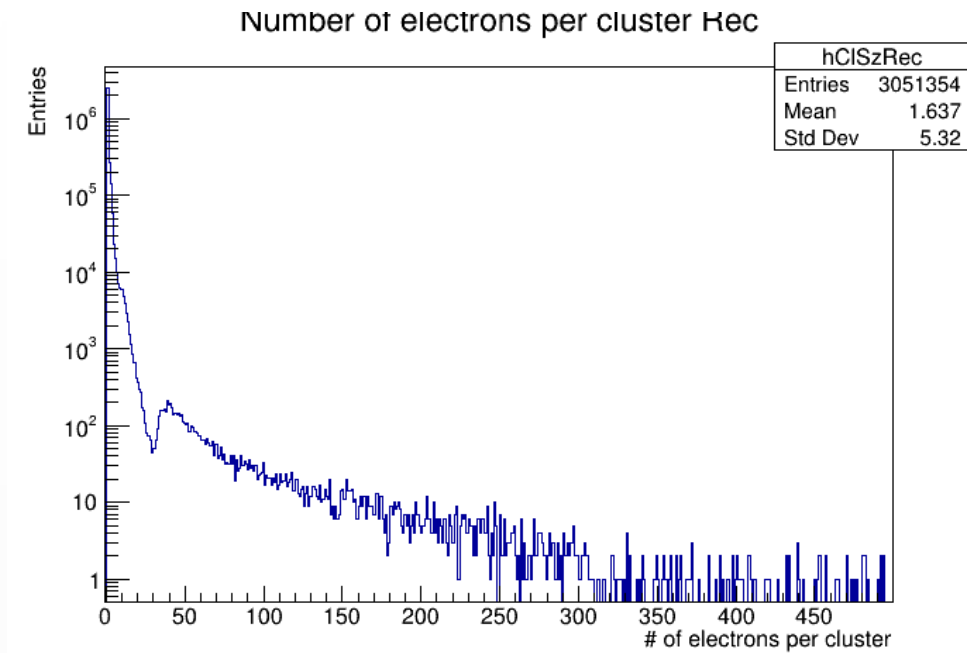
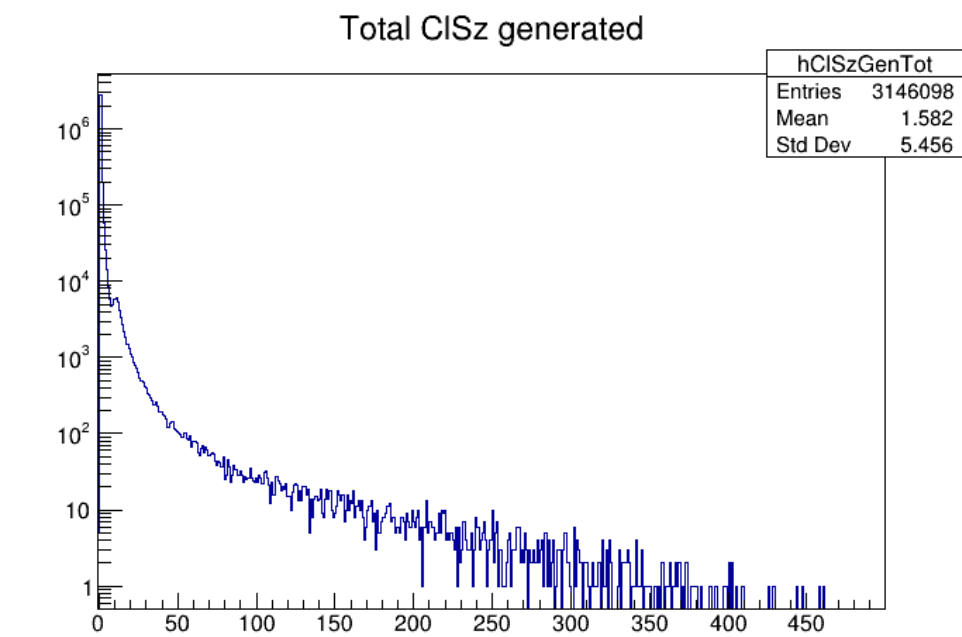
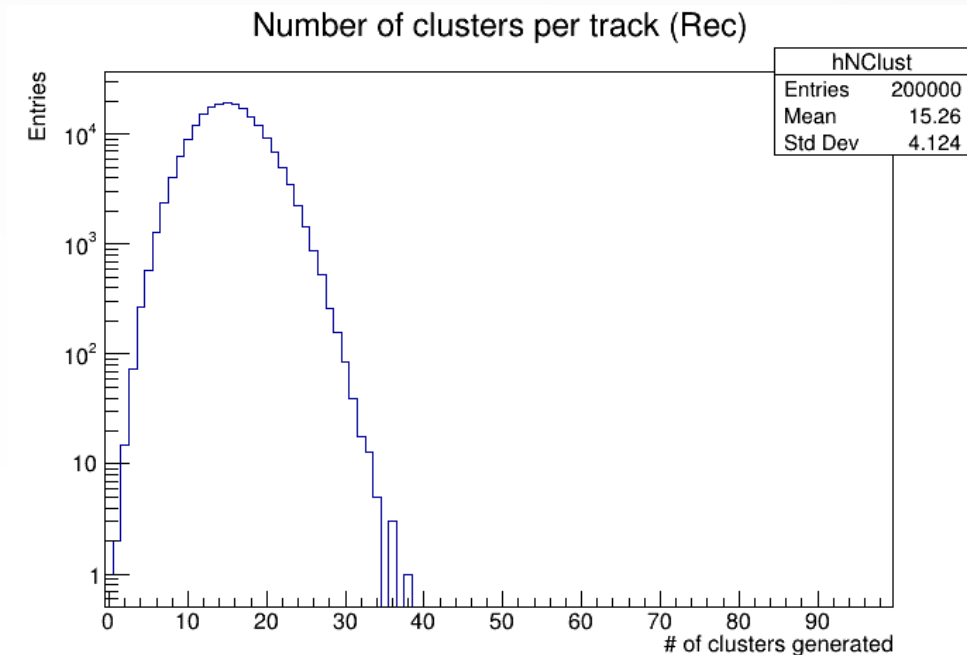
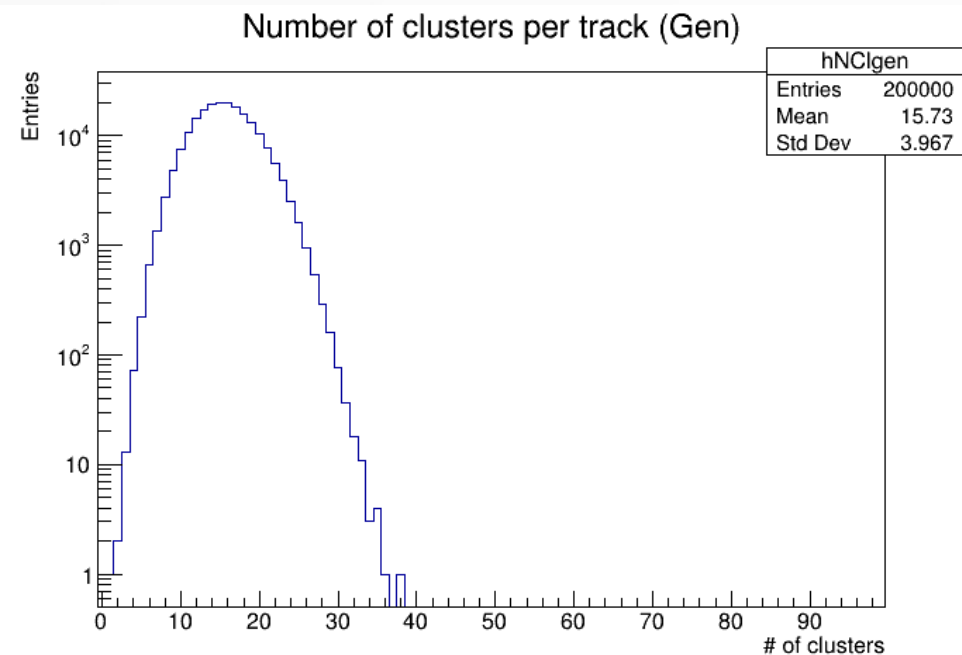
Muon at p= 10 GeV, second algorithm



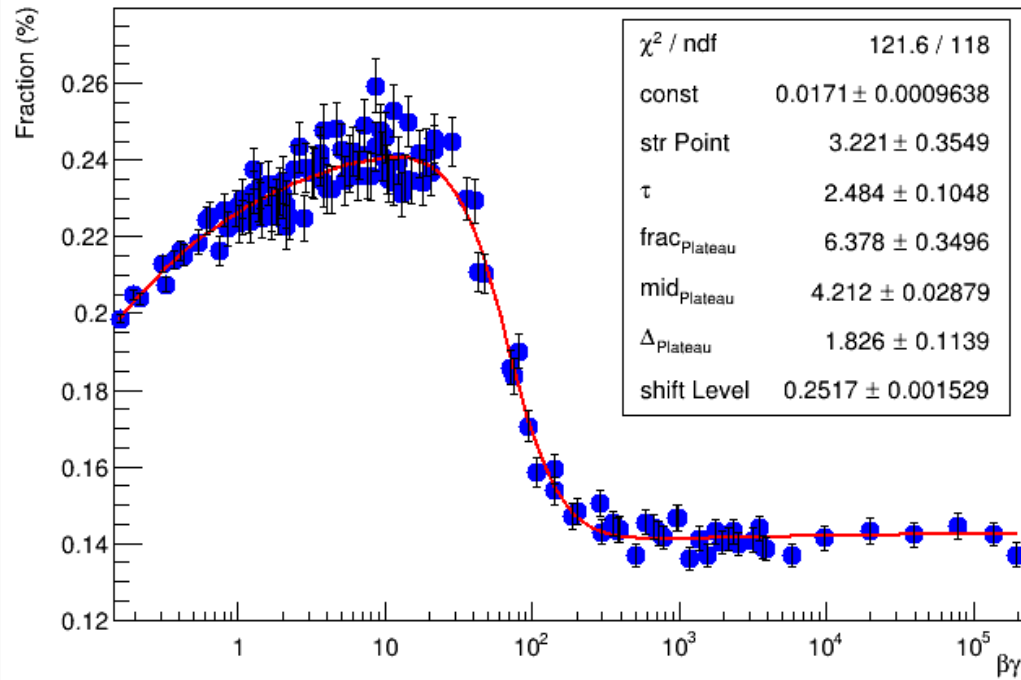
Pion at p= 300 MeV, second algorithm



Pion at p= 10 GeV, second algorithm



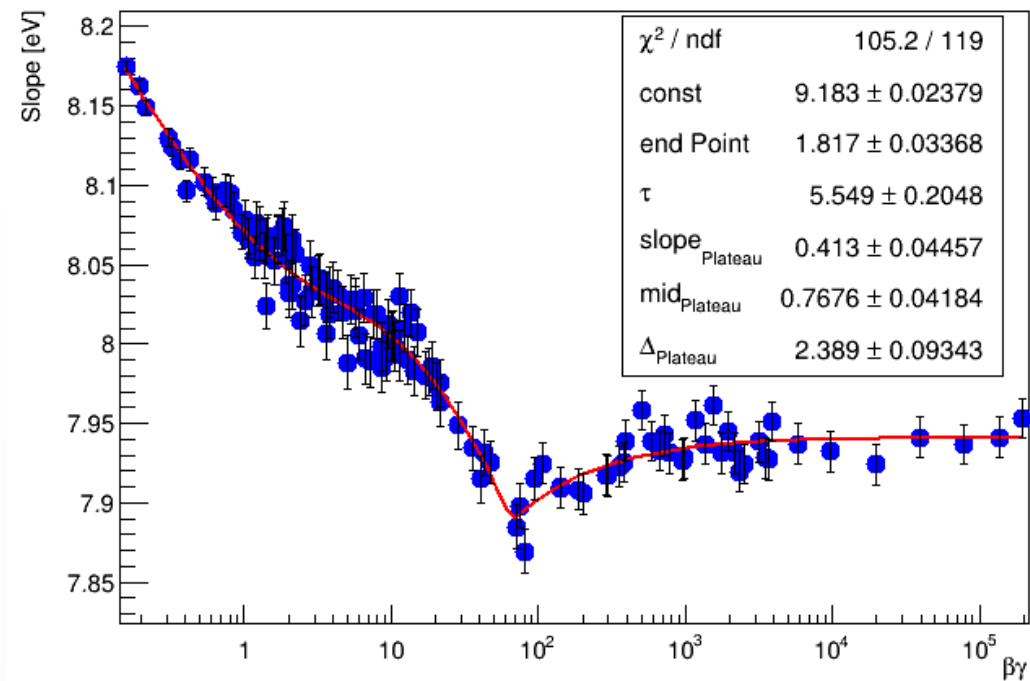
Fraction (%) gaus+exp fit of Extra Energy with CISz=1



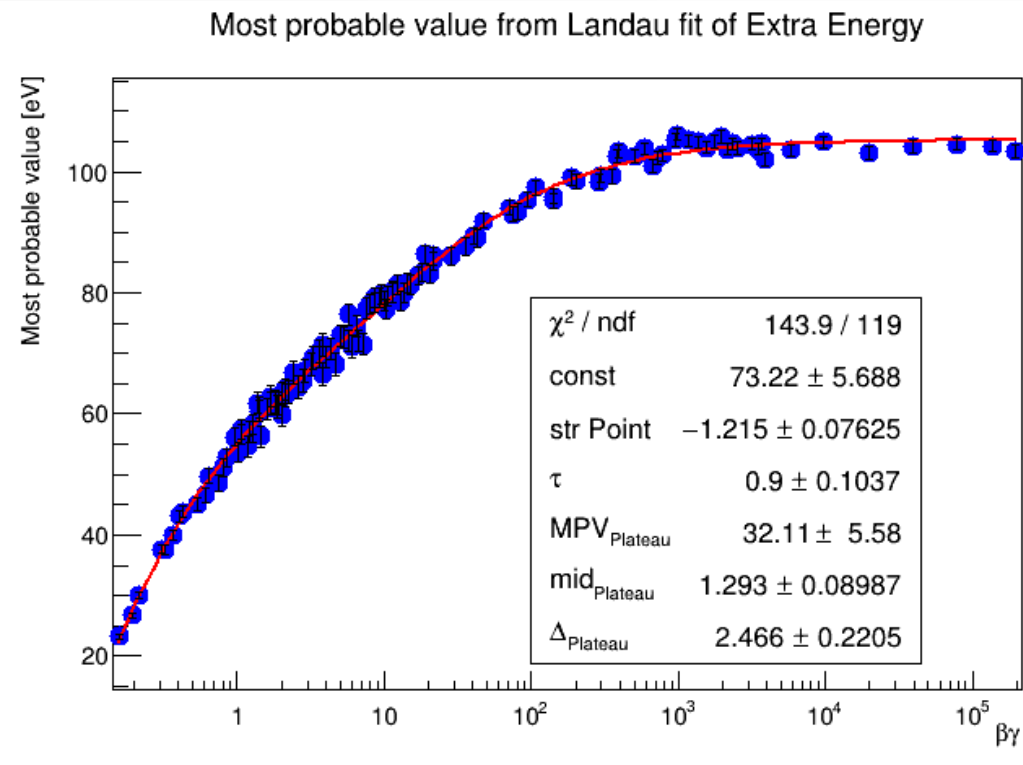
Here the distribution of the fraction value from fit, fitted with a decreasing exponential function plus an efficiency function.

Here the distribution of the slope value of fit, fitted with an 1-exponential function plus an efficiency function.

Slope from gaus+exp fit of Extra Energy with CISz=1

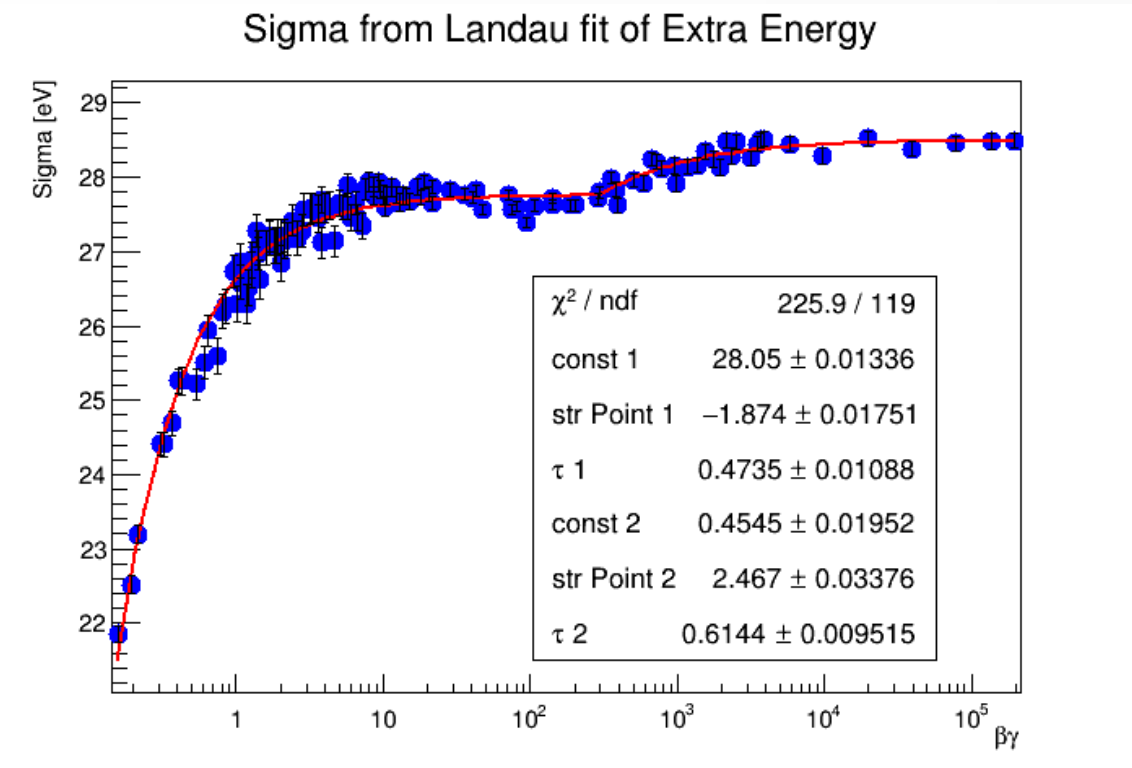


Fit parameters for kinetic energy distribution for cluster with cluster size higher than 1

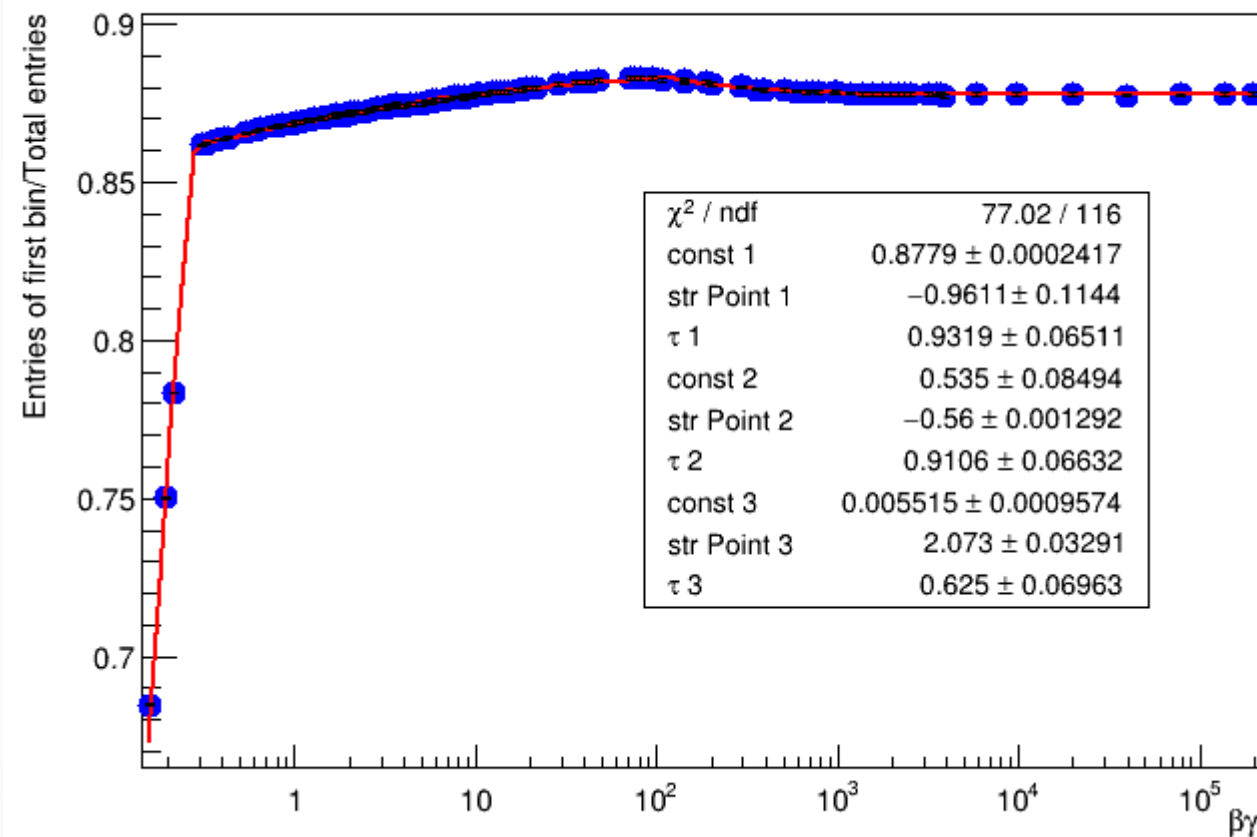


Here the distribution of the most probable value, fitted with 1- decreasing exponential function plus an efficiency function.

Here the distribution of the sigma value, fitted with 1- decreasing exponential function plus an efficiency function.



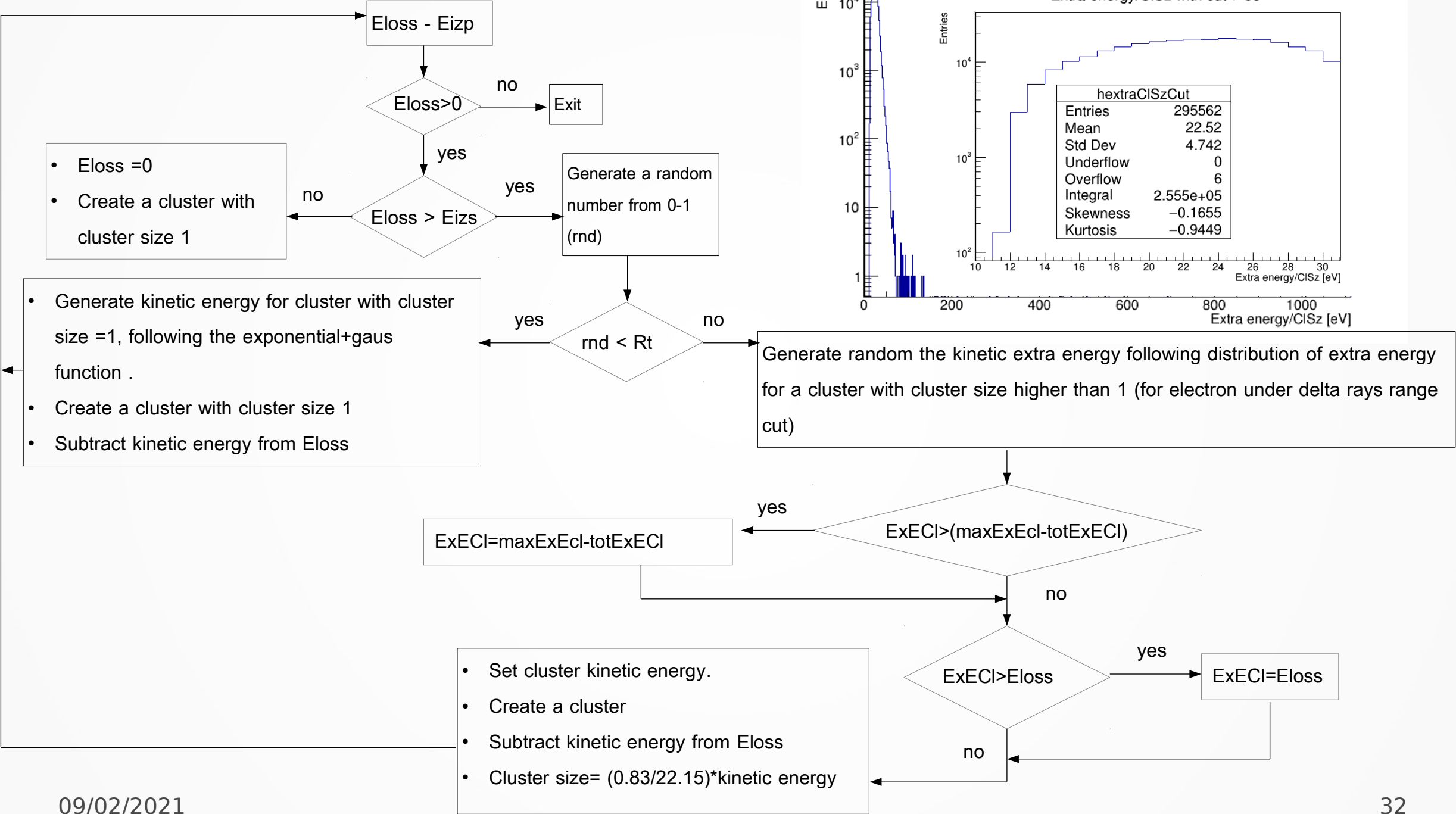
Entries of first bin/Total entries of CISz distribution



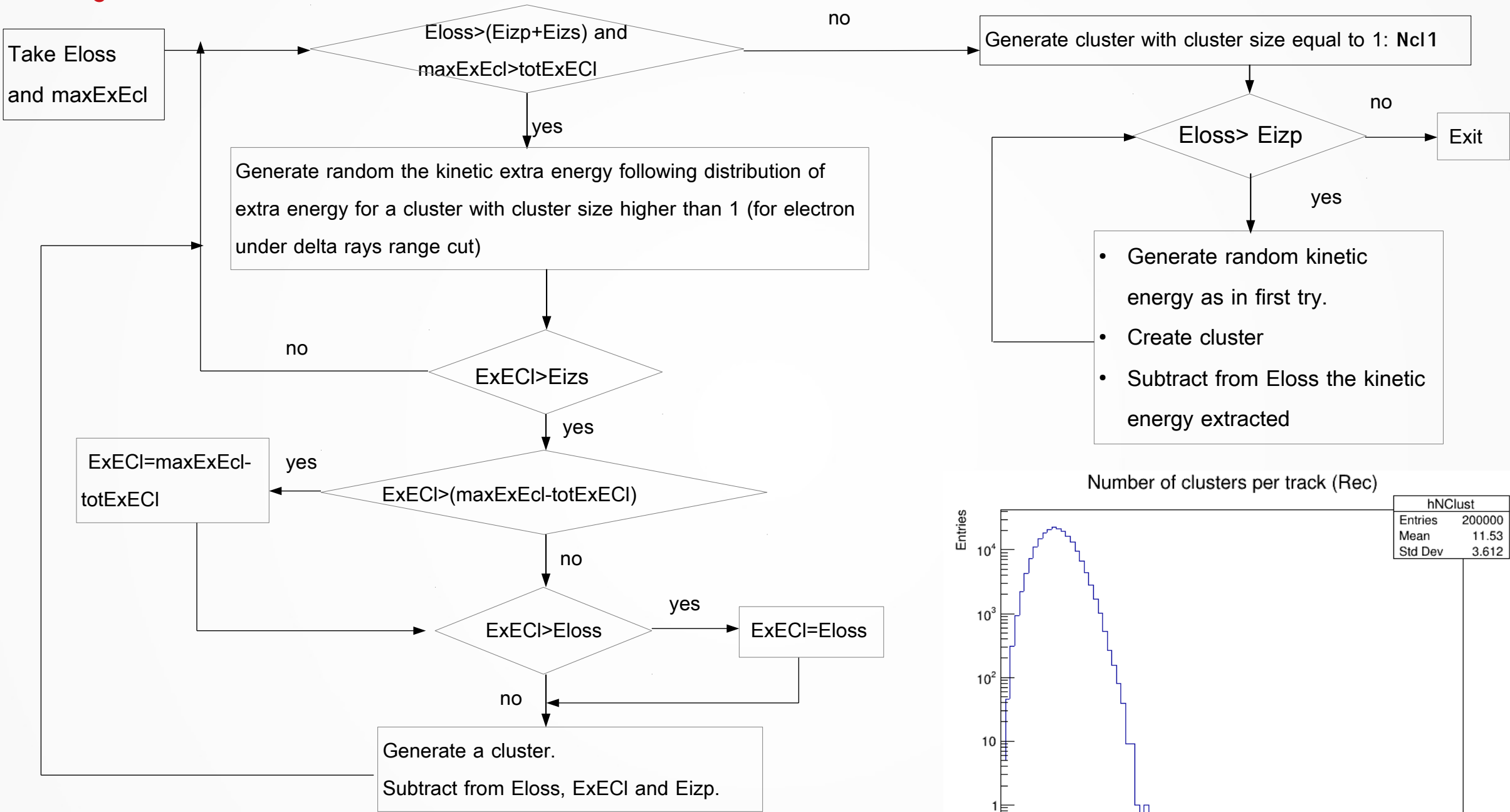
Here the ratio between the cluster containing a single electron and the cluster containing more than an electron.

The distribution is fitted with 1- decreasing exponential function plus two decreasing exponential functions.

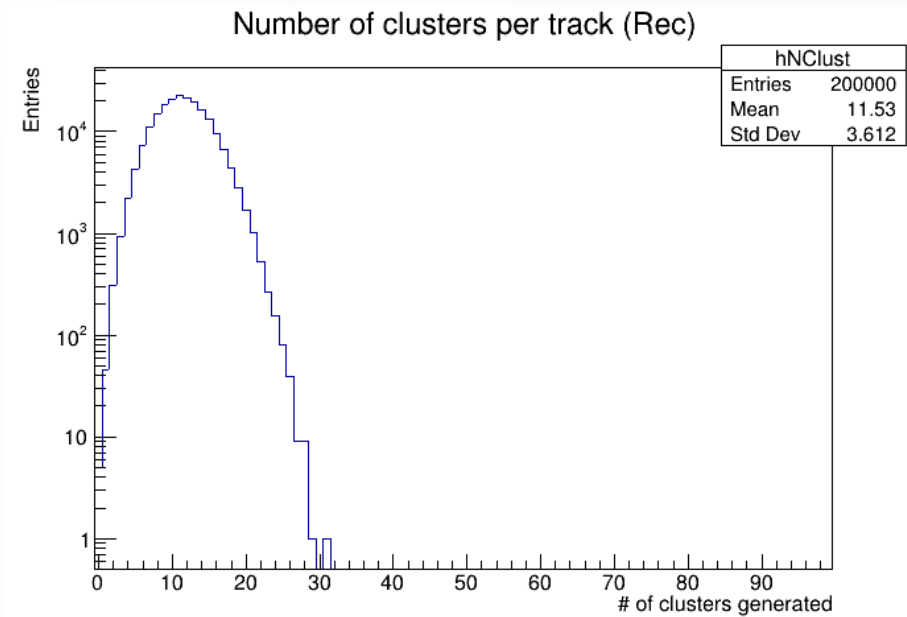
First algorithm



Second algorithm: reconstruction of number of cluster

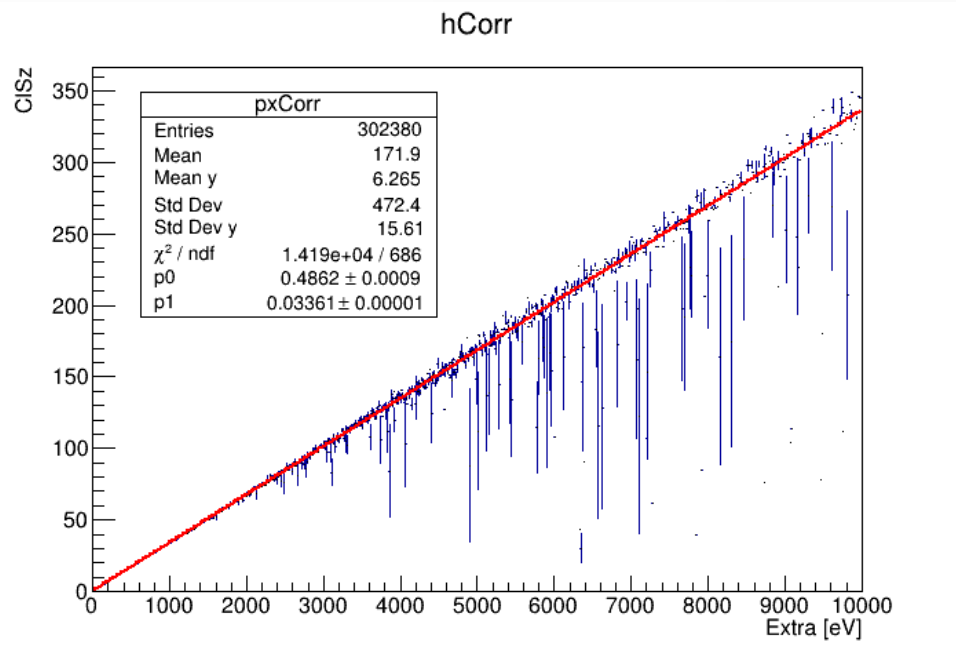
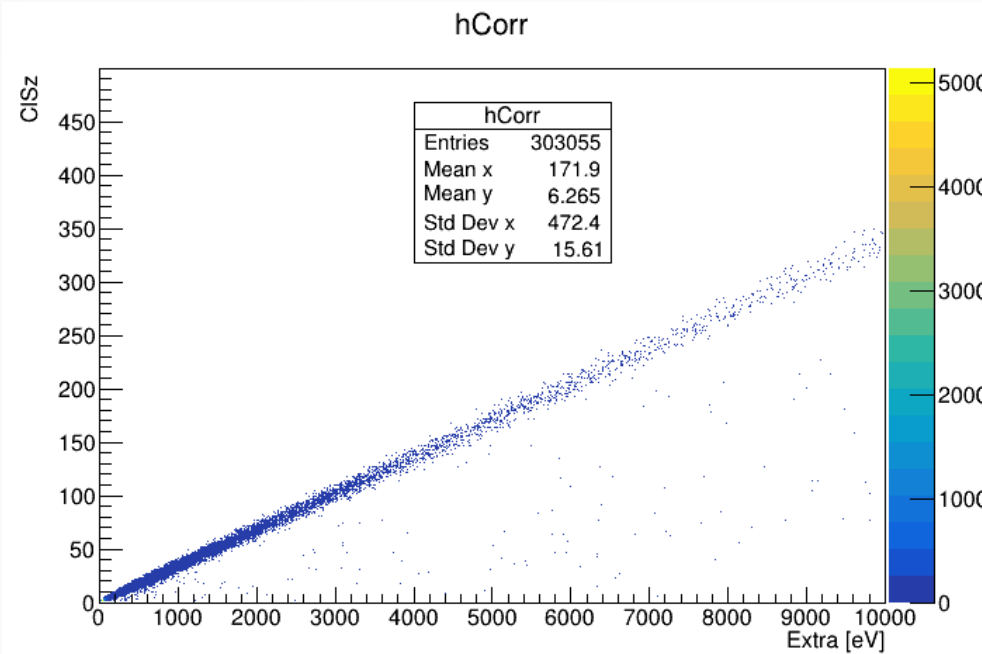


The evaluation of cluster with cluster size equal to one (Ncl1) remains the same for all the next attempts.



Second try: reconstruction of cluster size

We studied the relation between extra energy and cluster size for cluster with cluster size higher than 1 (delta rays are included).



We fit the correlation trend with a first degree polynomial and save the parameter p0 and p1.

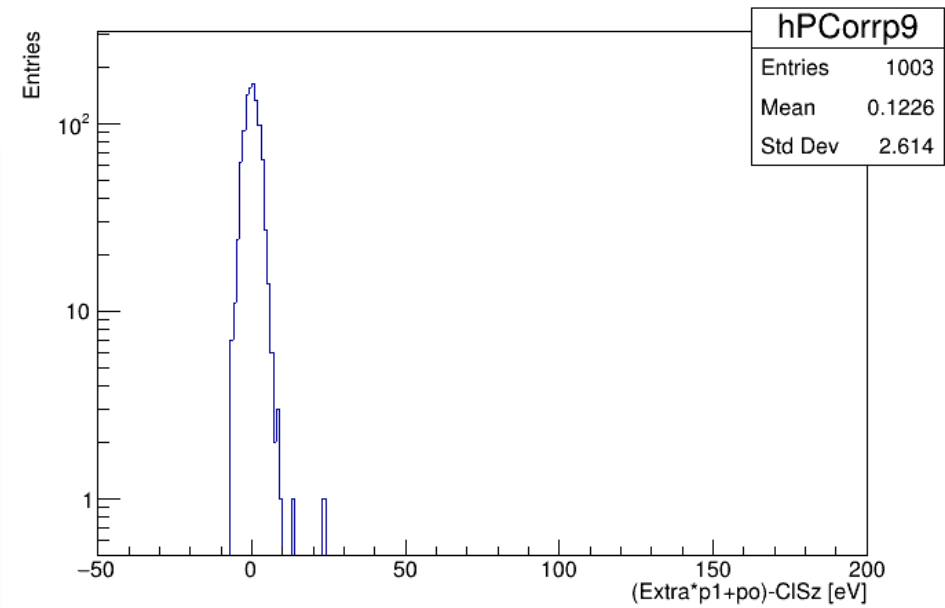
To evaluate the cluster size for cluster with more than one electron, we study the dispersion for different slice of extra energy up to delta rays cut range (1000 eV):

$$(Extra\ energy * p1 + p0) - ClSz.$$

The figure shows an example for extra energy between 900 and 1000 eV.

At the end, the cluster size is evaluated as :

$$(Extra\ energy * p1 + p0) \text{ minus } hPCorr \rightarrow GetRandom().$$



Same evaluation is performed for cluster size generated by delta rays.

We study the dispersion for different slice of extra energy above the value of delta rays cut range (1000 eV):

$$(Extra\ energy * p1 + p0) - CISz.$$

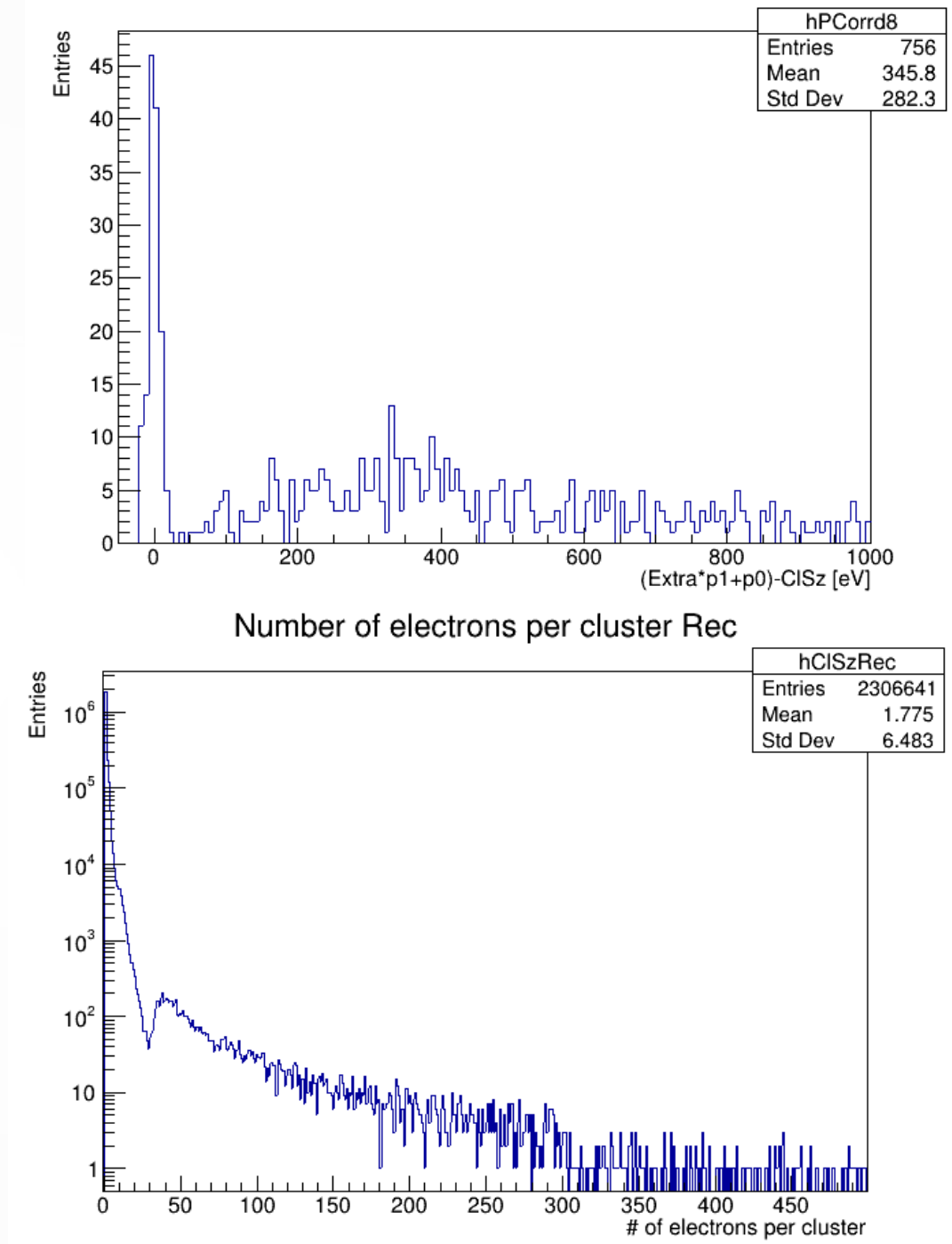
The figure shows a example for extra energy between 9 and 10 keV.

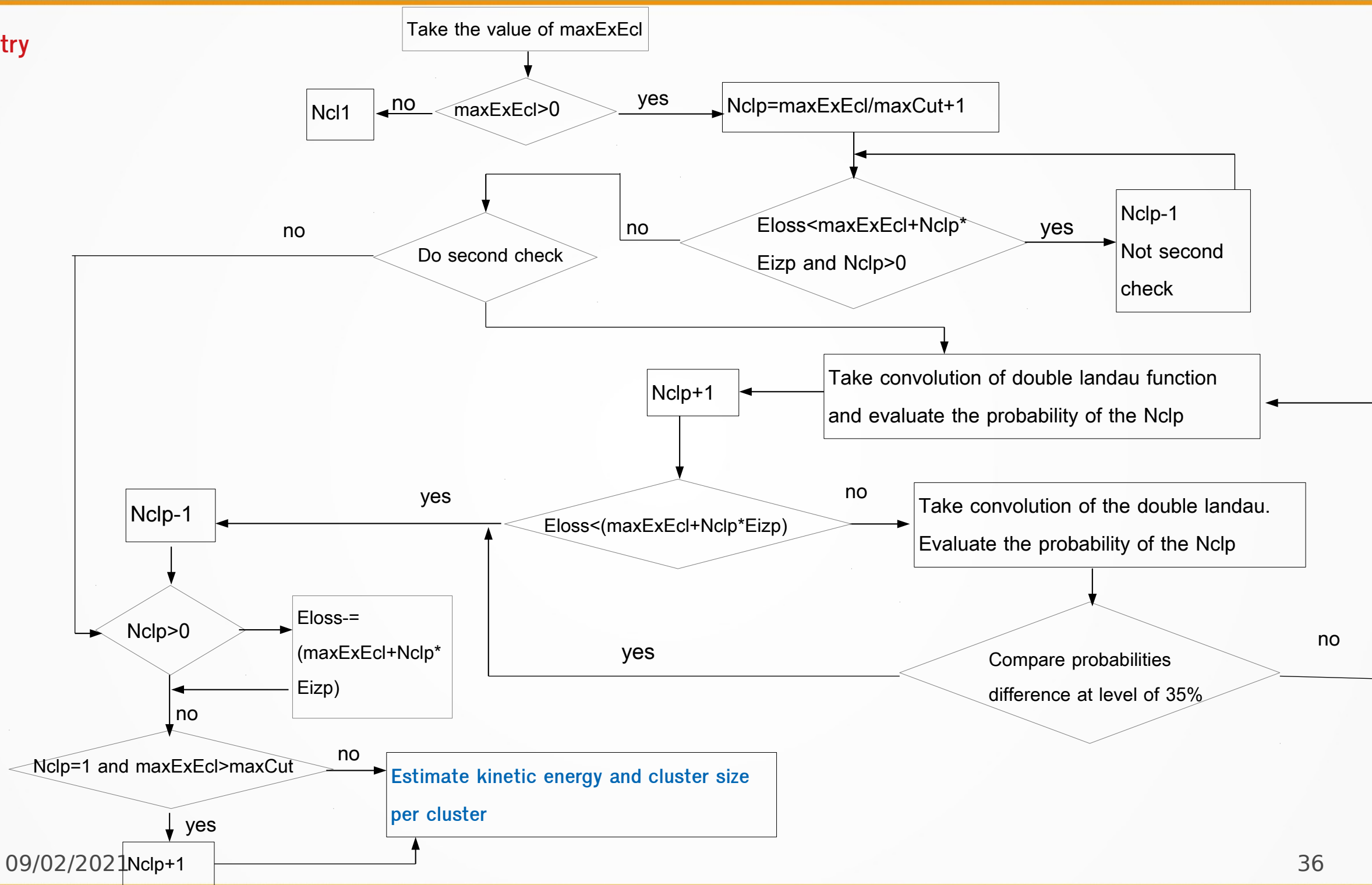
At the end, the cluster size is evaluated as :

$$(Extra\ energy * p1 + p0) \text{ minus } hPCorrd \rightarrow GetRandom().$$

The result is better than the first try, besides the depression remains.

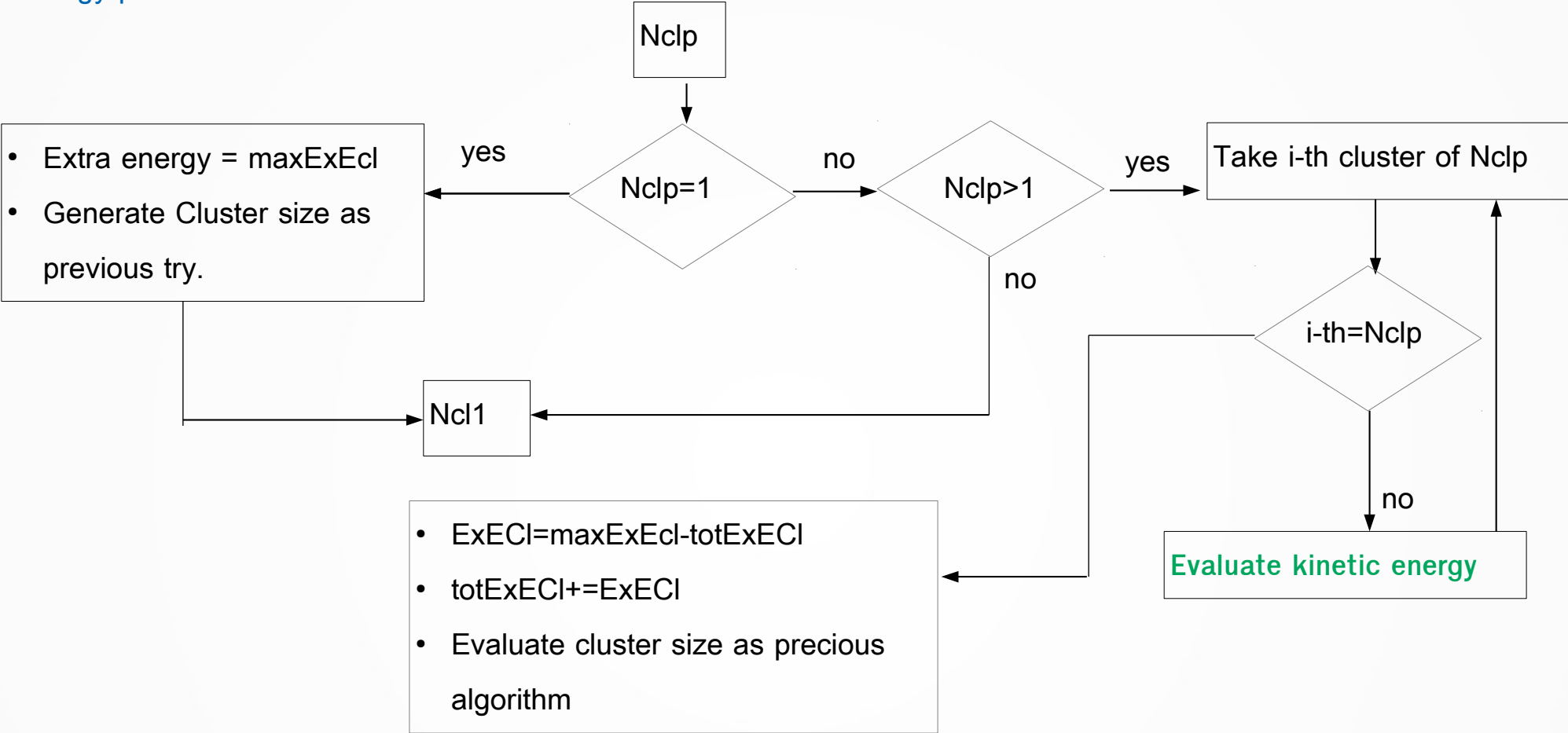
The evaluation of cluster size remains the same for all next attempts.





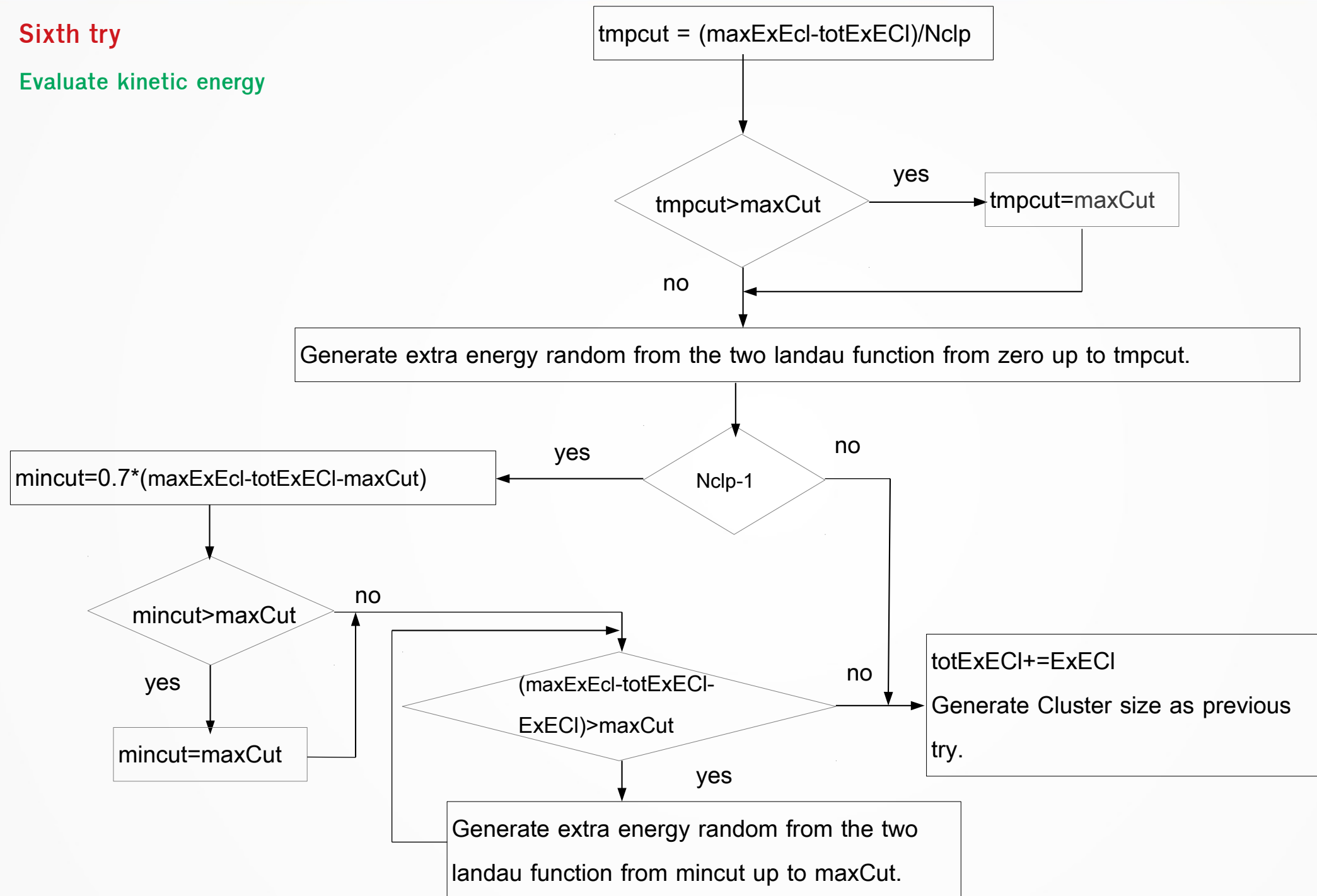
Sixth try

Estimate kinetic energy per cluster and cluster size

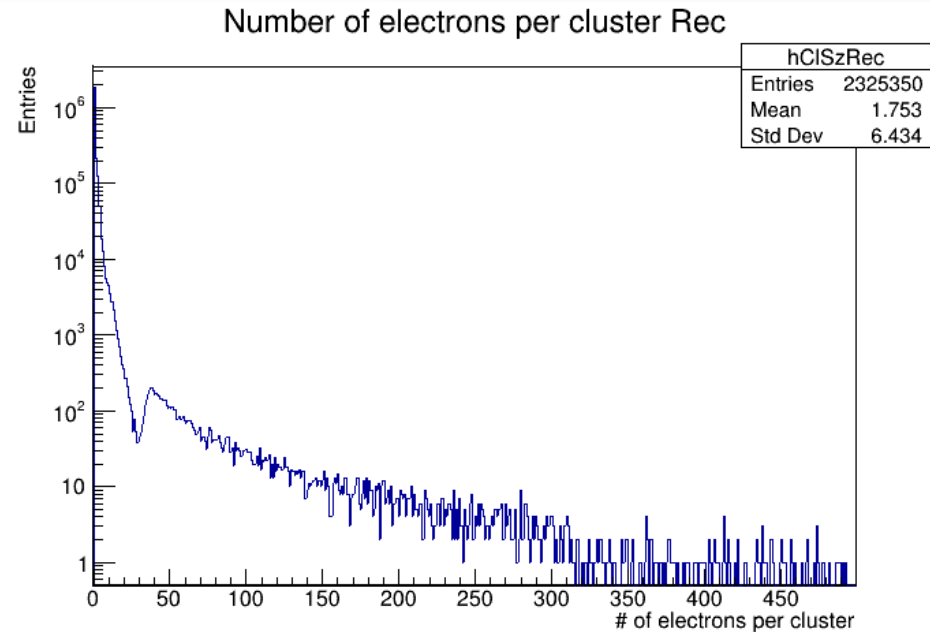
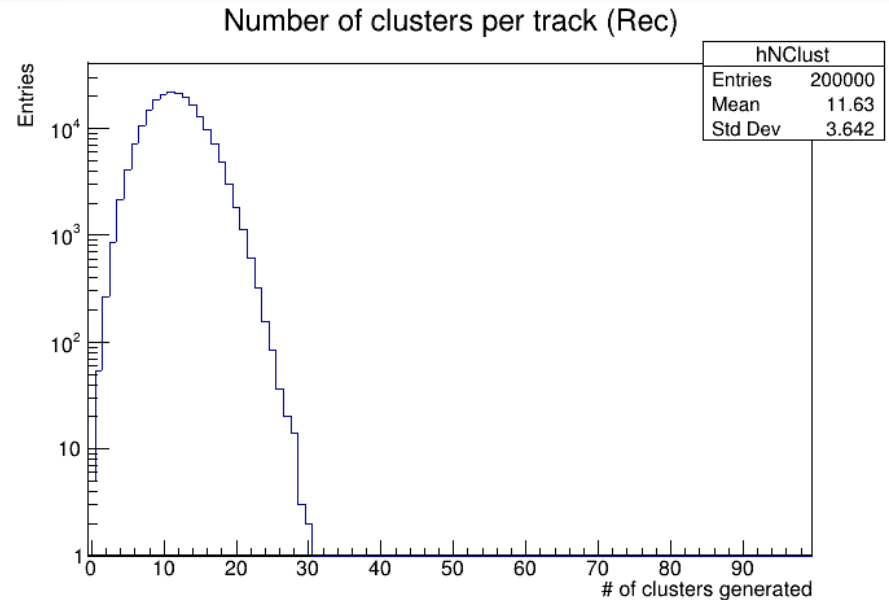


Sixth try

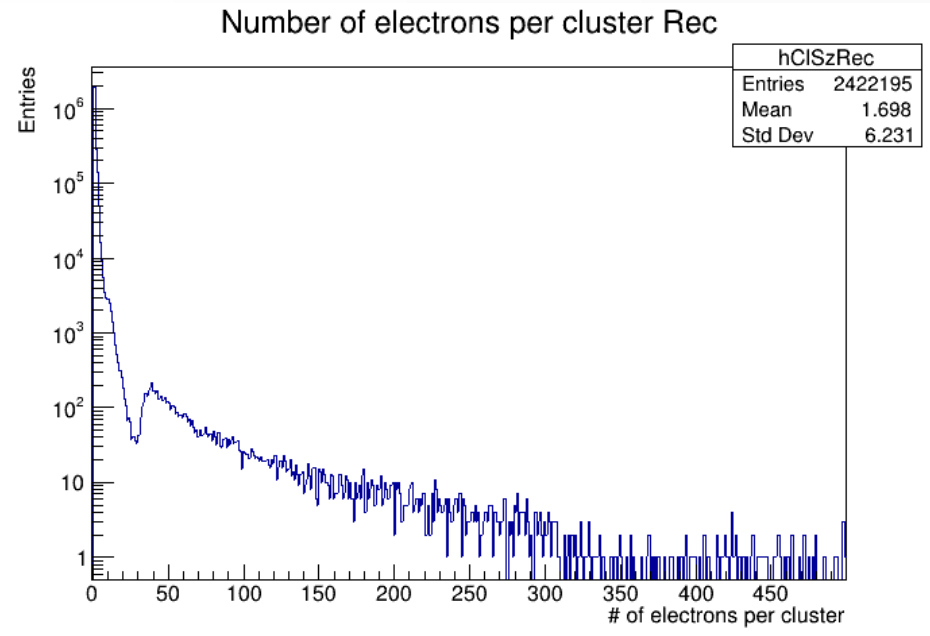
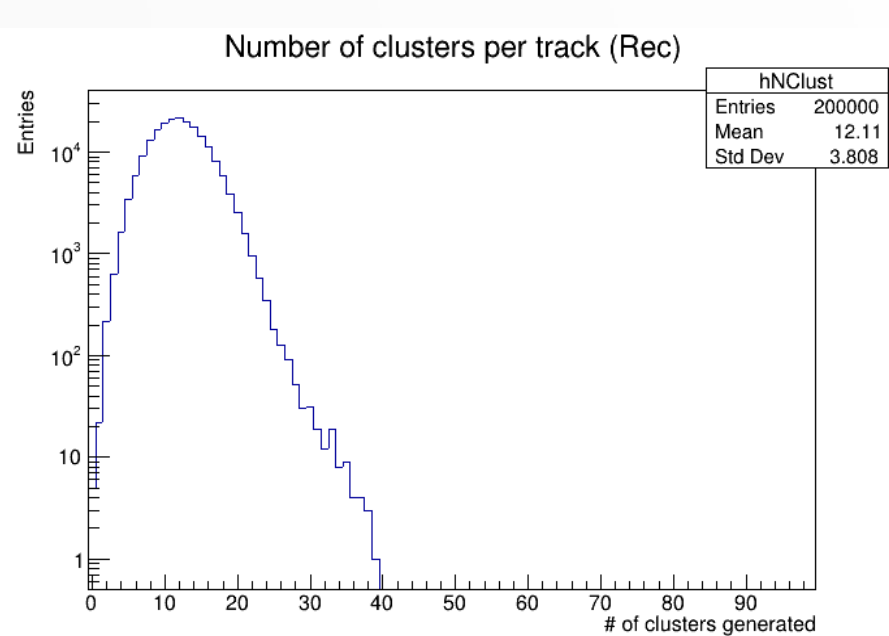
Evaluate kinetic energy



Fourth try

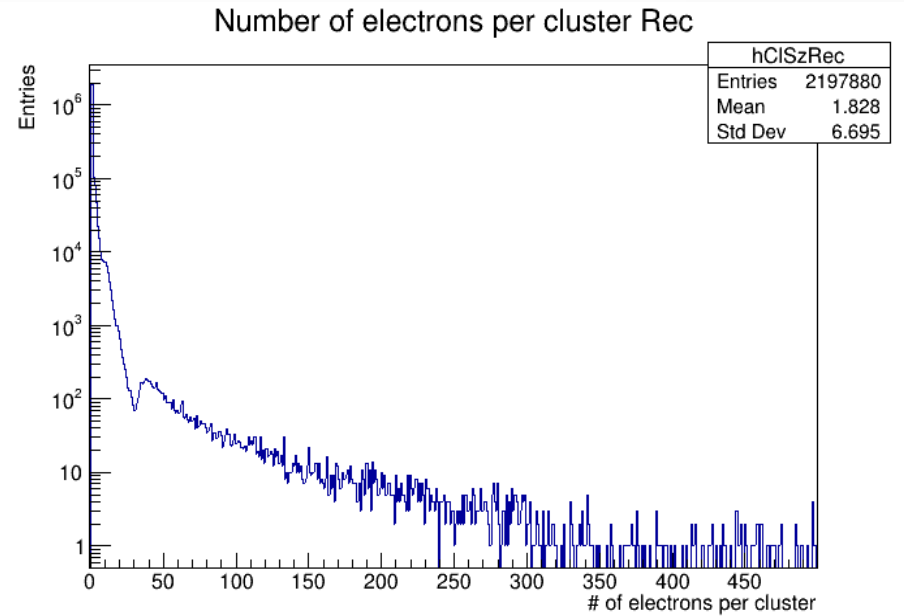
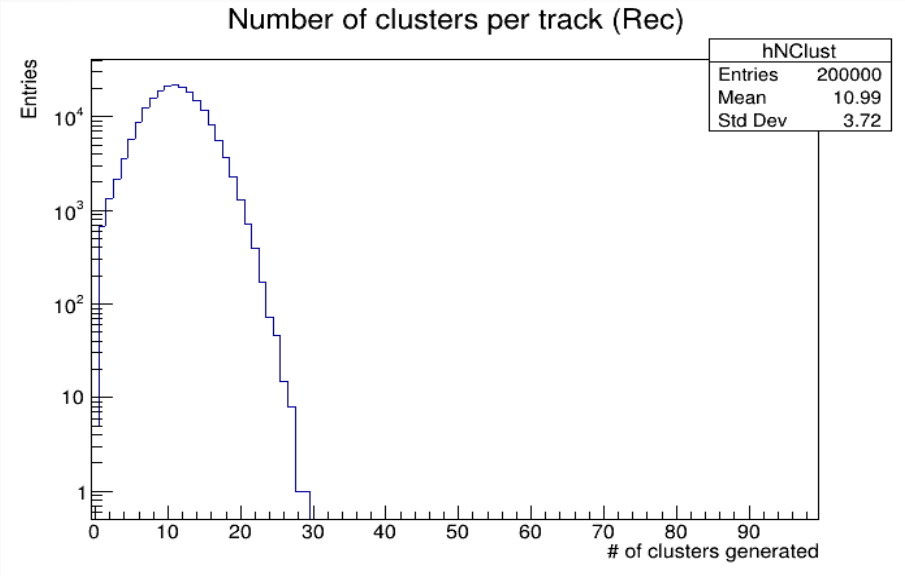


Fifth try

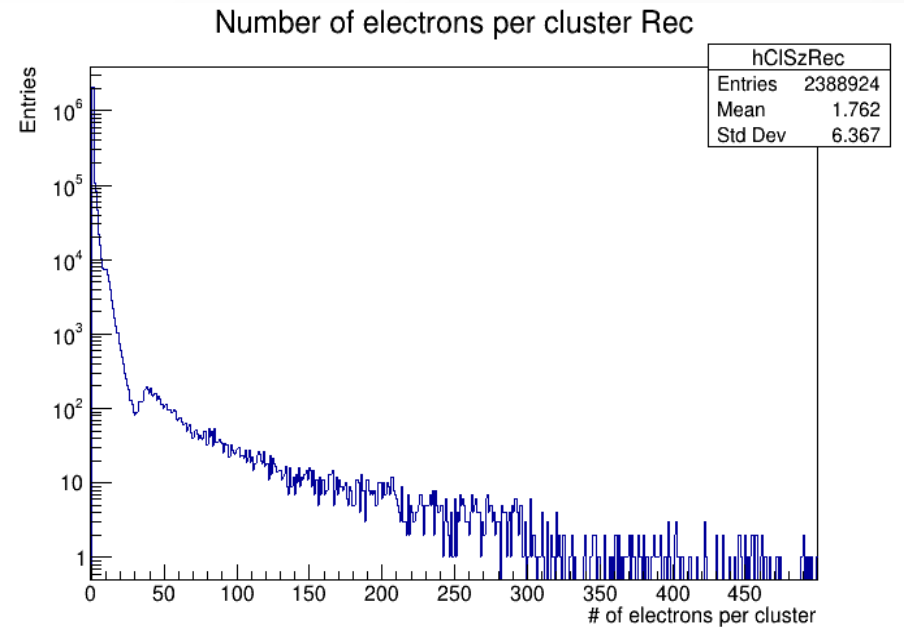
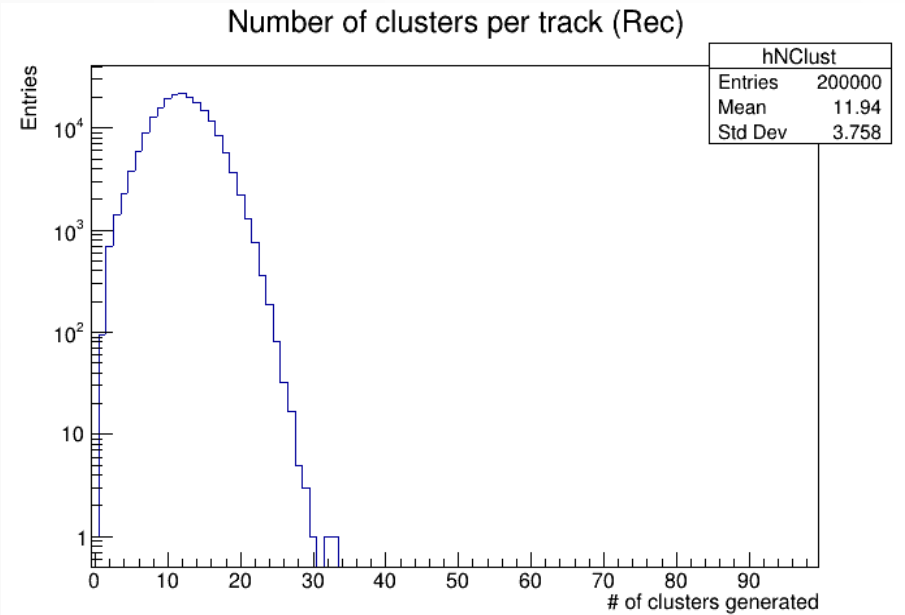


We made other algorithm trying to better reproduce the cluster size distribution.

Third try: without correction

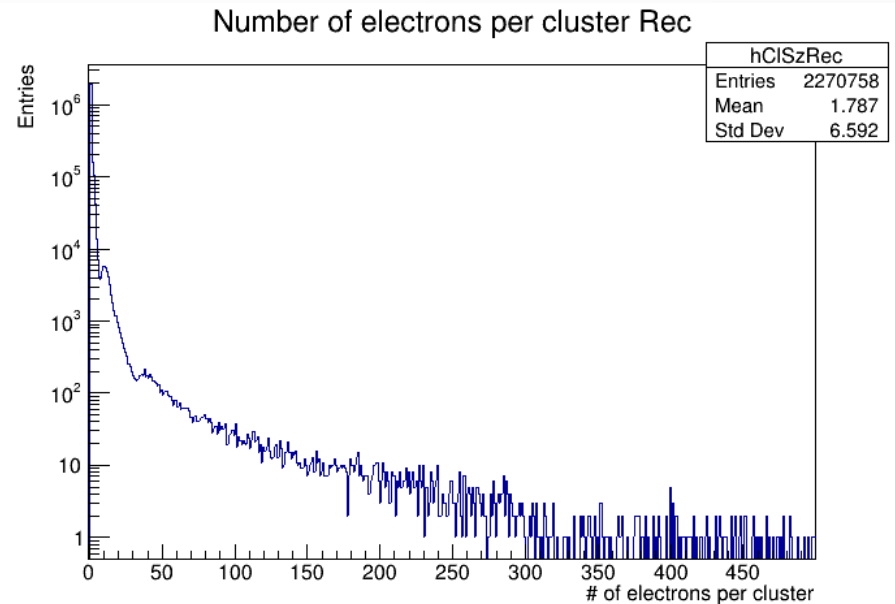
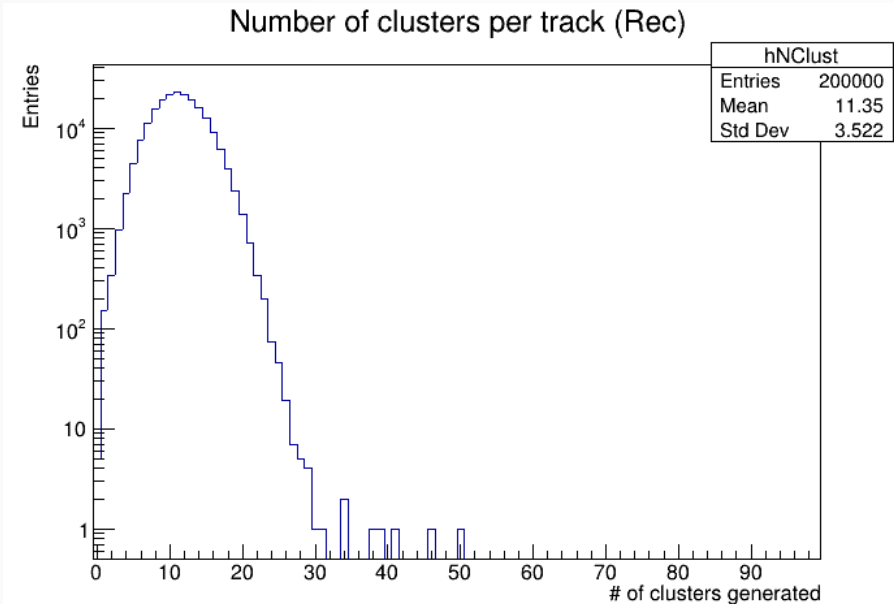


Third try: with correction

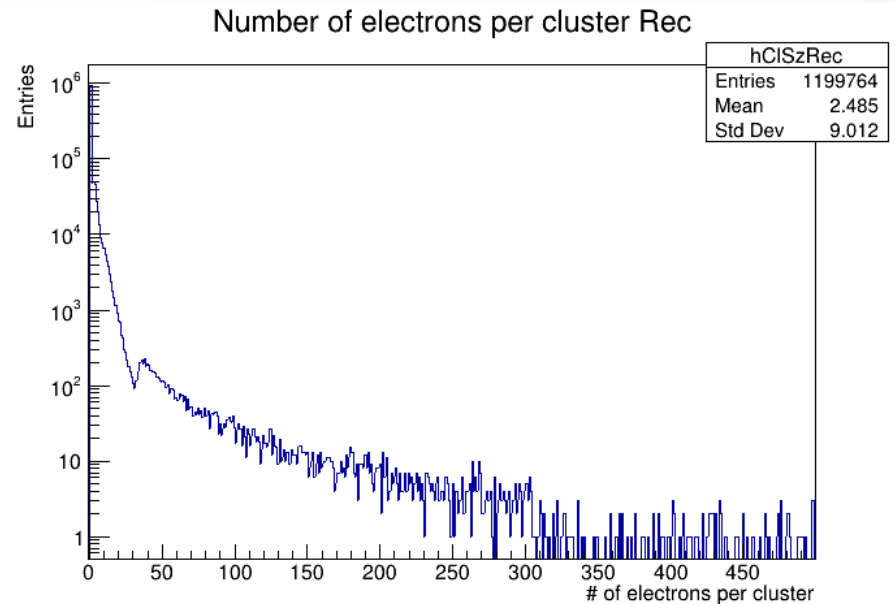
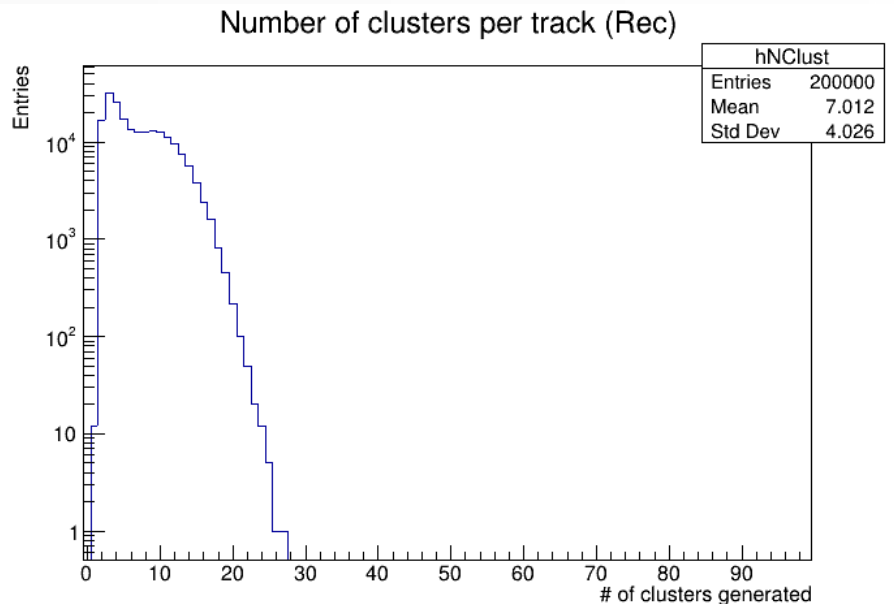


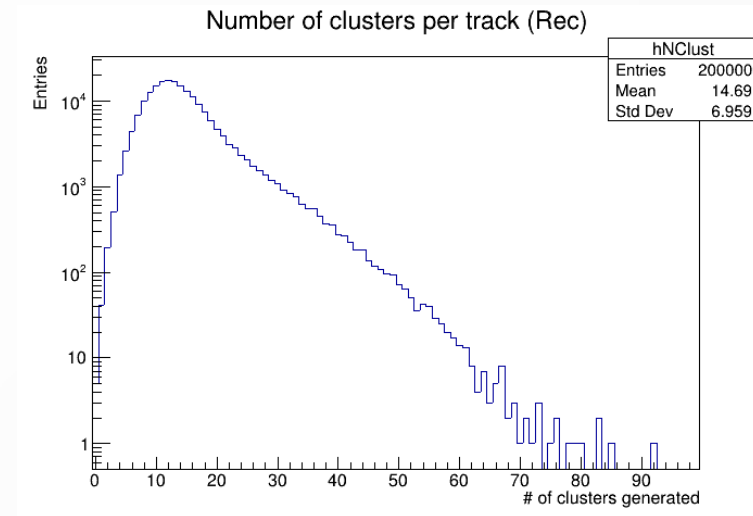
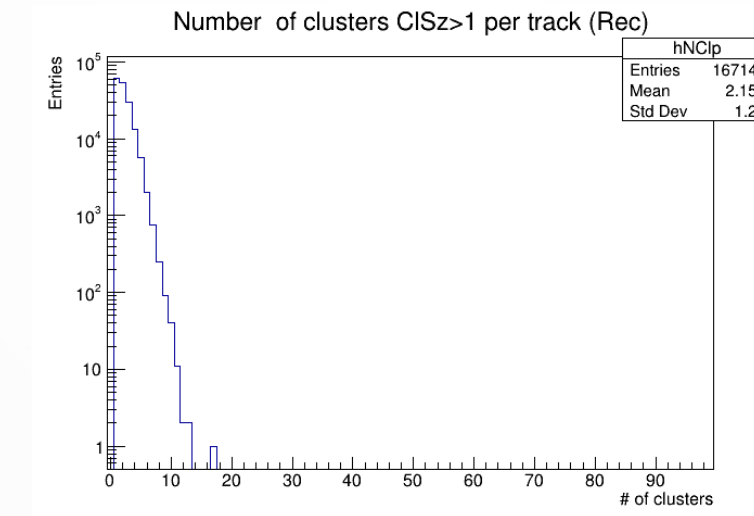
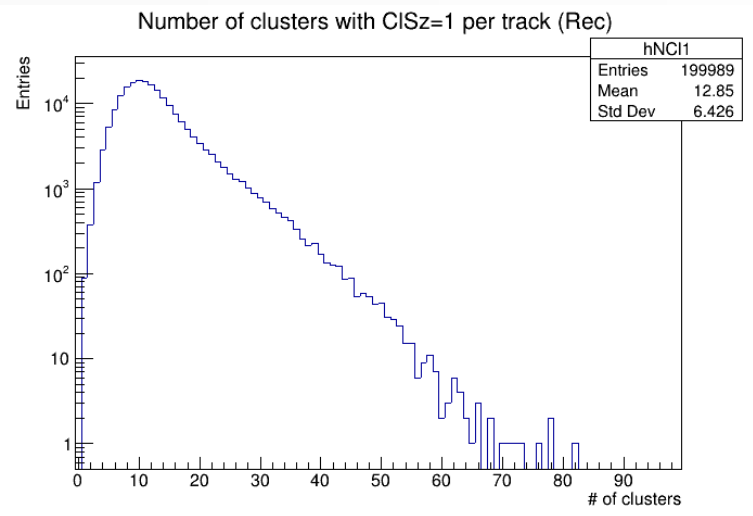
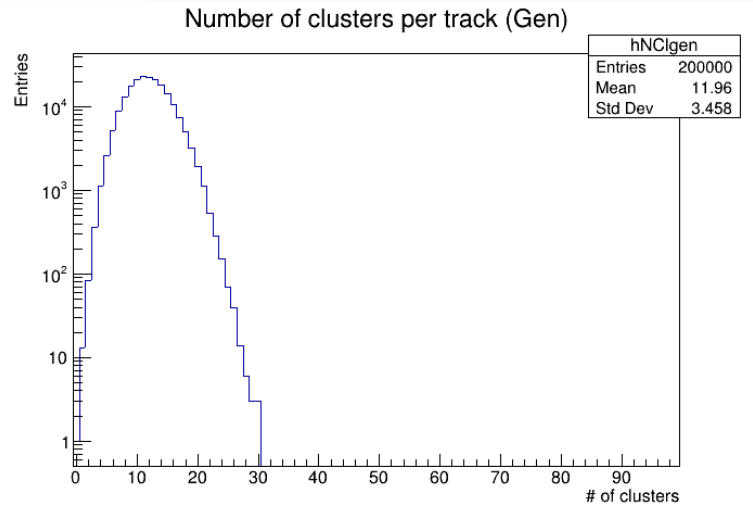
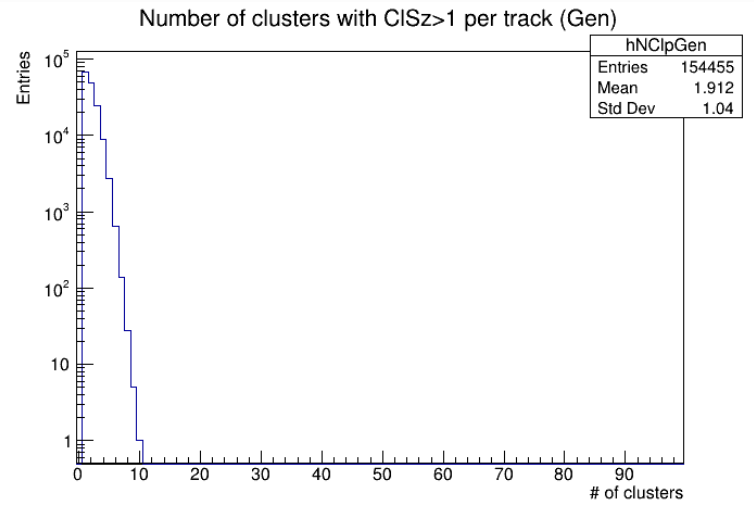
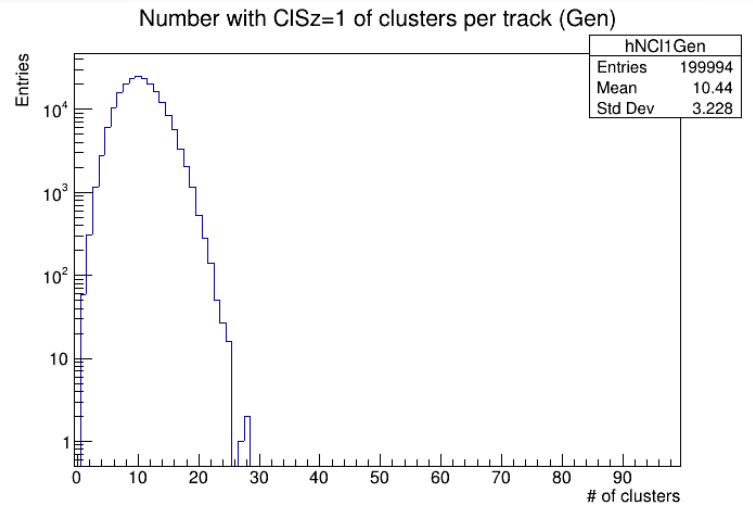
Sixth try

The sixth algorithm follows a different methodology. Indeed it uses the total kinetic energy of the event to evaluate a priori the number of cluster, applying the most likelihood criterium.

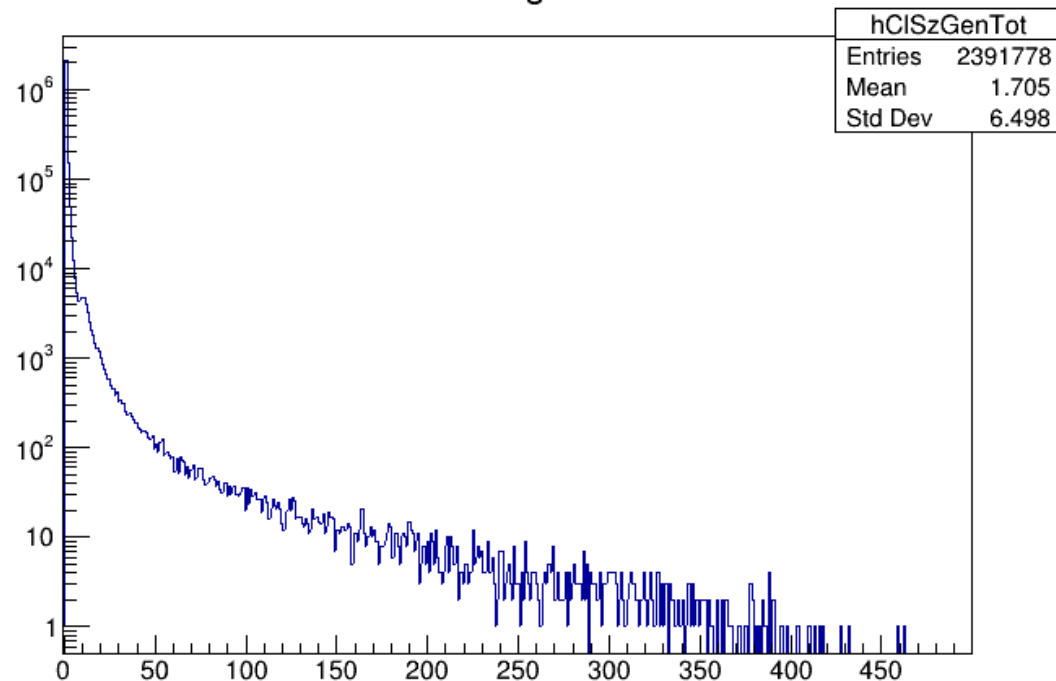


Seventh try

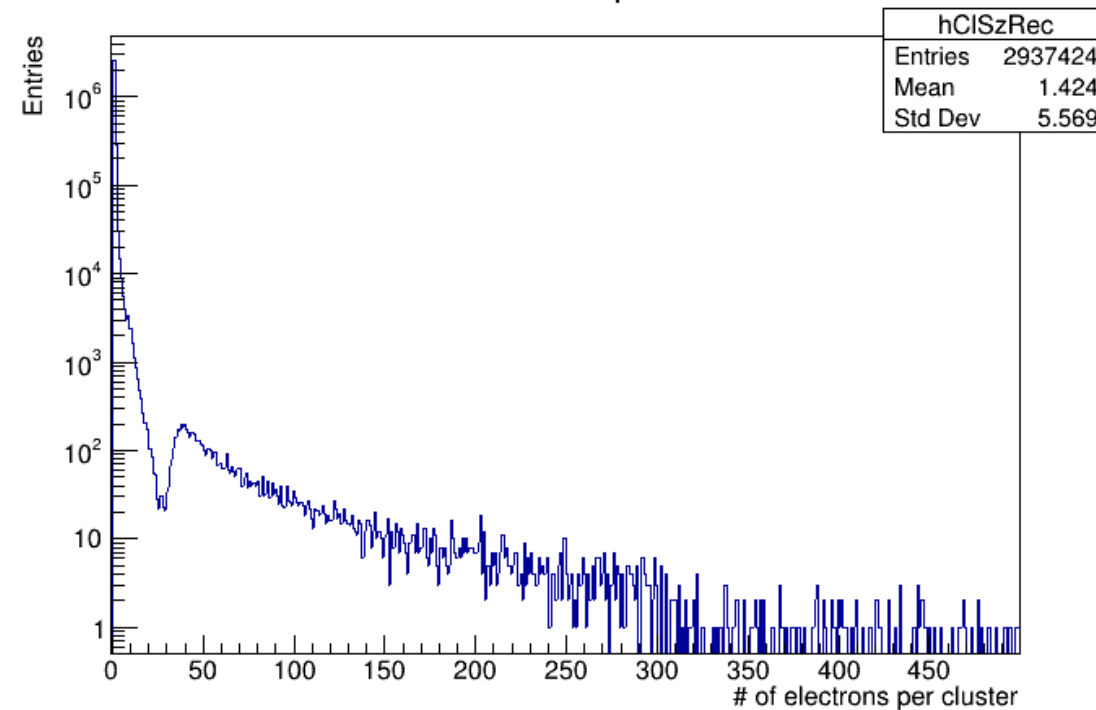


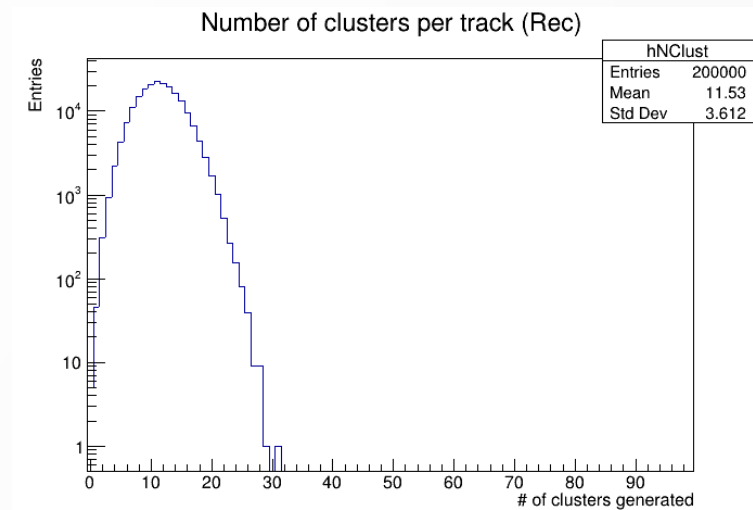
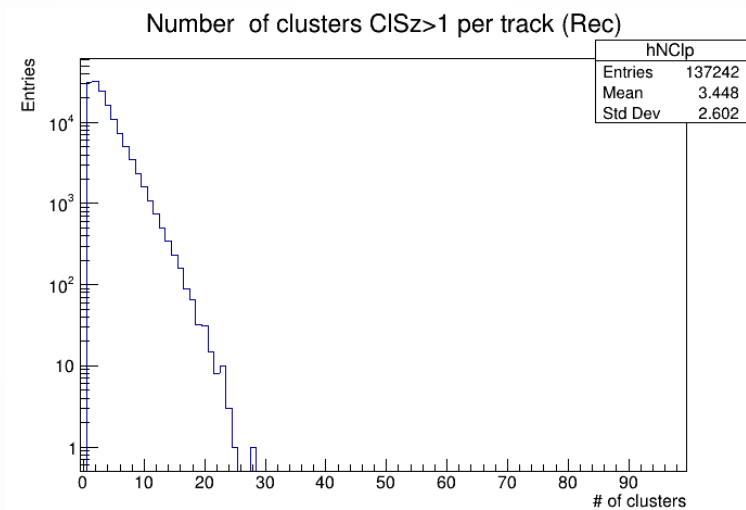
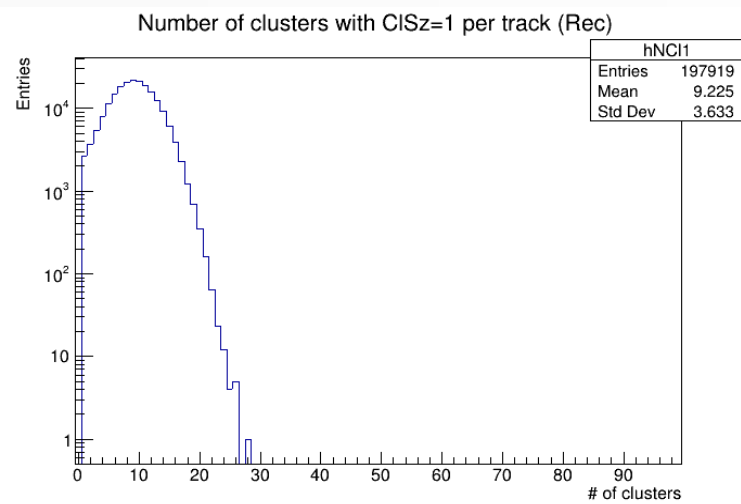
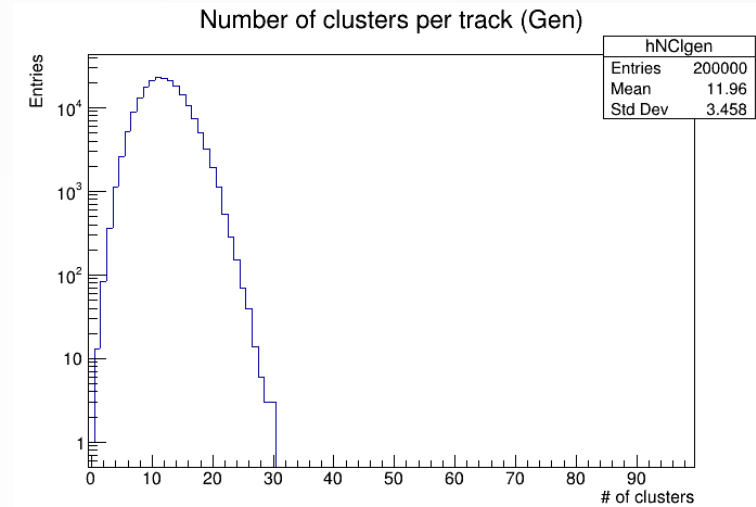
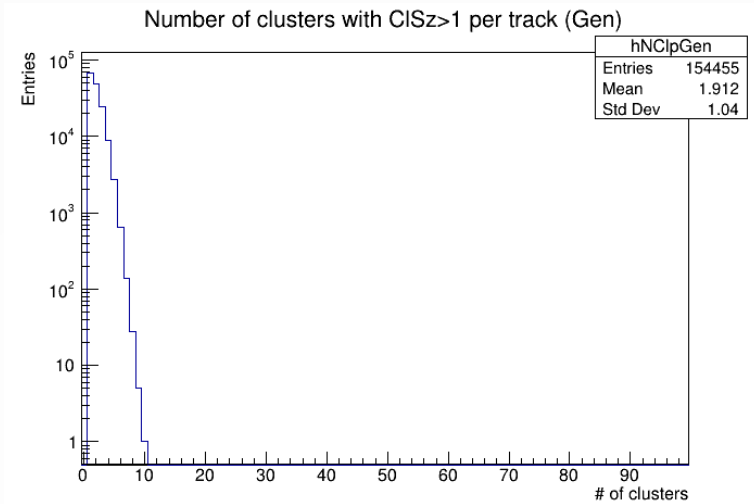
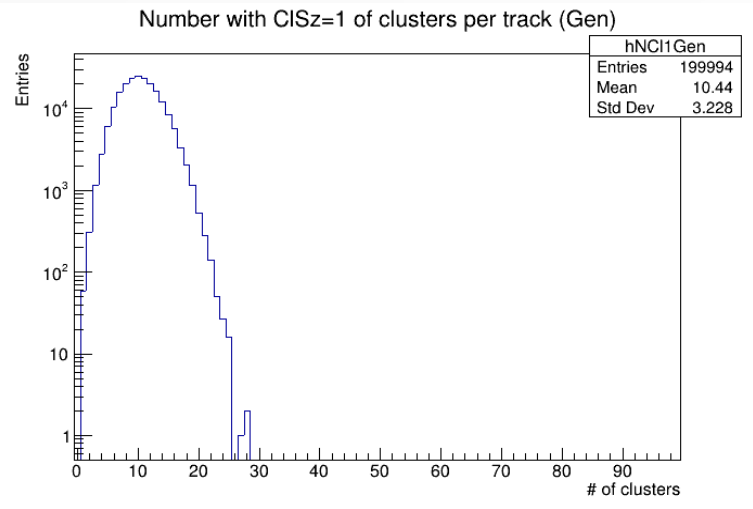


Total CISz generated

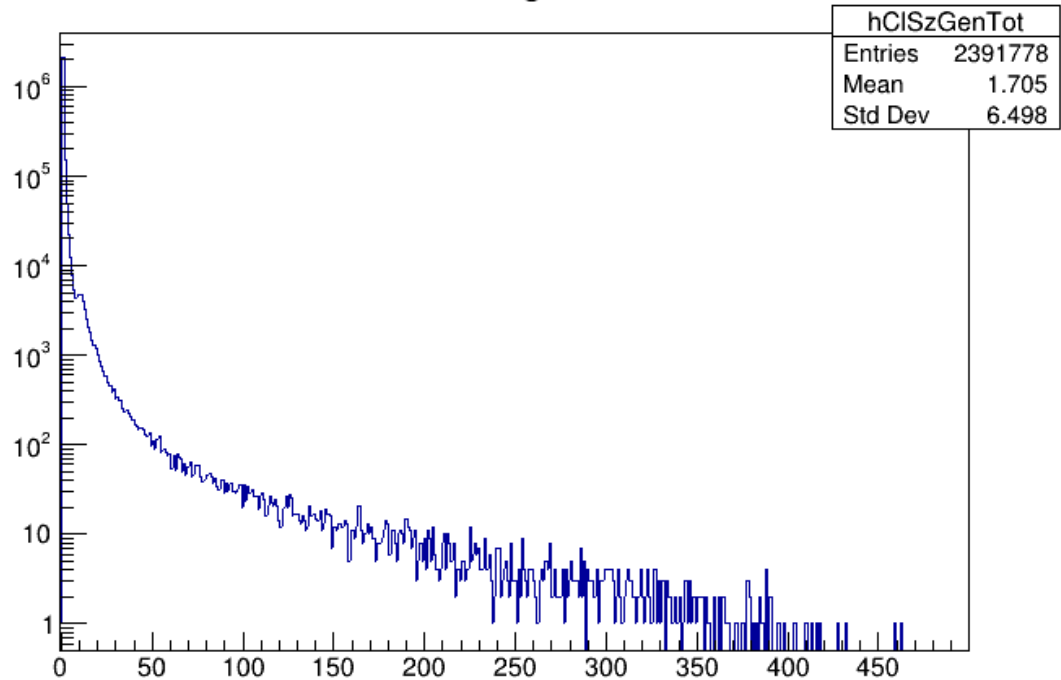


Number of electrons per cluster Rec

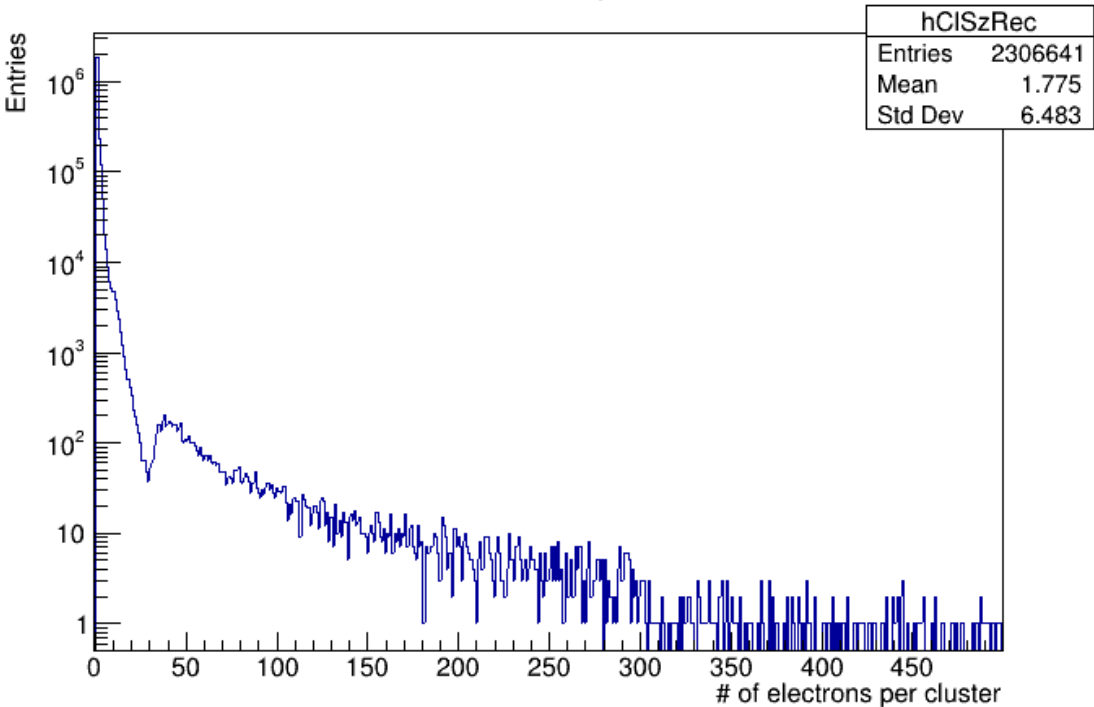


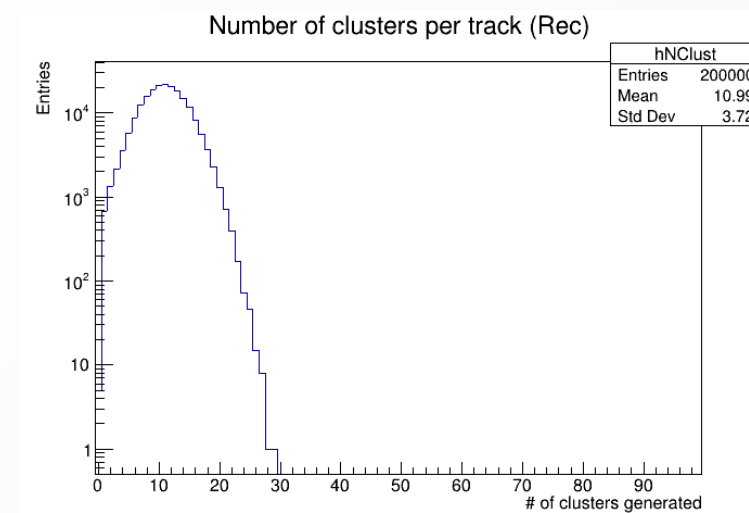
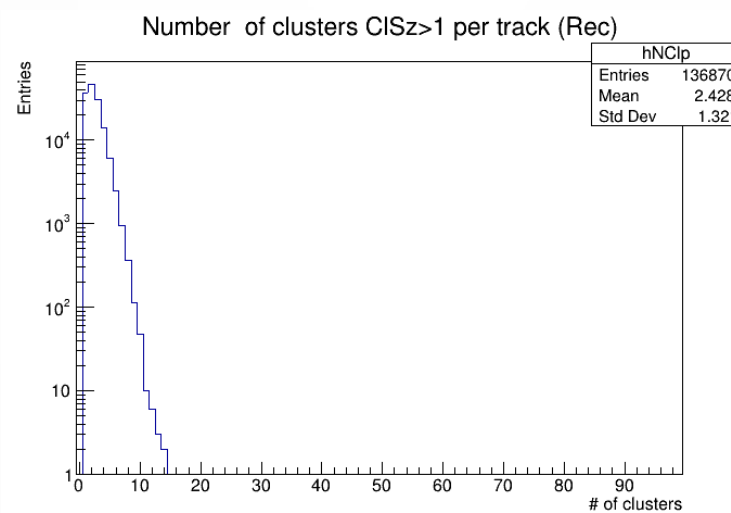
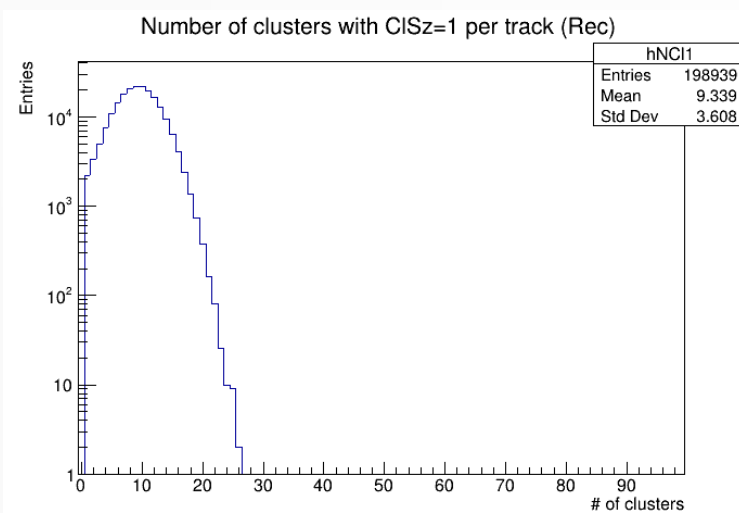
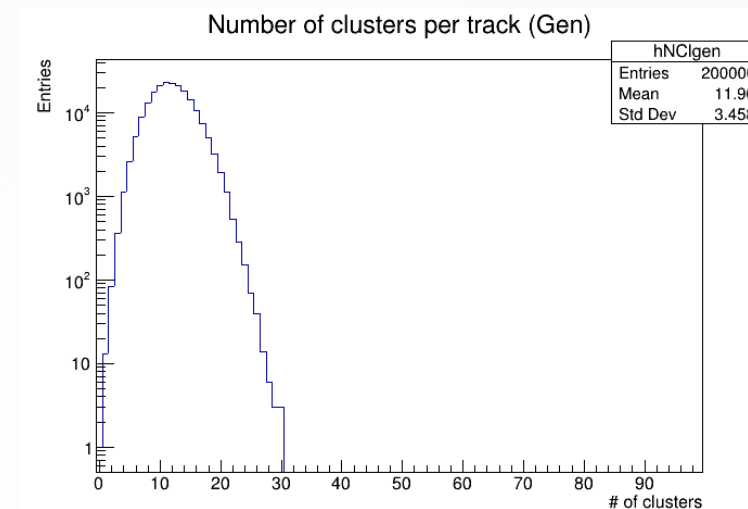
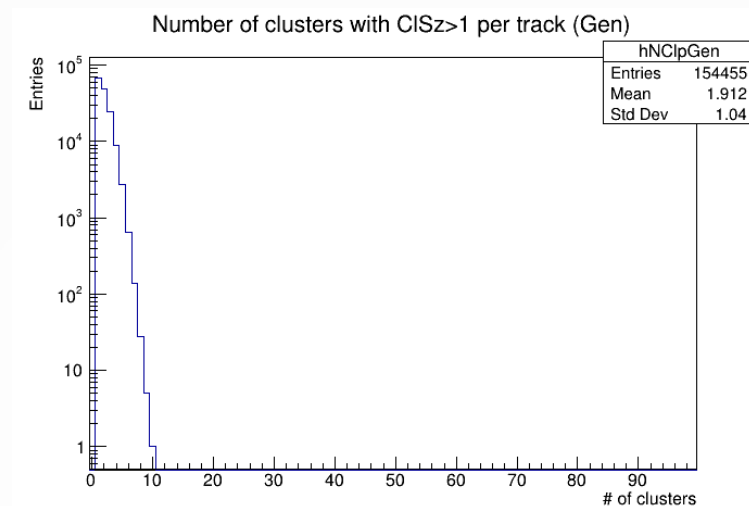
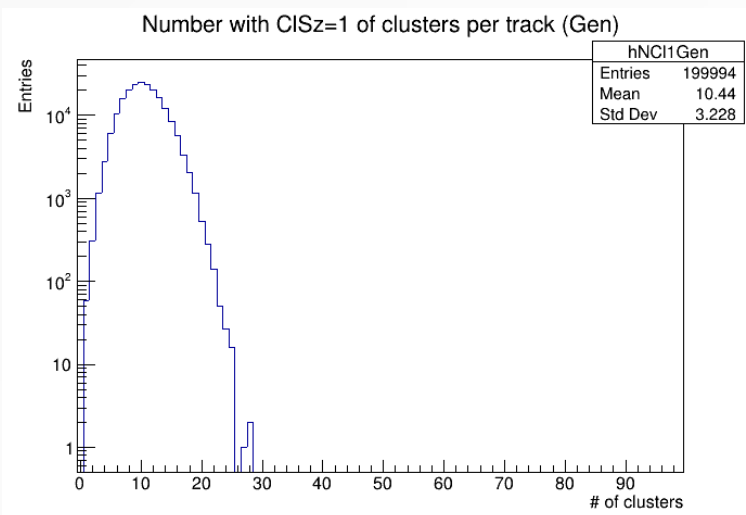


Total CISz generated

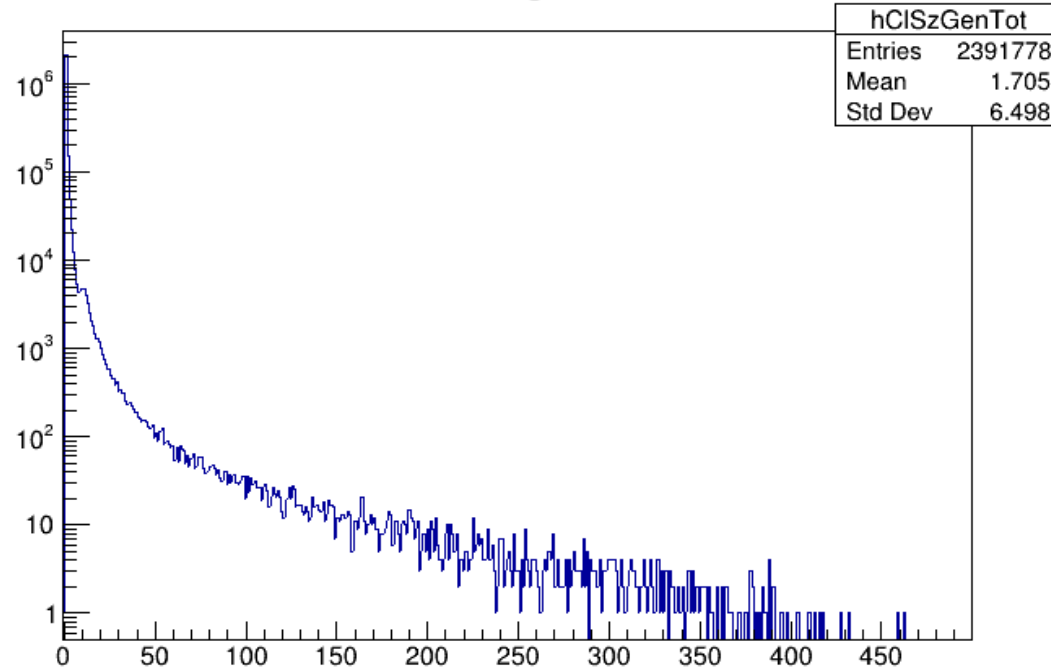


Number of electrons per cluster Rec

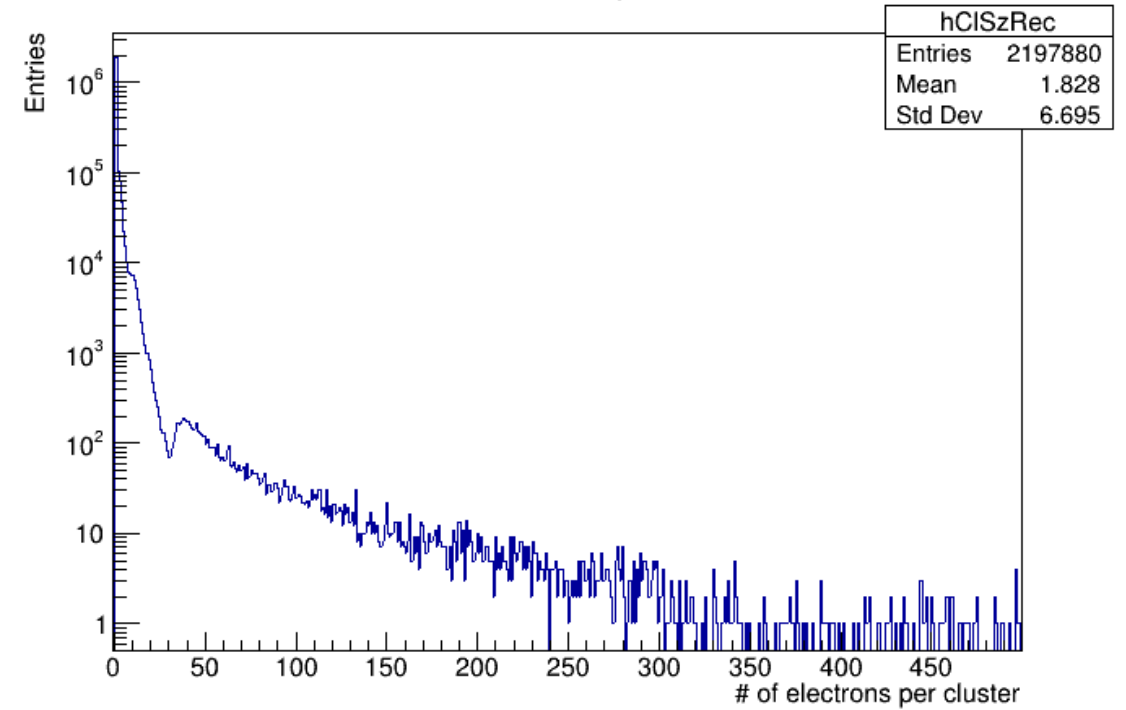


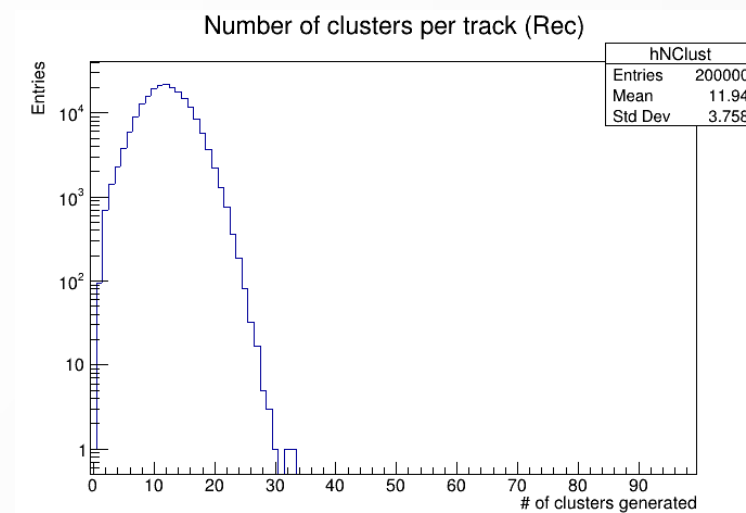
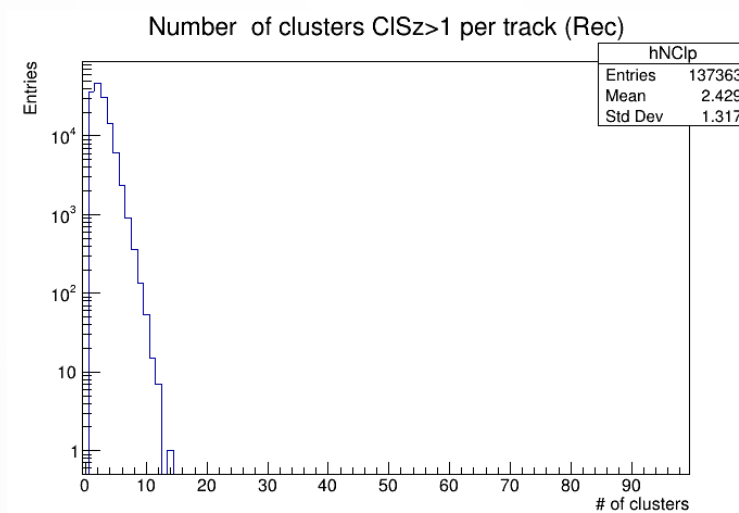
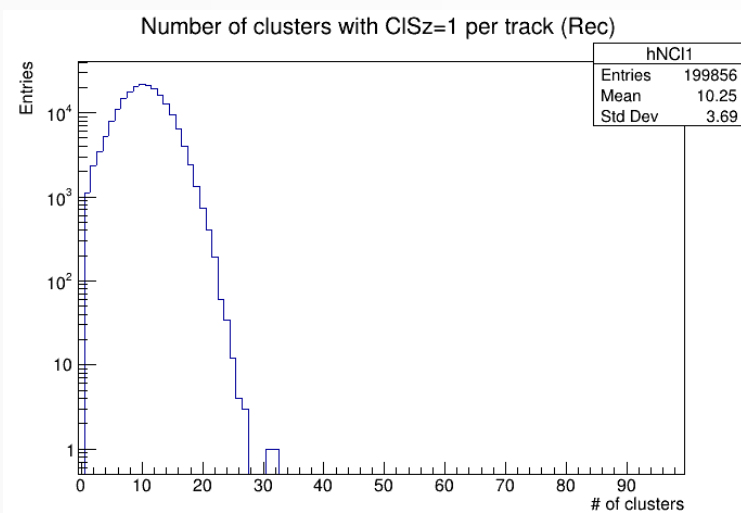
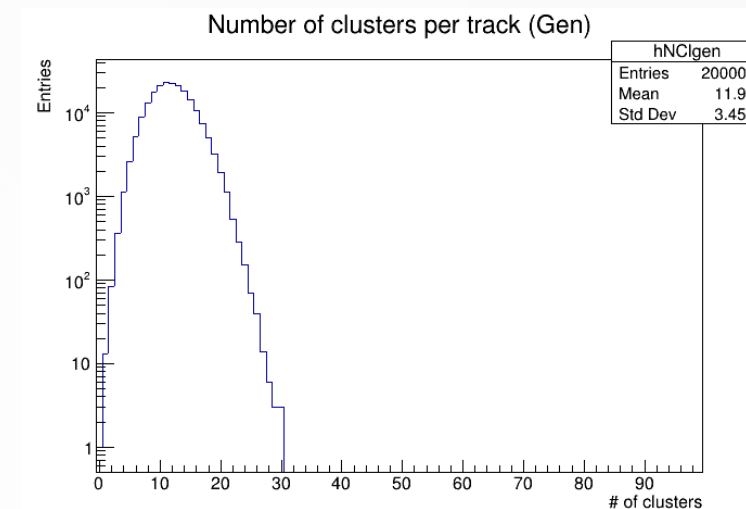
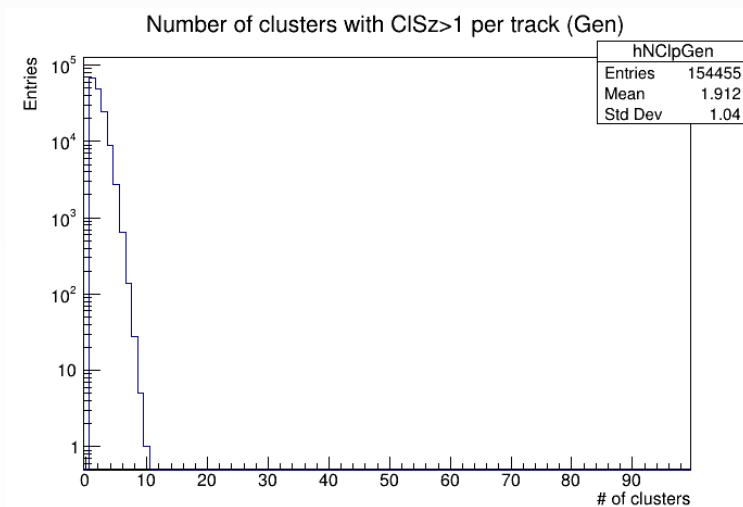
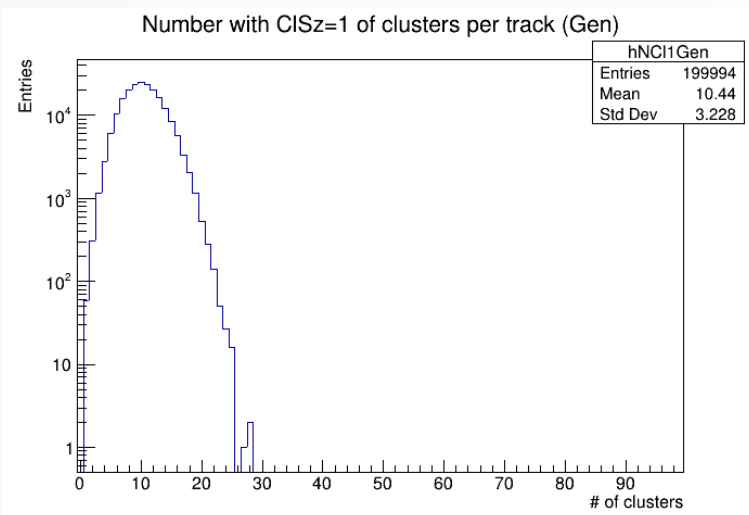


Total CIsZ generated

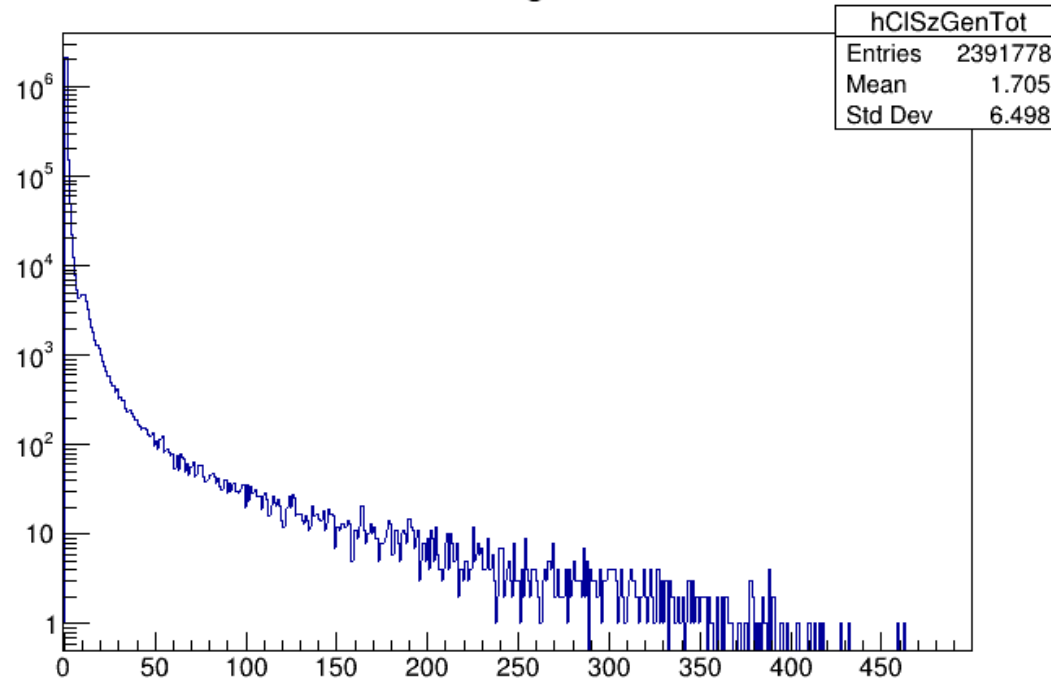


Number of electrons per cluster Rec

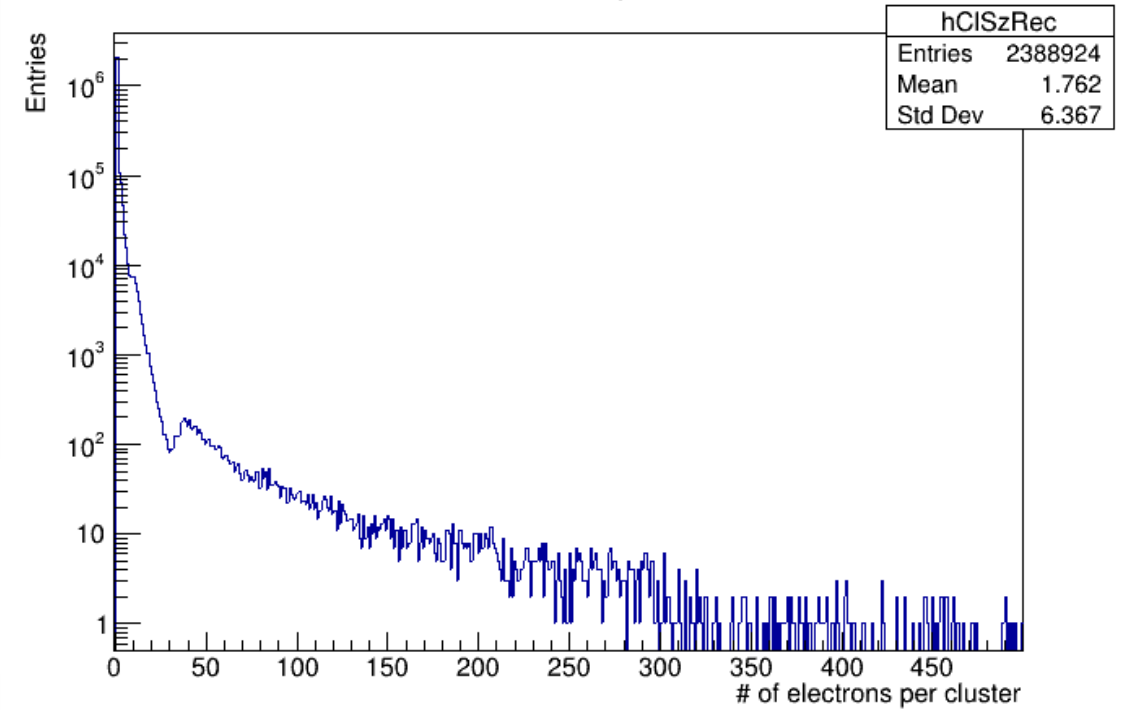


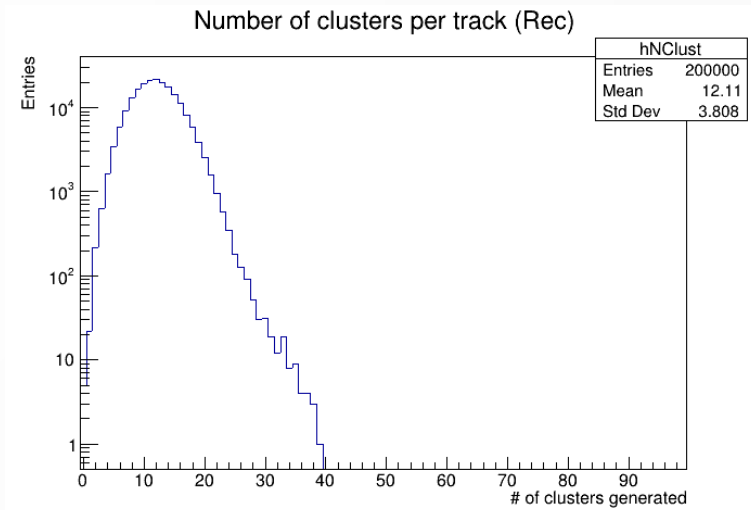
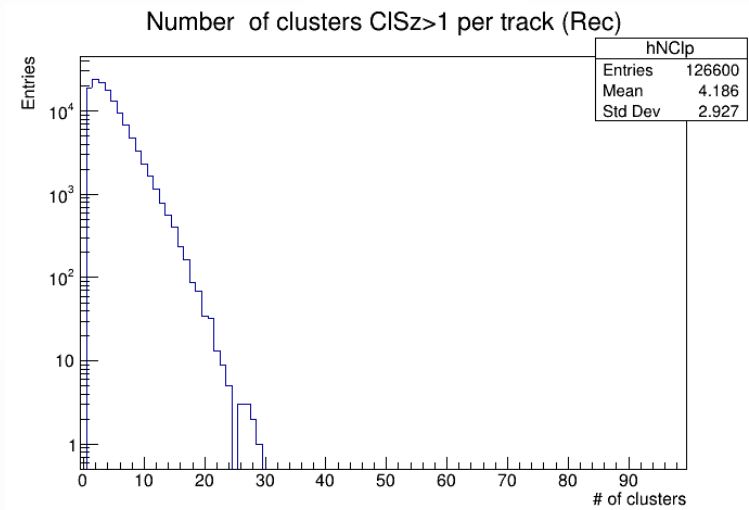
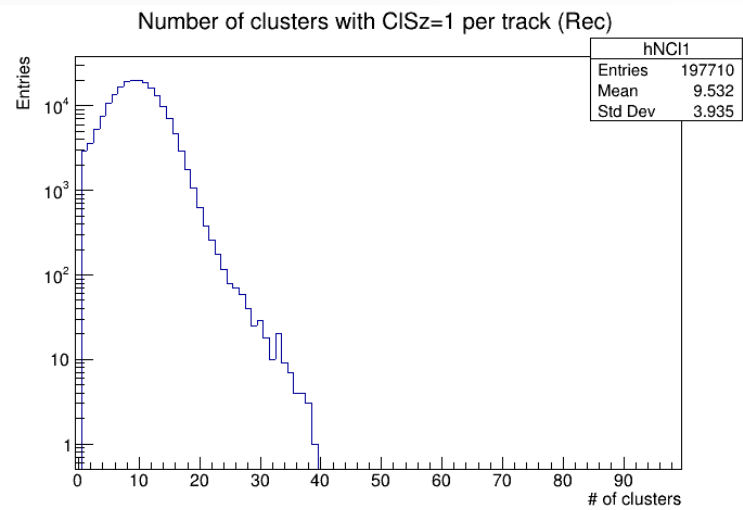
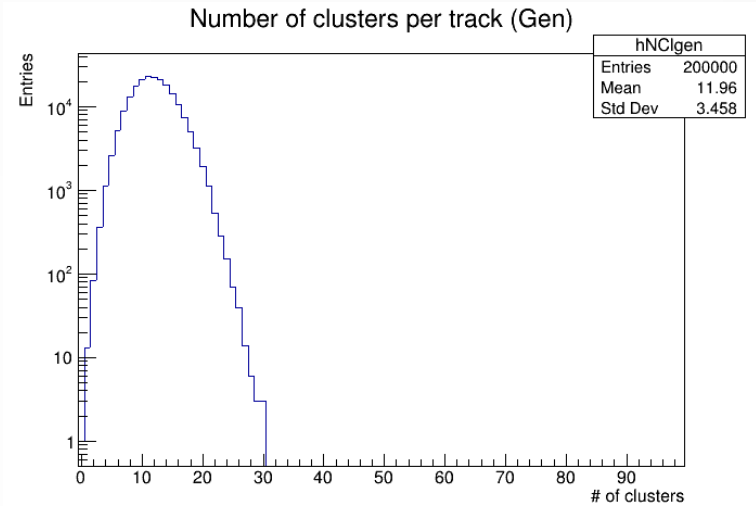
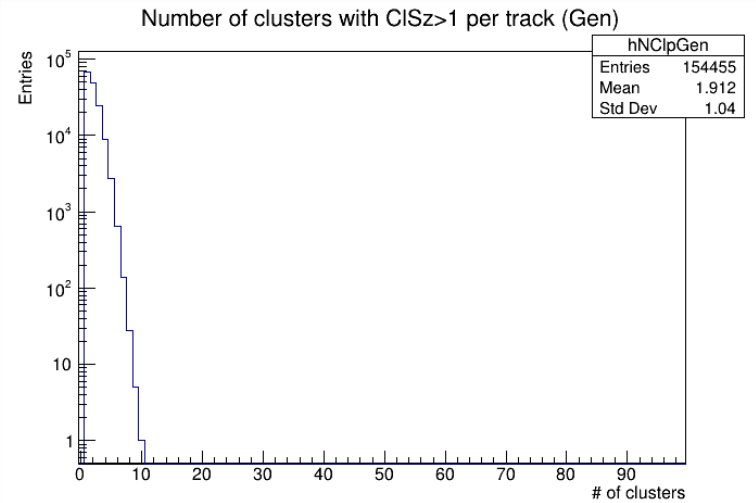
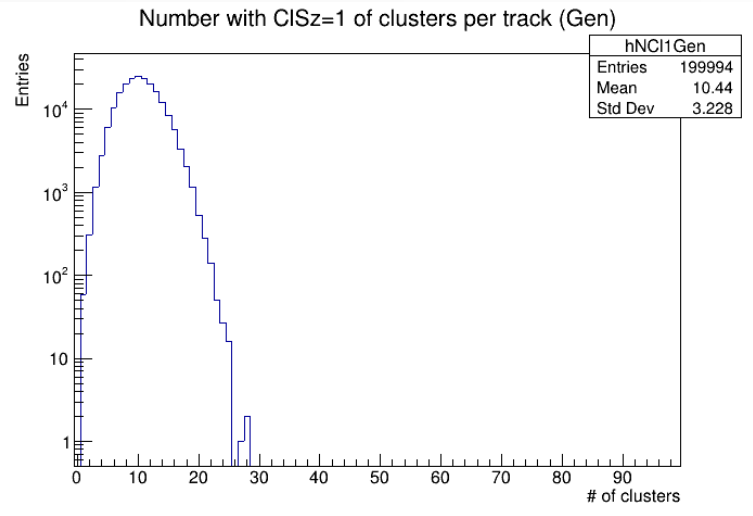


Total CIsZ generated

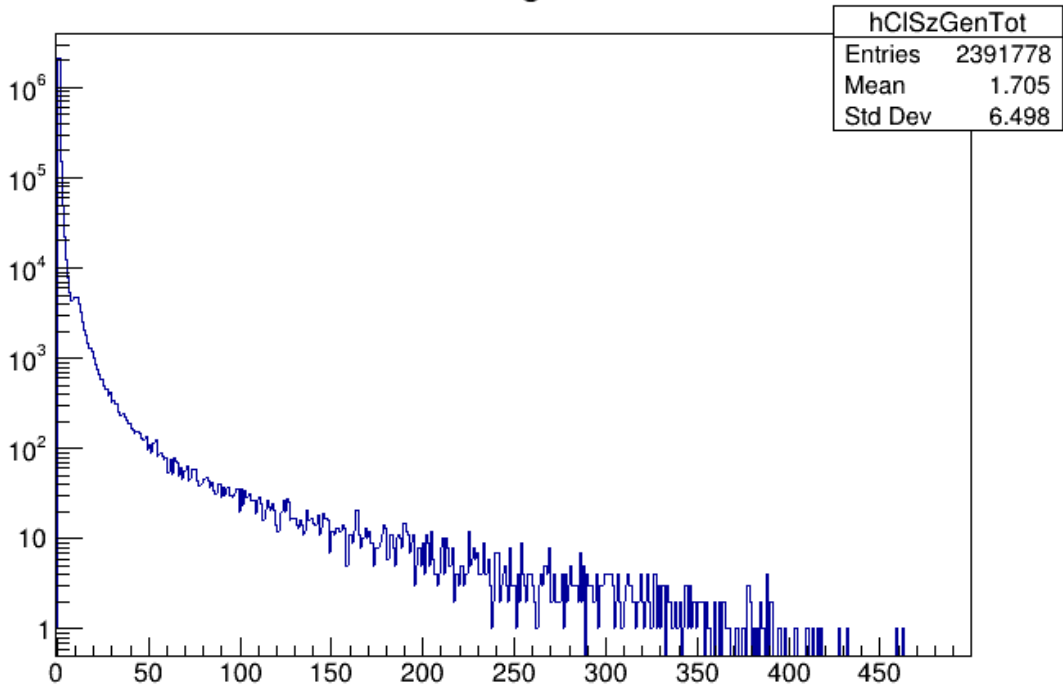


Number of electrons per cluster Rec

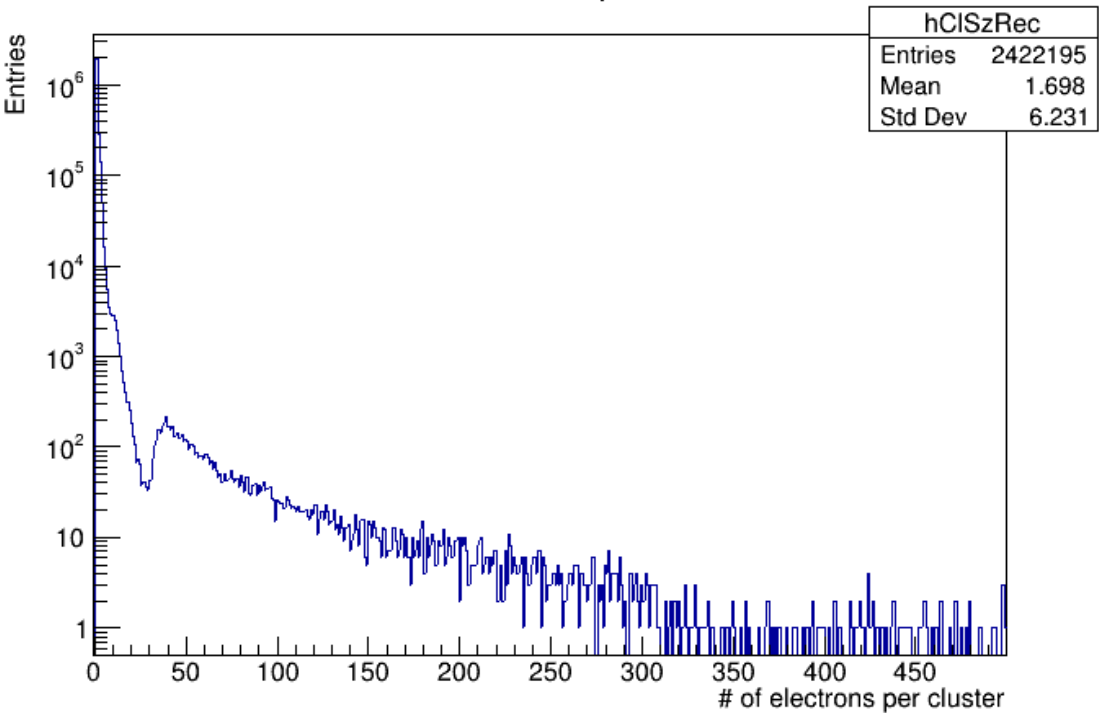


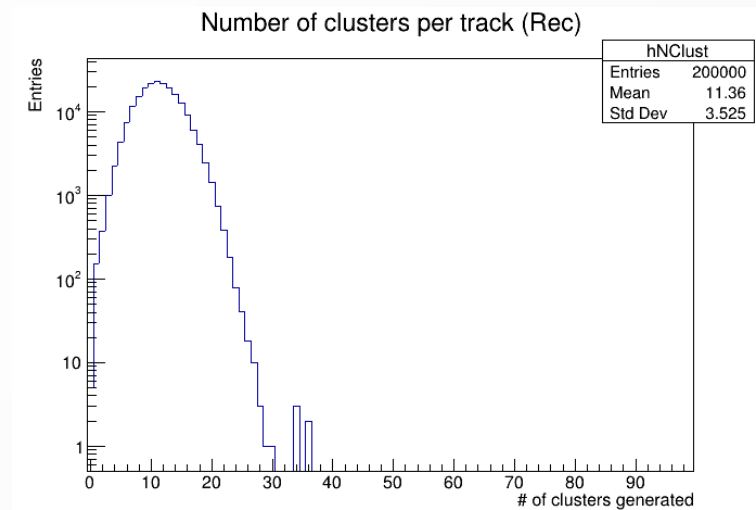
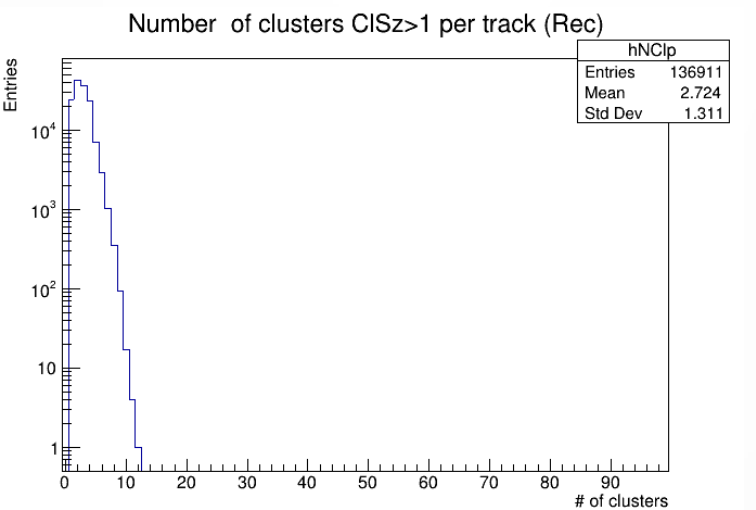
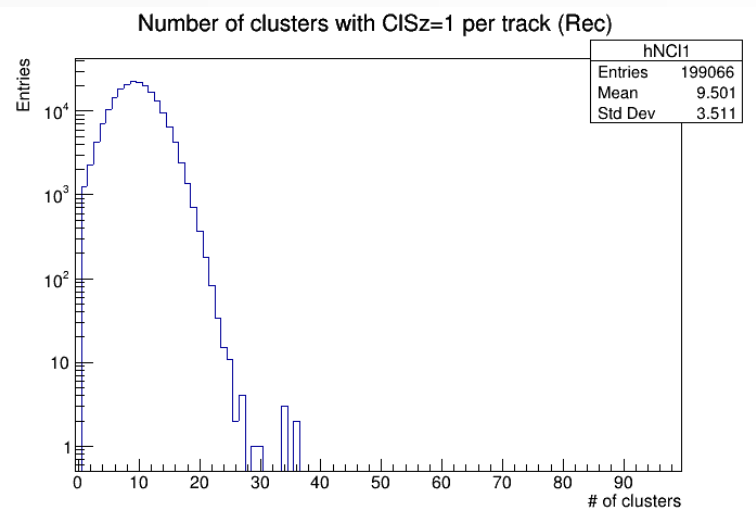
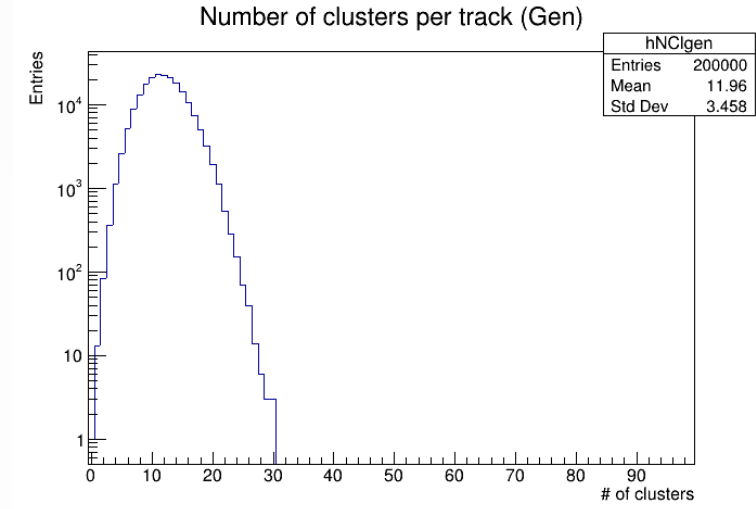
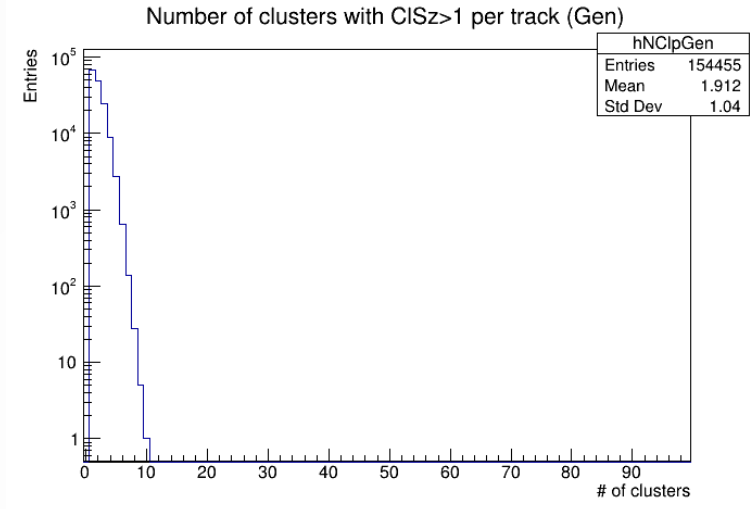
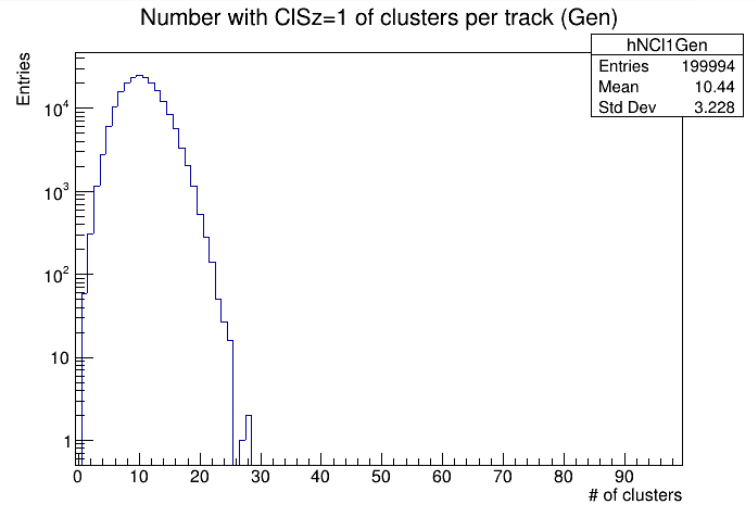


Total CISz generated

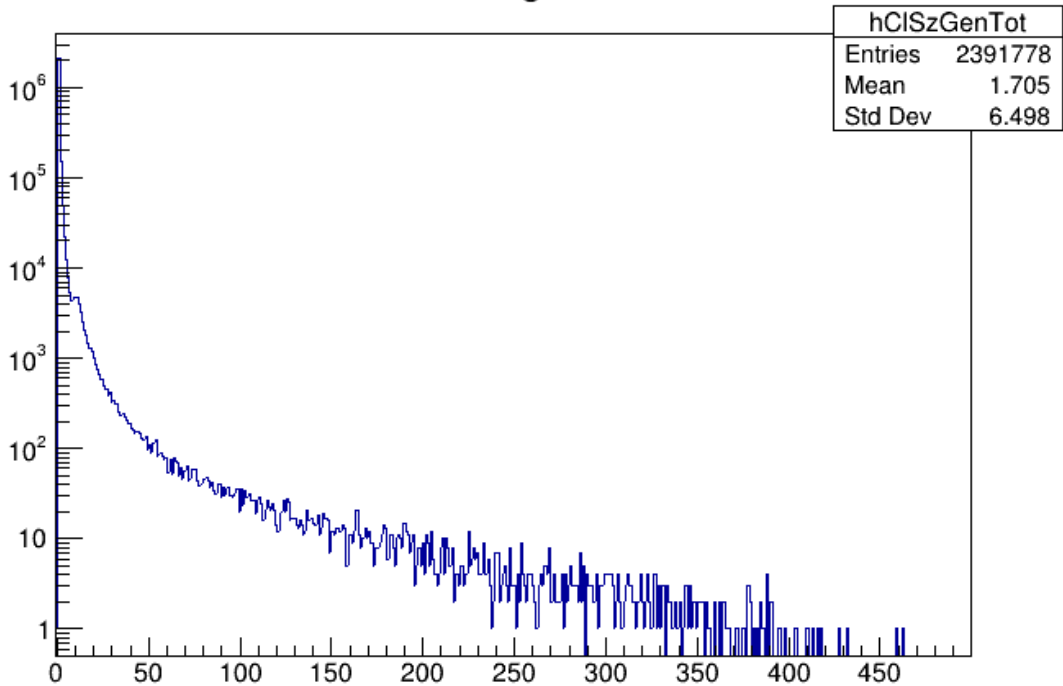


Number of electrons per cluster Rec

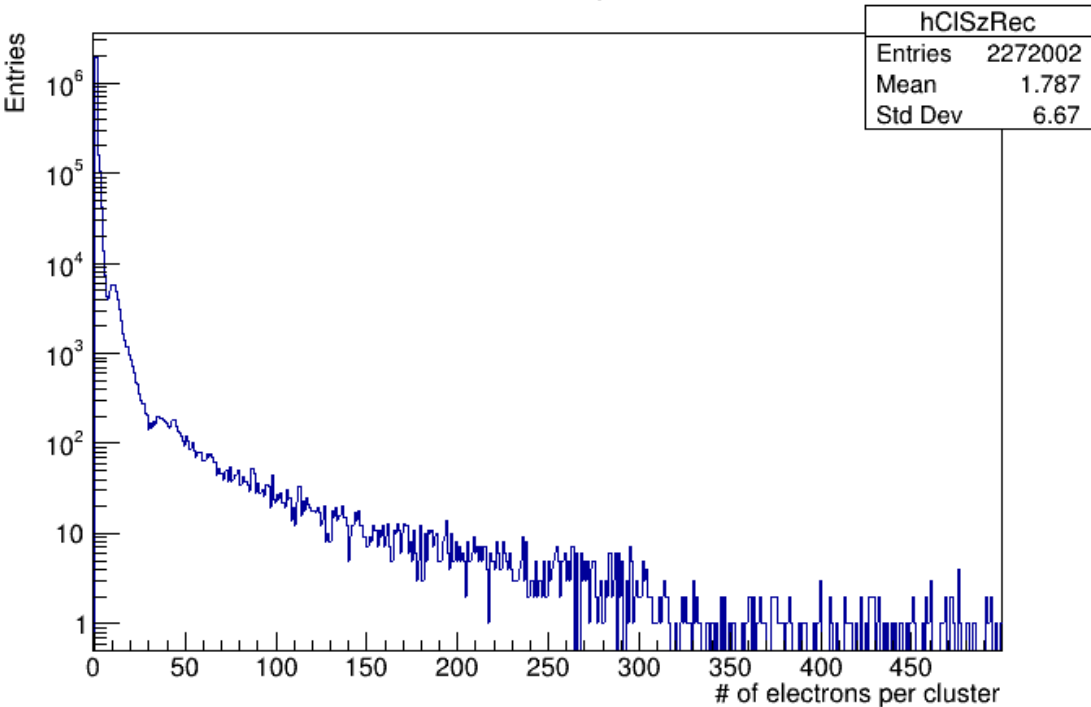




Total CIsZ generated



Number of electrons per cluster Rec



Physics list for geant4

The physics list is “QGSP_BERT” with
G4EmStandardPhysics_option3(),
G4EmExtraPhysics(), G4DecayPhysics(),G4EmLowEPPhysics().